

MSC.Nastran 2004

Reference Manual

Corporate

MSC.Software Corporation
2 MacArthur Place
Santa Ana, CA 92707 USA
Telephone: (800) 345-2078
Fax: (714) 784-4056

Europe

MSC.Software GmbH
Am Moosfeld 13
81829 Munich, Germany
Telephone: (49) (89) 43 19 87 0
Fax: (49) (89) 43 61 71 6

Asia Pacific

MSC.Software Japan Ltd.
Shinjuku First West 8F
23-7 Nishi Shinjuku
1-Chome, Shinjuku-Ku
Tokyo 160-0023, JAPAN
Telephone: (81) (3)-6911-1200
Fax: (81) (3)-6911-1201

Worldwide Web

www.mscsoftware.com

Disclaimer

This documentation, as well as the software described in it, is furnished under license and may be used only in accordance with the terms of such license.

MSC.Software Corporation reserves the right to make changes in specifications and other information contained in this document without prior notice.

The concepts, methods, and examples presented in this text are for illustrative and educational purposes only, and are not intended to be exhaustive or to apply to any particular engineering problem or design. MSC.Software Corporation assumes no liability or responsibility to any person or company for direct or indirect damages resulting from the use of any information contained herein.

User Documentation: Copyright © 2003 MSC.Software Corporation. Printed in U.S.A. All Rights Reserved.

This notice shall be marked on any reproduction of this documentation, in whole or in part. Any reproduction or distribution of this document, in whole or in part, without the prior written consent of MSC.Software Corporation is prohibited.

The software described herein may contain certain third-party software that is protected by copyright and licensed from MSC.Software suppliers.

MSC, MSC/, MSC., MSC.Dytran, MSC.Fatigue, MSC.Marc, MSC.Patran, MSC.Patran Analysis Manager, MSC.Patran CATXPRES, MSC.Patran FEA, MSC.Patran Laminate Modeler, MSC.Patran Materials, MSC.Patran Thermal, MSC.Patran Queue Manager and PATRAN are trademarks or registered trademarks of MSC.Software Corporation in the United States and/or other countries.

NASTRAN is a registered trademark of NASA. PAM-CRASH is a trademark or registered trademark of ESI Group. SAMCEF is a trademark or registered trademark of Samtech SA. LS-DYNA is a trademark or registered trademark of Livermore Software Technology Corporation. ANSYS is a registered trademark of SAS IP, Inc., a wholly owned subsidiary of ANSYS Inc. ABAQUS is a registered trademark of ABAQUS Inc. ACIS is a registered trademark of Spatial Technology, Inc. CATIA is a registered trademark of Dassault Systemes, SA. EUCLID is a registered trademark of Matra Datavision Corporation. FLEXlm is a registered trademark of GLOBEtrotter Software, Inc. HPGL is a trademark of Hewlett Packard. PostScript is a registered trademark of Adobe Systems, Inc. PTC, CADDs and Pro/ENGINEER are trademarks or registered trademarks of Parametric Technology Corporation or its subsidiaries in the United States and/or other countries. Unigraphics, Parasolid and I-DEAS are registered trademarks of Electronic Data Systems Corporation or its subsidiaries in the United States and/or other countries. All other brand names, product names or trademarks belong to their respective owners.

C O N T E N T S

MSC.Nastran Reference Manual

Preface

- List of MSC.Nastran Books, xii
- Technical Support, xiii
- Internet Resources, xv
- Permission to Copy and Distribute MSC Documentation, xvii

1

NASTRAN Command and Input Data

- Introduction, 2
- nastran Command, 7
- The NASTRAN Statement (Optional), 8
- File Management Section (FMS), 9
- Executive Control Section, 10
- Case Control Section, 11
 - Case Control Section Requirements for Solutions, 11
- Bulk Data Section, 34
- Use of Parameters, 35
 - Parameters in the Superelement Solution Sequences, 35

2

Grid Points and Coordinate Systems

- Introduction, 40
- Grid Point and Coordinate System Definition, 41
- Grid Point Properties, 46

3

Structural Elements

- Introduction, 48
- Line Elements, 58
 - Beam Element (CBEAM), 58
 - Bar Element (CBAR), 68
 - Bend Element (CBEND), 69
 - Rod Elements (CROD, CONROD, CTUBE), 73
 - Supplied Beam and Bar Libraries, 75
 - Adding Your Own Beam Cross Section Library, 88
 - Support for Offsets in Element Coordinates for CBAR & CBEAM Elements, 124
 - Guidelines and Limitations, 127

- Example, 127
- Surface Elements, 130
 - Shear Panel Element (CSHEAR), 130
 - Shell Elements (CTRIA3, CTRIA6, CTRIAR, CQUAD4, CQUAD8, CQUADR), 131
 - New QUADR and TRIAR Elements, 140
 - Conical Shell Element (RINGAX), 155
 - Shell Elements Corner Thickness Specification, 159
 - SHELL Structural Damping, 162
- Solid Elements (CTETRA, CPENTA, CHEXA), 163
- Rigid Elements and Multipoint Constraints (R-type, MPC), 167
 - Rigid Element Enhancements, 172
- Scalar Elements (CELASi, CMASSi, CDAMPi), 193
- Axisymmetric Solid Elements (CTRIAX6, CTRIAX, CQUADX), 195
- Crack Tip Elements (CRAC2D, CRAC3D), 197
- Gap and Line Contact Elements (CGAP, BCONP, BLSEG), 203
- Concentrated Mass Elements (CONM1, CONM2), 210
- General Element Capability (GENEL), 211
- Direct Matrix Input on Grid Points (DMIG), 215
- p-Elements, 216
- Hyperelastic Elements, 220
- Bushing Elements, 225
 - Bush Element (CBUSH), 225
 - Bush1d Element (CBUSH1D), 229
- Interface Elements, 232
 - Curve Interface Elements, 232
 - Surface Interface Elements, 235
- CWELD Element Enhancements, 239
- Element Geometry Checks, 252
- ELSUM Case Control Command, 264
 - Introduction, 264
 - Method and Theory, 264
 - Inputs, 265
 - Outputs, 265
 - Guidelines and Limitations, 268
 - Example, 268
- Introduction, 270
- Isotropic Material, 272
- Orthotropic Material, 275
- Anisotropic Material, 277

4

Material Properties

5

Applied Loads

- Nonuniform Material and Composite Properties for Shells, 280
- Hyperelastic Material, 281

- Static Loads, 284
- Point Loads, 285
- Distributed Loads, 286
- Inertia Loads, 287
- Thermal and Enforced Element Deformation Loads, 288
- Loads Due to Enforced Motion, 292
- Frequency-Dependent Loads, 294
- Time-Dependent Loads, 295
- Simplified Static Loading Data in Dynamic Analysis, 297
- Combining of Loads, 300
- p-Element Loads and Constraints, 302
- Summary of Nonlinear Static Loads, 306
- Summary of Nonlinear Transient Loads, 312

6

Constraint and Set Notation

- Degree-of-Freedom Sets, 322
 - Degree-of-Freedom Set Definitions, 322
 - Degree-of-Freedom Set Bulk Data Entries, 325
- Multipoint Constraints, 327
- Single Point Constraints, 329
- Rigid Body Supports, 331
- Sets for Dynamic Reduction, 333
- Sets for Aerodynamics, 335
- Rules of Sets for Undefined Degrees-of-Freedom, 336
- Output Selection Via Set Specification, 338

7

Solution Sequences

- Introduction, 342
- Output Description, 347
- Restart Procedures, 372
- General Operations, 374
 - Geometry Processing in SubDMAP PHASE0, 374
 - Matrix Assembly Operations in SubDMAP SEMG, 377
 - Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB, 379

8

Element Data Recovery Resolved at Grid Points

- Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB, 382
- Constraint and Mechanism Problem Identification in SubDMAP SEKR, 383
- Static Condensation in SubDMAPs SEKR and SEMR2, 388
- Static Solutions in SubDMAP SEKR, 391
- Static and Dynamic Load Generation, 395
- Dynamic Reduction and Component Mode Synthesis in SubDMAP SEMR3, 398
- Formulation of Dynamic Equations in SubDMAP GMA, 403
- Data Recovery Operations in SubDMAP SEDISP, 407
- Data Recovery Operations in SubDMAP SEDRCVR, 413
- Summary of Solution Sequence Operations, 415
 - Static Solutions, 415
 - Real Eigenvalue Analysis in SubDMAPs SEMR3 and MODERS, 423
 - Complex Eigenvalue Analysis in SubDMAP CEIGRS, 440
 - Frequency Response and Random Analysis in SubDMAP FREQRS, 441
 - Transient Response in SubDMAPs SEDTRAN and SEMTRAN, 441
 - Buckling Analysis in SubDMAP MODERS, 442
 - Superelement Analysis, 444
 - Nonlinear Heat Transfer Analysis, 460

9

DMAP

- Introduction, 464
- Stress Recovery at Grid Points, 466
 - Description of Method, 466
 - Topological Method -- Plate Elements and Solid Elements, 467
 - Geometric Method -- Surface Elements, 470
 - Treatment of Exception Points -- "BRANCHing", 472
 - User Information, 474
 - Grid Point Stress Output Description, 475
 - General Remarks, 477
- Mesh Stress Discontinuities at Grid Points, 478

10

Database Concepts

- Introduction, 486
- Introduction, 488
- Delivery Databases, 491
- Dbset Deletion, 492
- Database Autoassignment, 493
- Database Project and Version, 494
- Database Migration, 495
- Database Archival, Compression, and Transfer, 496

11

Plotting

- Database Archival and Retrieval, 497
- Database Compression, 499
- Database Transfer Across Different Computer Types, 500

12

Interface With Other Programs

- Introduction, 502
 - Structure Plotter Coordinate System and Orthographic Projection, 507
 - Perspective Projection, 510
 - Stereoscopic Projection, 511
 - Projection Plane to Plotter Transformations, 511
 - Deformation Scaling, 512
- Postprocessors, 513

- Introduction, 516
- POST Case Control Command, 517
- MSC.Patran Support of Heat Transfer Analysis, 525
 - Loads and Boundary Conditions, 525
 - Results Processing, 526
 - Limitations, 526
 - Documentation, 527

13

Additional Topics

- Introduction, 530
- Composites, 531
 - Introduction, 531
 - Classical Lamination Theory, 534
 - Transverse Shear Theory, 541
 - Data Recovery, 546
 - Failure Indices, 547
 - Strength Ratios, 552
 - "LAM" Options, 557
 - Temperature-Dependent Properties, 561
- Nonlinear Analysis, 567
 - Material Nonlinearity, 583
 - Nonlinear Elastic Material, 584
 - Hyperelastic Material, 585
 - Elastic-Plastic Material, 594
 - Viscoelastic Material in Nonlinear Analysis, 598
 - Temperature-Dependent Material, 598
 - Geometric Nonlinearity, 599
 - Nonlinear Elements, 600
 - Small Strain Elements, 610
 - Fully Nonlinear Hyperelastic Elements, 614
 - Bushing Elements, 629

- Adaptive Gap Element, 633
- 3-D Slideline Contact Capability, 639
- Nonlinear Static Analysis, 649
- Examples of Convergence Tolerance Changes, 657
- Changes to the NLPARM Entry, 658
- Nonlinear Buckling Analysis, 670
- Nonlinear Modal Analysis, 674
- Differential Stiffness and Follower Force Stiffness in Linear and Nonlinear Solution Sequences, 677
- Buckling Analysis Enhancement, 679
- Nonlinear Transient Response Analysis, 691
- Performance Improvements in SOL 129, 707
- Nonlinear Analysis With Superelements, 715
- Grid Point Force Balance and Element Strain Energy in Nonlinear Static Analysis, 715
- Prerelease of the General Nonlinear Solution Sequence SOL 400, 721
- MSC.Nastran Implicit Nonlinear -- SOL 600, 737
 - Introduction, 737
 - Background, 738
 - Example: Two Pipes Crushing (Contact & Large Strain):, 747
- Coupled Fluid-Structure Interaction, 753
 - Coupled Fluid-Structural Analysis, 753
 - Example 1, 768
 - Example 2, 775
 - Modal Participation Example, 784
 - Hydroelastic Modeling, 794
 - The Virtual Mass Method, 794
- Acoustic Cavity Modeling, 795
- Cyclic Symmetry, 803
 - Symmetry, 804
 - Theory, 809
- p-Elements, 829
 - Introduction to p-Elements, 829
 - p-Version Capabilities, 831
 - p-Element Example Problems, 837
 - Square Plate with a Circular Hole, 837
 - Spherical Structure Under Internal Pressure, 842
 - Motor Blade – Normal Modes, 846
 - Torsion of Elliptical Bar, 849
 - Circular Solid Model, 860
 - Clamped Rhombic Plate, 874
 - Helix Modeled with CBEAM p-Elements, 875
 - Plate with Stiffener, 877
 - Solid p-Elements in Linear Dynamic Solution Sequences, 880
 - p-Element Thermal Analysis, 882
- Response Spectrum Analysis, 886
- Viscoelastic Materials in Frequency Response Analysis, 887
- Introduction to Parallel Numeric Enhancements, 888
 - Domain Decomposition, 890

- New Distributed Memory Parallel Methods, 891
- Inputs, 893
- Method Selection Guidelines, 897
- Limitations, 898
- Examples, 900
- Parallel Performance Data, 904
- Other Numeric Enhancements, 904
- MSC.Nastran/MSC.ADAMS Interface, 906
- Rotordynamics, 951
 - Introduction, 951
 - Model Restrictions, 962
 - Examples, 963
 - Equations Used in Analyses, 970
- References, 978

INDEX

Preface

- List of MSC.Nastran Books
- Technical Support
- Internet Resources
- Permission to Copy and Distribute MSC Documentation

List of MSC.Nastran Books

Below is a list of some of the MSC.Nastran documents. You may order any of these documents from the MSC.Software BooksMart site at www.engineering-e.com.

Installation and Release Guides

- Installation and Operations Guide
- Release Guide

Reference Books

- Quick Reference Guide
- DMAP Programmer's Guide
- Reference Manual

User's Guides

- Getting Started
- Linear Static Analysis
- Basic Dynamic Analysis
- Advanced Dynamic Analysis
- Design Sensitivity and Optimization
- Thermal Analysis
- Numerical Methods
- Aeroelastic Analysis
- Superelement
- User Modifiable
- Toolkit

Technical Support

For help with installing or using an MSC.Software product, contact your local technical support services. Our technical support provides the following services:

- Resolution of installation problems
- Advice on specific analysis capabilities
- Advice on modeling techniques
- Resolution of specific analysis problems (e.g., fatal messages)
- Verification of code error.

If you have concerns about an analysis, we suggest that you contact us at an early stage.

You can reach technical support services on the web, by telephone, or e-mail:

Web Go to the MSC.Software website at www.mscsoftware.com, and click on **Support**. Here, you can find a wide variety of support resources including application examples, technical application notes, available training courses, and documentation updates at the MSC.Software Training, Technical Support, and Documentation web page.

**Phone
and
Fax**

United States

Telephone: (800) 732-7284

Fax: (714) 784-4343

Frimley, Camberley

Surrey, United Kingdom

Telephone: (44) (1276) 67 10 00

Fax: (44) (1276) 69 11 11

Munich, Germany

Telephone: (49) (89) 43 19 87 0

Fax: (49) (89) 43 61 71 6

Tokyo, Japan

Telephone: (81) (3) 3505 02 66

Fax: (81) (3) 3505 09 14

Rome, Italy

Telephone: (390) (6) 5 91 64 50

Fax: (390) (6) 5 91 25 05

Paris, France

Telephone: (33) (1) 69 36 69 36

Fax: (33) (1) 69 36 45 17

Moscow, Russia

Telephone: (7) (095) 236 6177

Fax: (7) (095) 236 9762

Gouda, The Netherlands

Telephone: (31) (18) 2543700

Fax: (31) (18) 2543707

Madrid, Spain

Telephone: (34) (91) 5560919

Fax: (34) (91) 5567280

Email Send a detailed description of the problem to the email address below that corresponds to the product you are using. You should receive an acknowledgement that your message was received, followed by an email from one of our Technical Support Engineers.

MSC.Patran Support	mscpatran.support@mscsoftware.com
MSC.Nastran Support	mscnastran.support@mscsoftware.com
MSC.Nastran for Windows Support	vn4w.support@mscsoftware.com
MSC.visualNastran Desktop 2D Support	vn2d.support@mscsoftware.com
MSC.visualNastran Desktop 4D Support	vndesktop.support@mscsoftware.com
MSC.Abaqus Support	mscabaqus.support@mscsoftware.com
MSC.Dytran Support	mscdytran.support@mscsoftware.com
MSC.Fatigue Support	mscfatigue.support@mscsoftware.com
MSC.Interactive Physics Support	ip.support@mscsoftware.com
MSC.Marc Support	mscmarc.support@mscsoftware.com
MSC.Mvision Support	mscmvision.support@mscsoftware.com
MSC.SuperForge Support	mscsuperforge.support@mscsoftware.com
MSC Institute Course Information	msctraining.support@mscsoftware.com

Training

The MSC Institute of Technology is the world's largest global supplier of CAD/CAM/CAE/PDM training products and services for the product design, analysis and manufacturing market. We offer over 100 courses through a global network of education centers. The Institute is uniquely positioned to optimize your investment in design and simulation software tools.

Our industry experienced expert staff is available to customize our course offerings to meet your unique training requirements. For the most effective training, The Institute also offers many of our courses at our customer's facilities.

The MSC Institute of Technology is located at:

2 MacArthur Place
Santa Ana, CA 92707
Phone: (800) 732-7211
Fax: (714) 784-4028

The Institute maintains state-of-the-art classroom facilities and individual computer graphics laboratories at training centers throughout the world. All of our courses emphasize hands-on computer laboratory work to facility skills development.

We specialize in customized training based on our evaluation of your design and simulation processes, which yields courses that are geared to your business.

In addition to traditional instructor-led classes, we also offer video and DVD courses, interactive multimedia training, web-based training, and a specialized instructor's program.

Course Information and Registration. For detailed course descriptions, schedule information, and registration call the Training Specialist at (800) 732-7211 or visit www.mscsoftware.com.

Internet Resources

MSC.Software (www.mscsoftware.com)

MSC.Software corporate site with information on the latest events, products and services for the CAD/CAE/CAM marketplace.

Simulation Center (simulate.engineering-e.com)

Simulate Online. The Simulation Center provides all your simulation, FEA, and other engineering tools over the Internet.

Engineering-e.com (www.engineering-e.com)

Engineering-e.com is the first virtual marketplace where clients can find engineering expertise, and engineers can find the goods and services they need to do their job

CATIASOURCE (plm.mscsoftware.com)

Your SOURCE for Total Product Lifecycle Management Solutions.



Permission to Copy and Distribute MSC Documentation

If you wish to make copies of this documentation for distribution to co-workers, complete this form and send it to MSC.Software Corporation. MSC will grant written permission if the following conditions are met:

- All copyright notices must be included on all copies.
- Copies may be made only for fellow employees.
- No copies of this manual, or excerpts thereof, will be given to anyone who is not an employee of the requesting company.

Please complete and mail to MSC for approval:

MSC.Software Corporation
Attention: Legal Department
2 MacArthur Place
Santa Ana, CA 92707

Name: _____

Title: _____

Company: _____

Address: _____

Telephone: _____ Email: _____

Signature: _____ Date: _____

Please do not write below this line.

APPROVED: MSC.Software Corporation

Name: _____

Title: _____

Signature: _____ Date: _____

Fold here

Place
Stamp
Here

MSC.Software Corporation
Attention: Legal Department
2 MacArthur Place
Santa Ana, CA 92707

Fold here

CHAPTER

1

NASTRAN Command and Input Data

- Introduction
- nastran Command
- The NASTRAN Statement (Optional)
- File Management Section (FMS)
- Executive Control Section
- Case Control Section
- Bulk Data Section
- Use of Parameters

1.1 Introduction

nastran Command

The recommended procedure for invoking MSC.Nastran is to use the nastran command. The nastran command is described in “[nastran Command and NASTRAN Statement](#)” on page 1 of the *MSC.Nastran Quick Reference Guide*.

Input Data

MSC.Nastran input requires records that are 80 characters (or columns) in length. The input file is comprised of five sections, which must be assembled in the sequence shown in [Figure 1-1](#).

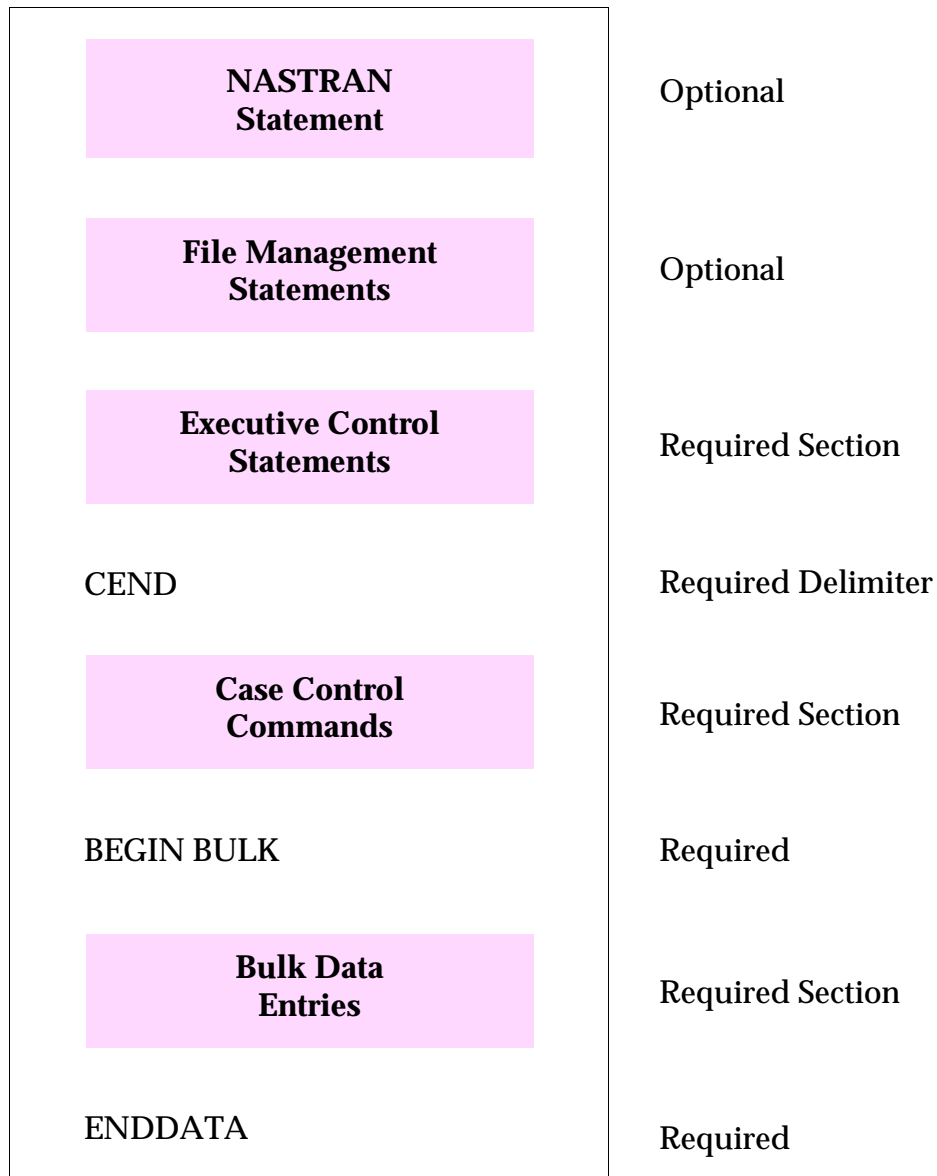


Figure 1-1 Structure of the MSC.Nastran Input File

The records of the first four sections are input in free-field format and only columns 1 through 72 are used for data. Any information in columns 73 through 80 may appear in the printed echo, but will not be used by the program. If the last character in a record is a comma, then the record is continued to the next record.

The NASTRAN statement is optional and is used in special circumstances (see the “[nastran Command and NASTRAN Statement](#)” on page 1 of the *MSC.Nastran Quick Reference Guide*).

The File Management Section (FMS) is optional and typically follows the NASTRAN statement(s). It ends with the specification of an Executive Control statement. This section provides for database initialization and management, along with job identification and restart conditions. The File Management statements are described in the “[File Management Statements](#)” on page 31 of the *MSC.Nastran Quick Reference Guide*.

The Executive Control Section begins with the first Executive Control statement and ends with the CEND delimiter. It identifies the job and the type of solution to be performed. It also declares the general conditions under which the job is to be executed, such as maximum time allowed and type of system diagnostics desired. If the job is to be executed with a solution sequence, the actual solution sequence is declared, along with any alterations to the solution sequence that may be desired. If DMAP (Direct Matrix Abstraction Program) is used, the complete DMAP sequence must appear in the Executive Control Section. The Executive Control statements and examples of their use are described in the “[Executive Control Statements](#)” on page 99 of the *MSC.Nastran Quick Reference Guide*.

The Case Control Section follows CEND and ends with the BEGIN BULK delimiter. It defines the subcase structure for the problem, defines sets of Bulk Data, and makes output requests for printing, punching and plotting. A general discussion of the functions of the Case Control Section and a detailed description of the commands used in this section are given in the “[Case Control Commands](#)” on page 175 of the *MSC.Nastran Quick Reference Guide*. The special requirements of the Case Control for each solution sequence are discussed in “[Restart Procedures](#)” on page 372.

The Bulk Data Section follows BEGIN BULK and ends with the ENDDATA delimiter. It contains all of the details of the model and the conditions for the solution. BEGIN BULK and ENDDATA must be present even though no new Bulk Data is being introduced into the problem or if all of the Bulk Data is coming from an alternate source, such as user-generated input. The BEGIN BULK entry is in free field format. The ENDDATA delimiter must begin in Column 1 or Column 2.

The input file may be formed by the insertion of other files, with the INCLUDE statement. This may be specified in any of the five parts of the input file. Note, however, the data referenced by the INCLUDE statement must be appropriate for that section of the input file.

Comments may be inserted in any of the parts of the input file. They are identified by a dollar sign (\$) in any column followed by any characters out to Column 80. Columns 2 through 72 may contain any desired text.

[Table 1-1](#) gives valid characters for MSC.Nastran.

Table 1-1 ASCII Subset for MSC.Nastran

Character	ASCII (HEX)	Character	ASCII (HEX)
HT (Horizontal Tabulation)	09	@	40
SP (Space "blank")	20	A	41
!	21	B	42
"	22	C	43
#	23	D	44
\$	24	E	45
%	25	F	46
&	26	G	47
'	27	H	48
(28	I	49
)	29	J	4A
*	2A	K	4B
+	2B	L	4C
,	2C	M	4D
-	2D	N	4E
.	2E	O	4F
/	2F	P	50
0	30	Q	51
1	31	R	52
2	32	S	53
3	33	T	54
4	34	U	55
5	35	V	56
6	36	W	57
7	37	X	58
		Y	59
		Z	5A

Table 1-1 ASCII Subset for MSC.Nastran (continued)

Character	ASCII (HEX)	Character	ASCII (HEX)
8	38	[5B
9	39	\	5C
:	3A]	5D
;	3B	^	5E
<	3C	-	5F
=	3D		
>	3E		
?	3F		

1.2 nastran Command

MSC.Nastran is executed with a command called `nastran`. (Your system manager may assign a different name to the command.) The `nastran` command permits the specification of keywords used to request options affecting MSC.Nastran job execution. The format of the `nastran` command is:

```
nastran keyword1 = value1 keyword2 = value2 ...
```

The `nastran` command description of keywords also appears in “[nastran Command and NASTRAN Statement](#)” on page 1 of the *MSC.Nastran Quick Reference Guide*.

1.3 The NASTRAN Statement (Optional)

The NASTRAN statement is used to change the default values for certain operational parameters (also called system cells). See the “[nastran Command and NASTRAN Statement](#)” on page 1 of the *MSC.Nastran Quick Reference Guide* for a description of the format and most often used system cells. The NASTRAN statement is used for special circumstances and is therefore not needed in most runs. The NASTRAN statement may also be specified in the Runtime Configuration (RC) files at the system, user, and job level. See the *MSC.Nastran 2004 Installation and Operations Guide*.

1.4 File Management Section (FMS)

The File Management Section (FMS) is primarily intended for the attachment and initialization of Database sets (DBsets) and FORTRAN files. The initialization of DBsets includes specification of their maximum size, member names, and physical filenames. The initialization of FORTRAN files includes the specification of their filenames, FORTRAN unit numbers, and FORTRAN attributes.

In most classes of problems that use MSC.Nastran solution sequences, no File Management statements are required because a default File Management Section is executed at the beginning of every run. The default File Management Section is described in “[Database Concepts](#)” on page 487. See “[File Management Statements](#)” on page 31 of the *MSC.Nastran Quick Reference Guide* for a detailed description.

1.5 Executive Control Section

The Executive Control statements, in general, select the solution sequences to be run and various diagnostics. The input specifications are given in “[Executive Control Statements](#)” on page 99 of the *MSC.Nastran Quick Reference Guide*.

Most statements are order independent. Exceptions are the COMPILE, COMPILER, ALTER, ENDALTER, and LINK statements. The LINK statement must appear after all COMPILE statements. The COMPILER statement (or equivalent DIAGs) must appear before all COMPILE statements. This statement sets the default print control for subsequent COMPILE statements.

1.6 Case Control Section

The Case Control Section defines the subcase structure for the problem, makes selections from the Bulk Data Section, and makes output requests. Grid point stress and stress discontinuities output request are treated separately in “[Stress Recovery at Grid Points](#)” on page 466 and “[Mesh Stress Discontinuities at Grid Points](#)” on page 478, respectively. Structure plotter output requests and curve plot requests are treated separately in [Section 11.2](#) and [Section 11.3](#), respectively.

The Case Control commands are fully described in the “[Case Control Commands](#)” on page 175 of the *MSC.Nastran Quick Reference Guide*.

The Case Control Section has several basic functions; specifically, it:

- Selects loads and constraints.
- Requests printing, plotting, and/or punching of input and output data (plotter commands are described in “[Plotting](#)” on page 501).
- Defines the subcase structure for the analysis.

1.6.1 Case Control Section Requirements for Solutions

This section describes the options and requirements in the Case Control Section for the various solution sequences.

In general, a separate subcase is defined for each loading condition and/or each set of constraints. Subcases may also be used in connection with output requests, such as requesting different output for each mode in a real eigenvalue problem. Only one level of subcase definition is provided. All items placed above the subcase level (ahead of the first subcase) will be used for all following subcases unless overridden within the individual subcase.

Case Control Commands for Solutions

This section describes the Case Control structure for the various solution sequences.

Static Analysis (SOLutions 101, 114, 144, and 200)

1. A separate subcase must be defined for each unique combination of constraints and static loads. Multiple constraint sets are permitted in SOLutions 101, and 200 (for ANALYSIS = STATICS only) and for residual structures only.

2. A static loading condition must be defined for each subcase with a LOAD, TEMPERATURE(LOAD) or DEFORM selection unless all loading is specified with grid point displacements on SPC entries. In SOLution 144, a loading condition is optional, but a trim variable constraint must be selected with the TRIM command.
3. An SPC set must be selected for each subcase, unless the model is a properly supported free body, or all constraints are specified on GRID entries, scalar connection entries, or with general elements.
4. REPCASE may be used to repeat subcases in order to allow multiple sets of the same output item.
5. Direct input matrices may be selected with K2GG and/or M2GG above the subcase level. P2G is allowed in all the above SOLutions.
6. In SOLutions 101 and 200, the SUPORT1 command may be used to select SUPORT1 Bulk Data entries for inertia relief analysis. See “[Superelement Subcase Structure](#)” on page 24 for additional Case Control requirements for superelements.
7. For an adaptive p-version element solution (multiple p-levels) the ADAPT command must select an ADAPT Bulk Data entry. The ADAPT command may be specified to limit the subcases participating in the adaptive analysis. For a single p-level analysis the ADAPT command must select a PSET entry. The DATAREC and OUTRCV commands control the output of p-element stresses, strains, and displacements. The VUGRID command requests output at view-elements and view-grids.
8. The SMETHOD command may be used to select parameters for an iterative solution technique.

In statics problems, provision has been made for the combination of the results of several subcases. This is convenient for studying various combinations of individual loading conditions and for the superposition of solutions for symmetrical and antisymmetrical boundaries.

The following examples of Case Control indicate typical ways of defining subcases:

1. Static analysis with multiple loads and boundary conditions (Note that multiple boundary conditions are supported only in SOLutions 101, 103, 105, 145, and 200). See the BC Case Control command.

```
DISPLACEMENT = ALL
MPC = 3
SUBCASE 1
  SPC = 2
  TEMPERATURE(LOAD) = 101
  LOAD = 11
SUBCASE 2
  SPC = 2
  DEFORM = 52
  LOAD = 12
SUBCASE 3
  SPC = 4
  LOAD = 12
SUBCASE 4
  MPC = 4
  SPC = 4
```

Four subcases are defined in this example. The displacements at all grid points will be printed for all four subcases. MPC = 3 will be used for the first three subcases and will be overridden by MPC = 4 in the last subcase. Since the constraints are the same for subcases 1 and 2 and the subcases are contiguous, the static solutions will be performed simultaneously. In SUBCASE 1, thermal load 101 and external load 11 are internally superimposed, as are the external and deformation loads in SUBCASE 2. In SUBCASE 4, the static loading will result entirely from enforced displacements of grid points.

2. Linear combination of subcases

```
SPC = 2
SET 1 = 1 THRU 10,20,30
DISPLACEMENT = ALL
STRESS = 1
SUBCASE 1
  LOAD = 101
  OLOAD = ALL
SUBCASE 2
  LOAD = 201
  OLOAD = ALL
SUBCOM 51
  SUBSEQ = 1.0,1.0
SUBCOM 52
  SUBSEQ = 2.5,1.5
```

Two static loading conditions are defined in subcases 1 and 2. SUBCOM 51 defines the sum of subcases 1 and 2. SUBCOM 52 defines a linear combination consisting of 2.5 times SUBCASE 1 plus 1.5 times SUBCASE 2. The displacements at all grid points and the stresses for the element numbers in SET 1 will be printed for all four subcases. In addition, the nonzero components of the static load vectors will be printed for subcases 1 and 2.

3. Statics with one plane of symmetry. (Note that this is not supported in the superelement solution sequences.)

```

SET 1 = 1,11,21,31,51
SET 2 = 1 THRU 10, 101 THRU 110
DISPLACEMENT = 1
ELFORCE = 2
SYM 1
  SPC = 11
  LOAD = 21
  OLOAD = ALL
SYM 2
  SPC = 12
  LOAD = 22
SYMCOM 3
SYMCOM 4
SYMSEQ 1.0,-1.0

```

Two SYM subcases are defined in subcases 1 and 2. SYMCOM 3 defines the sum and SYMCOM 4 the difference of the two SYM subcases. The nonzero components of the static load will be printed for SUBCASE 1 and no output is requested for SUBCASE 2. The displacements for the grid point numbers in SET 1 and the forces for elements in SET 2 will be printed for subcases 3 and 4.

4. Use of REPCASE in statics problems

```

SET 1 = 1 THRU 10, 101 THRU 110,201 THRU 210
SET 2 = 21 THRU 30,121 THRU 130,221 THRU 230
SET 3 = 31 THRU 40,131 THRU 140,231 THRU 240
SUBCASE 1
  LOAD = 10
  SPC = 11
  DISP = ALL
  SPCFORCE = 1
  FORCE = 1
REPCASE 2
  FORCE = 2
REPCASE3
  FORCE=3

```


This example defines one subcase for solution and two subcases for output control. The displacements at all grid points and the nonzero components of the single-point forces of constraint along with forces for the elements in SET 1 will be printed for SUBCASE 1. The forces for elements in SET 2 will be printed for REPCASE 2 and the forces for elements in SET 3 will be printed for REPCASE 3.

Normal Modes (SOLutions 103, 115, and 200)

1. METHOD must be used to select an EIGR or EIGRL entry that exists in the Bulk Data Section. In SOLutions 103 and 200 multiple subcases may be specified with different sets of METHOD, MPC, SPC, and SUPORT1 commands. In all other SOLs only one set may be specified. (See the BC Case Control command.)
2. An SPC set must be selected unless the model is a free body or all constraints are specified on GRID entries, scalar connection entries or with general elements.
3. Multiple subcases may be used to control output requests. A single subcase is sufficient if the same output is desired for all modes. If multiple subcases are present, the output requests will be honored in succession for increasing mode numbers. MODES may be used to repeat subcases in order to make the same output request for several consecutive modes.
4. Direct input matrices may be selected with K2GG and/or M2GG above the subcase level.
5. In SOLutions 103, and 200, the SUPORT1 command may be used to select SUPORT1 Bulk Data entries for computing rigid body modes.
6. See “[Superelement Subcase Structure](#)” on page 24 for additional Case Control requirements for superelements.
7. Use of MODES in Eigenvalue Problems

```
METHOD = 2
SPC = 10
SUBCASE 1
  DISP = ALL
  STRESS = ALL
  MODES = 2
SUBCASE 3
  DISP = ALL
```

In this example, the displacements at all grid points will be printed for all modes. The stresses in all elements will be printed for the first two modes. If subcases are not present, output requests act on all computed eigenvectors.

8. Use of REPCASE in Normal Modes Analysis

In normal modes analysis the REPCASE command is applied only to the last mode in the previous subcase.

```

SET 200 = 1,7,10,15
DISP(PLOT) = ALL
SUBCASE 1
    LABEL = MODE 1
REPCASE 2
    LABEL = REPCASE FOR MODE 1
    DISP = 200
SUBCASE 3
    LABEL = MODE 2
REPCASE 4
    LABEL = REPCASE FOR MODE 2
    DISP = 200

```

This example requests the output of the first two eigenvectors. In the SUBCASEs, output is computed at all points for plotting purposes. In the REPCASEs, output is printed only at selected points.

9. Prestressed structures may be analyzed by specifying static subcases to define the pre-loads. Then the STATSUB(PRELOAD) command that references the desired static subcases must be specified in the dynamic response subcases to request the inclusion of differential stiffness in the dynamic formulation. By default, stiffness due to follower forces is also included in the differential stiffness.
10. For an adaptive p-version element solution (multiple p-levels) the ADAPT command must select an ADAPT Bulk Data entry. The ADAPT command may be specified to limit the subcases participating in the adaptive analysis. For a single p-level analysis the ADAPT command must select a PSET entry. The DATAREC and OUTRCV commands control the output of p-element stresses, strains, and displacements. The VUGRID command requests output at view-elements and view-grids.
11. For fluid-structure analysis, the METHOD(FLUID) and SDAMPING(FLUID) may be used to request eigensolution and modal damping parameters for the fluid which are different from the structure.

Buckling Analysis (SOLutions 105 and 200)

1. The Case Control Section must contain at least two subcases. In SOL 105, multiple subcases may be specified with different sets of METHOD, SPC, and MPC commands. See also the BC Case Control command. The buckling subcase(s) may have a different boundary condition than the statics subcase(s).
2. In SOL 105 and 200, the METHOD and STATSUB commands must be specified in one or more subcases. In SOLution 105 and 200, two different STATSUB conditions may be selected: STATSUB(PRELOAD) for prestressing and STATSUB(BUCKLE) for determining the buckling loading condition. Both may appear in the same subcase with a METHOD command.
3. A static loading condition must be defined in the first subcase with a LOAD, TEMPERATURE(LOAD), or DEFORM selection, unless all loading is specified by grid point displacements on SPC entries.
4. Output requests that apply to both the static solution and the buckling modes may be placed above the subcase level.

Modal Formulation (SOLutions 110, 111, 112, 145, 146, and 200)

1. An SPC set must be selected above the subcase level unless the model is a free body or all constraints are specified on GRID entries, scalar connection entries, or with general elements.
2. SDAMPING must be used to select a TABDMP1 entry if modal damping is desired. Only one selection is allowed in superelement solution sequences.
3. For SOLutions 110, 111, and 112, STATSUB selects a static solution used to form differential stiffness.
4. For SOLution 145, see also the BC Case Control command.

Complex Eigenvalue and Flutter Analysis (SOLutions 107, 110, 145, and 200)

1. At least one subcase must be defined for each unique set of direct input matrices (K2PP, M2PP, B2PP). Only one set is allowed in SOLutions 107 and 110. More than one set may be selected in SOLution 145.
2. Multiple subcases for each set of direct input matrices are used only to control output requests. A single subcase for each set of direct input matrices is sufficient if the same output is desired for all modes. If consecutive multiple subcases are present for a single set of direct input matrices, the output requests will be honored in succession for increasing mode numbers. MODES may be used to repeat subcases in order to make the same output request for several consecutive modes.

3. CMETHOD must be used to select an EIGC entry in the Bulk Data Section for each set of direct input matrices. Only one selection is allowed in SOLUTIONS 107 and 110. In SOLUTION 145, CMETHOD is required only if k-method flutter analysis is also selected.
4. Constraints must be defined above the subcase level for SOLUTIONS 107 and 110. Multiple boundary conditions are allowed for SOLUTION 145 with a BC Case Control command.
5. In SOLUTION 145, FMETHOD must be used to select a FLUTTER entry in the Bulk Data Section.
6. Prestressed structures may be analyzed by specifying static subcases to define the pre-loads. Then the STATSUB(PRELOAD) command that references the desired static subcases must be specified in the dynamic response subcases to request the inclusion of differential stiffness in the dynamic formulation. By default, stiffness due to follower forces is also included in the differential stiffness.
7. For fluid-structure analysis, the METHOD(FLUID) and SDAMPING(FLUID) may be used to request eigensolution and modal damping parameters for the fluid which are different from the structure.
8. The SMETHOD command may be used to select parameters for an iterative solution technique.

Frequency, Random, and Aeroelastic Response (SOLUTIONS 108, 111, 118, 146, and 200)

1. At least one subcase must be defined for each unique set of direct input matrices (K2PP, M2PP, B2PP) or frequencies. Only one set is allowed in SOLUTIONS 108, 111, 118, 146, and 200.
2. Consecutive subcases for each set of direct input matrices or frequencies are used to define the loading conditions - one subcase for each dynamic loading condition.
3. Constraints must be defined above the subcase level.
4. DLOAD must be used to select a frequency-dependent loading condition DLOAD and RLOADi Bulk Data entries, for each subcase. In SOLUTIONS 108, 111, and 146, DLOAD may also select a time-dependent loading condition, i.e., TLOADi entries.
5. FREQUENCY must be used to select a set of FREQ, FREQ1, FREQ2, FREQ3, FREQ4, or FREQ5 entries from the Bulk Data Section for each unique set of direct input matrices. Only one selection is allowed in SOLUTIONS 118 and 146.

6. OFREQUENCY may be used above the subcase level or within each subcase to select a subset of the solution frequencies for output requests. The default is to use all solution frequencies.
7. If Random Response calculations are desired, RANDOM must be used to select RANDPS and RANDTi entries from the Bulk Data Section. Only one OFREQUENCY and FREQUENCY command can be used for each set of direct input matrices.
8. LOADSET may be used in the first subcase to select a set of LSEQ entries which define a set of static load vectors to be used as DAREA entries in dynamic analysis.
9. Prestressed structures may be analyzed by specifying static subcases to define the pre-loads. Then the STATSUB(PRELOAD) command that references the desired static subcases must be specified in the dynamic response subcases to request the inclusion of differential stiffness in the dynamic formulation. By default, stiffness due to follower forces is also included in the differential stiffness.
10. Multiple subcases with different SPC, MPC and SUPORT1 requests may be specified in SOLs 108 and 111.

Transient Response (SOLutions 109, 112, and 200)

1. One subcase must be defined for each dynamic loading condition. DLOAD or NONLINEAR must be used to define a time-dependent loading condition for each subcase. Only one DLOAD and NONLINEAR command is allowed in SOLution 200.
2. Constraints must be defined above the subcase level.
3. TSTEP must be used to select the time-step intervals to be used for integration and output in each subcase. Only one TSTEP selection is allowed in SOLution 200.
4. If nonzero initial conditions are desired, IC must be used to select a TIC entry in the Bulk Data Section.
5. LOADSET may be used in the first subcase to select a set of LSEQ entries which define a set of static load vectors to be used as DAREA entries in dynamic analysis.

6. Prestressed structures may be analyzed by specifying static subcases to define the pre-loads. Then the STATSUB(PRELOAD) command that references the desired static subcases must be specified in the dynamic response subcases to request the inclusion of differential stiffness in the dynamic formulation. By default, stiffness due to follower forces is also included in the differential stiffness.
7. At least one subcase must be defined for each unique set of direct input matrices (K2PP, M2PP, B2PP), dynamic loading (DLOAD) and time step parameters (TSTEP). Only one set of direct input matrices is allowed.

Linear Steady State Heat Transfer (SOLution 101)

1. A separate subcase must be defined for each unique combination of constraints and static loads.
2. A static loading condition must be defined for (not necessarily within) each subcase with a LOAD selection, unless all loading is specified with grid point temperatures on SPC entries.
3. An SPC set must be selected for (not necessarily within) each subcase, unless all constraints are specified on GRID entries or scalar connection entries.
4. REPCASE may be used to repeat subcases in order to allow multiple sets of the same output item.

Nonlinear Steady State Heat Transfer (SOLution 153)

1. A separate subcase must be defined for each unique combination of temperature loads (LOAD Case Control command), temperature constraints (SPC and MPC command), and nonlinear iteration strategy (NLPARM command). ANALYSIS = HEAT must be specified above the subcase level.
2. The LOAD Case Control command references the static temperature load entries: QVECT, QVOL, QHBDY, and QBDYi.
3. An estimated temperature distribution must be defined on TEMP or TEMPD entries and selected by the TEMPERATURE(INITIAL) Case Control command. Temperature constraints are specified on SPCi entries.
4. Restarts are controlled by the PARAMeters SUBCASID, SUBID and LOOPID.
5. See the *[MSC.Nastran Thermal Analysis User's Guide](#)* for further details.

Nonlinear Transient Heat Transfer (SOLution 159)

1. Only one set of temperature constraints (MPC and SPC Case Control command) may be requested and must be specified above the subcase level. ANALYSIS = HEAT must be specified above the subcase level.
2. A subcase must be defined for each unique combination of dynamic temperature load conditions (DLOAD command) and nonlinear iteration strategy (TSTEPNL command).
3. The DLOAD and/or NONLINEAR command must be used to specify a time-dependent loading condition. The static temperature load entries: QVECT, QVOL, QHBDY, and QBDYi may be used in defining a dynamic load specified by the TLOADi entry. The set identification number on the static load entries, usually field 2, is specified in the DAREA field of the TLOADi entry.
4. The initial temperature condition is requested by the IC command which selects TEMP or TEMPD entries. The IC command must be specified above the subcase level.
5. Restarts are controlled by the PARAMeters STIME, LOOPID, and SLOOPID.
6. See the *MSC.Nastran Thermal Analysis User's Guide* for further details.

Cyclic Symmetry Statics (SOLution 114)

1. Each loading condition must have a separate subcase.
2. All constraints must be specified above the subcase level.
3. A single set of direct input matrices may be selected with K2GG and/or M2GG above the subcase level. P2G is allowed in SOLution 114.
4. See also “[Cyclic Symmetry](#)” on page 803.

Cyclic Symmetry Modes (SOLution 115)

1. METHOD must be included above the subcase level to select an EIGR or EIGRL entry that exists in the Bulk Data Section.
2. All constraints must be specified above the subcase level.
3. A single set of direct input matrices may be selected with K2GG and/or M2GG above the subcase level.
4. See also “[Cyclic Symmetry](#)” on page 803.

Cyclic Symmetry Buckling (SOLution 116)

1. Case Control requirements for cyclic symmetry are the same as in SOLution 105. However, only one static and buckling subcase may be specified.

2. For the static subcase requirements see the description under “**Cyclic Symmetry Statics (SOLution 114)**” on page 21.
3. See also “**Cyclic Symmetry**” on page 803.

Design Sensitivity Analysis and Optimization (SOLution 200)

1. SOLution 200 has the same case control requirements as the solution sequences identified on the ANALYSIS Case Control command for SOL 200. The ANALYSIS command is specified in each subcase to identify the analysis to which the subcase is to be applied. If superelements are present, then the ANALYSIS command need only be specified in the residual structure subcases.
2. **Table 1-2** indicates the analysis types that are available in SOL 200 and whether the types support multiple subcases and multiple boundary conditions.

Table 1-2 Analysis disciplines supported in Multidisciplinary Analysis and Design of SOL 200

Analysis	SOL* Numbers	Multiple Subcases	Multiple Boundary Conditions
Statics	101	Y	Y
Normal Modes	103	Y	Y
Buckling	105	Y	Y
Direct Complex Eigenanalysis	107	N	N
Direct Frequency	108	Y	N
Modal Complex Eigenanalysis	110	N	N
Modal Frequency	111	Y	N
Modal Transient	112	N	N
Static Aeroelasticity	144	Y	Y
Flutter	145	Y	Y

*The column of Solution Numbers refers to the number MSC.Nastran would use to perform the separate analyses.

3. The DESGLB and DESSUB commands select design constraints specified on DCONSTR and DCONADD Bulk Data entries. The DESGLB selection is independent of subcase and the DESSUB applies only to the current subcase.
4. The DESOBJ command selects a scalar design objective specified on a DRESP1, DRESP2 or DRESP3 Bulk Data entry.
5. The optional DESVAR command selects the set of design variables to be used. If this command is not specified, all DESVAR Bulk Data entries are used.
6. The MODTRAK command selects mode tracking parameters.
7. The AUXCASE and AUXMODEL indicate subcases to be applied to auxiliary models.
8. The DSAPRT command requests the output of design sensitivity coefficients.
9. See the *MSC.Nastran Design Sensitivity and Optimization User's Guide* for further information.

Nonlinear Static Analysis (SOLution 106)

1. A separate subcase must be defined for each unique combination of static loads (LOAD and TEMPERATURE(LOAD)), constraints (SPC and MPC), and nonlinear iteration strategy (NLPARM and NLPCI).
2. A static loading condition may be defined for each subcase with a LOAD and/or TEMPERATURE(LOAD) selection unless all loading is specified through enforced displacements on SPC entries. If unloading or creep under residual load is desired, then no static load is specified in the subcase.
3. If thermal loads are present, then TEMPERATURE(INIT) must be specified above the subcase level.
4. An SPC command must be specified unless the model is constrained on the GRID entries.
5. A single set of direct input matrices may be selected with K2GG, M2GG, and P2G.
6. If upstream superelements have static loads, then LOADSET must be used to select LSEQ entries and, in the residual structure subcases only, the CLOAD command must be used to assemble the upstream loads.
7. Restarts are controlled by PARAMeters LOOPID and SUBID or SUBCASID. See "[Parameters](#)" on page 601 of the *MSC.Nastran Quick Reference Guide*.

8. The normal, complex, or unsymmetric modes due to the total nonlinear stiffness may be computed at the end of each subcase as long as PARAM,NMLOOP,<>0 and a METHOD, CMETHOD, or UMETHOD command is specified in the desired subcase and selects EIGB, EIGR, EIGRL, EIGC, or Bulk Data UEIG entry.
9. The buckling modes due to the total nonlinear stiffness may be computed at the end of each subcase as long as PARAM,BUCKLE,>0 and a METHOD command is specified in the desired subcase and selects an EIGB, EIGR, or EIGRL Bulk Data entry.
10. See “” on page 568 for further information.

Nonlinear Transient Response Analysis (SOLution 129)

1. A subcase must be defined for each unique combination of dynamic loads (DLOAD) and nonlinear iteration strategy (TSTEPNL).
2. Only one set of constraints (SPC and MPC) may be selected and must be specified above the subcase level.
3. LOADSET may be used to select LSEQ entries which reference static load sets to be used in the generation of dynamic loads.
4. Restarts are controlled by PARAMeters STIME, LOOPID, and SLOOPID. See “[Parameters](#)” on page 601 of the *MSC.Nastran Quick Reference Guide*.
5. Upstream superelements may be reduced dynamically using the METHOD commands.
6. See “” on page 568 for further information.

Superelement Subcase Structure

Subcases must be defined for each superelement to specify load, constraint, and output selection. The syntax rules described above also apply for superelements, i.e., requests above the subcase level are overridden by requests inside the subcase level. See Superelement Case Control in this section for exceptions to the above examples. Static analysis with multiple loads, two superelements.

```

DISP = ALL
ESE = ALL
SPC = 1000
SUBCASE 11
LOAD = 1000
SUBCASE 12
LOAD = 1002

```

For most applications, a conventional subcase structure may be used in superelement analysis. The subcase command SEALL = ALL and SUPER = ALL are the default in superelement solution sequences. SEALL = ALL requests the program to run to completion. SUPER = ALL instructs the program to use the same subcases for all superelements. See “[Restart Procedures](#)” on page 372, for use of commands alternative to the SEALL command for restarts and for applications where it is necessary to use more than one SUPER command.

The specification of the METHOD command in a superelement subcase requests the calculation of component modes.

In this example, the same output, constraint, and load requests are used for all superelements and the residual structure. Two loading conditions are analyzed.

Case Control Commands for Output

Printed output may be requested in either a SORT1 or SORT2 format. The SORT1 format is a tabular listing of the requested output, sorted by grid point number or element number, for each requested loading condition, forcing frequency or time step. The SORT2 format is a tabular listing of the requested output, sorted by loading condition, forcing frequency or time step, for each requested grid point or element.

In any one run, the output requests must be all SORT1 or all SORT2. If both types of output format are desired, a restart must be made to satisfy the second format request. Any request for deformed structure plots results in a request for the SORT1 format. Any X-Y output request results in a request for the SORT2 format.

The default for all analyses except transient response is the SORT1 format. The default for transient response is the SORT2 format.

In the case of the SORT1 format for static or transient analysis, there is an additional option to sort the stresses on the basis of magnitude and to print only the largest stresses. See the description of “[DTI,INDTA](#)” on page 1308 of the , for details on preparing the user request.

This section describes the output options for the various solution sequences. MSC.Nastran is designed to allow the solution of large problems, which can generate very large amounts of output. For this reason, most types of output (e.g., stresses in static analysis, or eigenvectors in modal analysis) are not automatically printed, but must be specifically requested by the user. Options are provided which allow the user to restrict the printed output to only significant results. See Parameter Descriptions for additional output options.

Static Analysis (SOLutions 101, 106, 114, 144, and 200)

1. Displacements at selected grid points (DISPLACEMENT).
2. Nonzero components of the applied static loads at selected grid points (OLOAD).
3. Nonzero components of single point forces of constraint, including reactions on free-body supports due to applied loads, at selected grid points or scalar points (SPCF).
4. Forces and stresses in selected elements (FORCE and STRESS). In SOLution 144, forces and pressures on aerodynamic elements (APRESSURE and AEROF).
5. In all SOLutions except 4 and 106, grid point force balance at selected grid points (GPFORCE).
6. In all SOLutions except 4 and 106, strain energy in selected elements (ESE).
7. Undeformed and deformed plots of the structural model (geometry for fundamental region in cyclic symmetry) (OUTPUT(PLOT)).
8. Multipoint forces of constraint for MPCs and rigid elements (MPCF).
9. Contour plots for stresses in selected elements and displacements at selected grid points (geometry for fundamental region in cyclic symmetry) (OUTPUT(PLOT)).
10. Grid point stresses and strains in selected surfaces or volumes are available (GPSTRESS and GPSTRAIN).
11. Nonlinear stress output is controlled by NLSTRESS. Its default is ALL.

Normal Modes (SOLutions 103, 115, and 200)

1. The eigenvector for a list of selected GRID points (g-set) or SOLUTION points (a-set) along with the associated eigenvalue for each mode (VECTOR or SVECTOR).
2. Nonzero components of the single-point forces of constraint for selected modes at selected grid points (SPCF).
3. Forces, stresses, and strains in selected elements for selected modes (FORCE, STRESS, and STRAIN).
4. Grid point force balance at selected grid points (GPFORCE).
5. Strain energy and kinetic energy in selected elements (ESE, EKE).

6. Undeformed plot of the structural model and mode shapes for selected modes (geometry for fundamental region in cyclic symmetry) (OUTPUT(PLOT)).
7. Grid point stresses and strains in selected surfaces or volumes (GPSTRESS and GPSTRAIN).
8. The application of response spectrum curves may be performed in SOLutions 103 and 115. See “[Response Spectrum Analysis](#)” on page 886.
9. Multipoint forces of constraint for MPCs and rigid elements (MPCF).

Buckling Analysis (SOLutions 105 and 200)

1. Displacements and nonzero components of the static loads and single-point forces of constraint at selected grid points for the static analysis (DISPLACEMENT, OLOAD and SPCF).
2. Forces and stresses in selected elements for the static loading condition and selected buckling modes (FORCE and STRESS).
3. Mode shapes and nonzero components of the single-point forces of constraint at selected grid points for selected modes (VECTOR and SPCF).
4. Undeformed plot of the structural model and mode shapes for selected buckling modes (OUTPUT(PLOT)).
5. Grid point force balance at selected grid points for the static loading condition (GPFORCE).
6. Strain energy in selected elements for the static loading condition (ESE).
7. Grid point stresses and strains in selected surfaces or volumes for the static loading condition (GPSTRESS and GPSTRAIN).
8. Multipoint forces of constraint for MPCs and rigid elements (MPCF).

Complex Eigenvalue Analysis (SOLutions 107, 110, 145, and 200)

The following printed output, sorted by complex eigenvalue root number (SORT1), may be requested for any complex eigenvalue extracted, as either real and imaginary parts or magnitude and phase angle ($0^\circ - 360^\circ$ lead):

1. The eigenvector for a list of PHYSICAL points (grid points and scalar points) or SOLUTION points (points used in formulation of the general K system) (VECTOR or SVECTOR).
2. Nonzero components of the single-point forces of constraint for a list of PHYSICAL points (SPCF).

3. Forces, stresses, or strains in selected elements (FORCE and STRESS STRAIN).
4. In addition, an undeformed plot of the structural model may be requested. (OUTPUT(PLOT))
5. Multipoint forces or constraint for MPCs and rigid elements (MPCF).
6. Strain energy, kinetic energy, and energy loss (ESE, EKE, EDE).

Frequency, Random, and Aeroelastic Response (SOLutions 108, 111, 118, 146, and 200)

The following printed output, sorted by frequency (SORT1) or by point number or element number (SORT2), is available, either as real and imaginary parts or magnitude and phase angle ($0^\circ - 360^\circ$), for the list of frequencies specified by OFREQUENCY:

1. Displacements, velocities, and accelerations for a list of PHYSICAL points (grid points and scalar points) or SOLUTION points (points used in formulation of the generalized coordinate system) (DISPLACEMENT or SDISPLACEMENT, VELOCITY or SVELOCITY, ACCELERATION or SACCELERATION).
2. Nonzero components of the applied load vector and single-point forces of constraint for a list of PHYSICAL points (OLOAD and SPCF).
3. Stresses and forces in selected elements (ALL available only for SORT1) (STRESS and FORCE). In SOLution 146, forces on aerodynamic elements. (AEROF).
4. The following plotter output is available for frequency response calculations:
 - Undeformed and deformed structure plots of the structural model (OUTPUT(PLOT)).
 - X-Y plot of any component of displacement, velocity, or acceleration of a PHYSICAL point or SOLUTION point (OUTPUT(XYPLOT)).
 - X-Y plot of any component of the applied load vector of single-point force of constraint (OUTPUT(XYPLOT)).
 - X-Y plot of any stress or force component for an element (OUTPUT(XYPLOT)).

5. The following plotter output is available for random response calculations: The data used for preparing the X-Y plots may be punched or printed in tabular form. Also, a printed summary is prepared for each X-Y plot which includes the maximum and minimum values of the plotted function. The printing of the X-Y pairs is the only form of printed output for Random Response. SORT2 output is required for X-Y plots. If both SORT1 and SORT2 requests are made, only the SORT2 output will be printed.
 - X-Y plot of the power spectral density versus frequency for the response of selected components for points or elements (OUTPUT(XYPLOT)).
 - X-Y plot of the autocorrelation versus time lag for the response of selected components for points or elements (OUTPUT(XYPLOT)).
6. Multipoint forces of constraint for MPCs and rigid elements (MPCF).
7. Strain energy, kinetic energy, and energy loss (ESE, EKE, EDE).

Transient Response (SOLutions 109, 112, 129, and 200)

The following printed output, sorted by point number or element number is available at selected multiples of the integration time step:

1. Displacements, velocities, and accelerations for a list of PHYSICAL points (grid points and scalar points) or SOLUTION points (points used in formulation of the general K system) (DISPLACEMENT or SDISPLACEMENT, VELOCITY or SVELOCITY, ACCELERATION or SACCELERATION).
2. Nonzero components of the applied load vector and single-point forces of constraint for a list of PHYSICAL points (OLOAD and SPCF).
3. Nonlinear force vector for a list of SOLUTION points (NONLINEAR).
4. Stresses and forces in selected elements (STRESS and FORCE). In SOLutions 109 and 129, forces are not computed for nonlinear elements.
5. Grid point stresses in selected surfaces or volumes (GPSTRESS).
6. The following plotter output is available. The data used for preparing the X-Y plots may be punched or printed in tabular form. Also, a printed summary is prepared for each X-Y plot which includes the maximum and minimum values of the plotted function.
 - Undeformed plot of the structural model (OUTPUT(PLOT)).
 - Deformed shapes of the structural model for selected time intervals (OUTPUT(PLOT)).

- Contour plots for stresses in selected elements, and displacements, velocities or accelerations at selected grid points for selected time intervals (OUTPUT(PLOT)).
 - X-Y plot of any component of displacement, velocity, or acceleration of a PHYSICAL point or SOLUTION point (OUTPUT(XYPLOT)).
 - X-Y plot of any component of the applied load vector, nonlinear force vector, or single-point force of constraint (OUTPUT(XYPLOT)).
 - X-Y plot of any stress or force component for an element (OUTPUT(XYPLOT)).
7. Multipoint forces of constraint for MPCs and rigid elements except SOLution 129.

Linear Steady State Heat Transfer (SOLutions 101 and 153)

1. Temperatures (THERMAL) and nonzero components of static loads (OLOAD) and constrained heat flow (SPCFORCE) at selected grid points or scalar points.
2. The punch option of a THERMAL request to produce TEMP Bulk Data entries.
3. Flux density (FLUX) in selected elements.
4. Grid point heat balance (GPFORCE) at a selected set of grid points or scalar points (SOLution 101).
5. Undeformed plots of the structural model and temperature profiles.
6. Contour plots for temperatures at selected grid points.

Steady State Heat Transfer (SOLution 153)

The following output may be requested for the last iteration in Heat Transfer Analysis:

1. Temperature (THERMAL) and nonzero components of static loads (OLOAD) and constrained heat flow (SPCFORCE) at selected grid points or scalar points.
2. The punch option of a THERMAL request to produce TEMP Bulk Data entries.
3. Flux density (FLUX) in selected elements. In the case of CHBDYi elements, a flux density summary is produced that includes applied flux, radiation flux, and convective flux.
4. Undeformed plots of the structural model and temperature profiles.

5. Contour plots of temperatures at selected grid points.
6. See the *MSC.Nastran Thermal Analysis User's Guide* for further details.

Nonlinear Transient Heat Transfer (SOLution 159)

The following printed output, is available at selected multiples of the integration time step:

1. Temperatures (THERMAL or SDISPLACEMENT) and derivatives of temperatures (VELOCITY or SVELOCITY) for a list of PHYSICAL points (grid points and scalar points) or SOLUTION points (points used in formulation of dynamic equation).
2. Nonzero components of the applied load vector (OLOAD) and constrained heat flow (SPCFORCE) for a list of PHYSICAL points.
3. Nonlinear load vector (NLLOAD) for a list of SOLUTION points.
4. Flux density (FLUX) in selected elements (ALL not allowed).
5. Rate of change of enthalpy (HDOT).

The following plotter output is available for Transient Heat Transfer Analysis:

1. Undeformed plot of the structural model.
2. Temperature profiles for selected time intervals.
3. X-Y plot of temperature (TEMP or SDISP) or derivative of temperature (VELO or SVELO) for PHYSICAL or SOLUTION points.
4. X-Y plot of the applied load vector (OLOAD), nonlinear load vector (NOLI), or constrained heat flow (SPCF).
5. X-Y plot of flux density (FLUX) for an element.
6. Contour plots of temperatures at selected grid points.

The data used for preparing the X-Y plots may be punched or printed in tabular form. Also, a printed summary is prepared for each X-Y plot which includes the maximum and minimum values of the plotted function.

Case Control Looping

Boundary Condition Changes

More than one set of MPC, SPC, and SUPORT1 selections is allowed in all of the subcases for the residual structure during one run in SOLutions 101, 103, 105, 106, 108, 111 and 200. Only one boundary condition is allowed for superelements.

Frequency and Dynamic Load Set Changes in SOLutions 108, 111, 118, 146 and 200

More than one FREQUENCY Case Control command may be used per run except in SOLutions 146 and 200. Residual structure subcases may be used to select different dynamic loads using the DLOAD command. Output requests may differ between subcases.

Transient and Dynamic Load Set Changes in SOLutions 109, 112 and 200

More than one TSTEP and DLOAD Case Control command may be used per run except in SOLution 200. Only one residual structure subcase is allowed.

Formulation of Dynamic Equations in All Dynamic Response SOLutions

Multiple subcases with different requests of K2PP, M2PP, B2PP, TFL, and SDAMPING Case Control commands may be specified in SOLs 107 through 112, 129 and 145.

DMIG Bulk Data Entry Processing

Two forms of direct input matrix processing are provided. The Case Control commands B2GG, M2GG, K2GG, and P2G select matrices added to the global matrices before constraints are applied and only one set may be selected for superelement or residual structures. These terms may be added to any superelement or the residual structure. K2GG, B2GG and M2GG must be real and symmetric (IF = 6). Their precision should be set equal to the precision of [K2GG] (TOUT = 0 will provide this).

The Case Control commands B2PP, M2PP and K2PP select matrices processed only in the residual structure. In the modal formulations they are added after real eigenvalue extraction. These matrices need not be symmetric.

REPCASE, SUBCOM and SUBSEQ Case Control Commands

These commands may be used for superelement processing, with the provision that the SUPER Case Control command must be used with REPCASE and SUBCOM. An example of a statics subcase structure follows:

```
SUBCASE 10
SUPER = 10,1
LOAD = 1
SET 1 = 1 THRU 10
DISP = 1
REPCASE 11
SUPER = 10,1
SET 2 = 11 THRU 100
DISP = 2
REPCASE 12
SUPER = 10,1
SET 3 = 101 THRU 200
DISP = 3
SUBCASE 20
SUPER = 10,2
...
```

The REPCASEs must immediately follow the subcase they reference. They should contain the same SUPER = i,j command. In this example, the two REPCASEs give output for the loading condition described in SUBCASE 10.

1.7 Bulk Data Section

The primary MSC.Nastran input medium is the Bulk Data entry. These entries are used to define the model and data which may be selected in the Case Control Section at execution time. For large problems the Bulk Data may consist of several thousand entries.

The Bulk Data may be submitted with the entries in any order since a sort (see “**XSORT**” on page 1730 of the) is performed prior to the execution of the Input File Processor (see “**IFP**” on page 1189 of the *MSC.Nastran 2004 DMAP Programmer’s Guide*). It should be noted that the machine time to perform this is minimized for input that is already sorted. The sort time for badly sorted data will become significant for large numbers of entries. The user may obtain a printed copy of either the unsorted or sorted Bulk Data by selection in the Case Control Section. A sorted echo is automatically provided unless specifically suppressed by the user.

Starting in MSC.Nastran 2001, setting SYSTEM CELL 357 to 1 will tell MSC.Nastran to ignore the continuation field, and hence eliminating the concern of duplicate continuation identifiers.

In order to minimize the handling of a large number of entries, provision has been made in the Structured Solution Sequences (see “**Solution Sequences**” on page 341) to store the Bulk Data in the database, and it may be modified on subsequent runs. For any cold start, the entire bulk data must be submitted. Thereafter, if the database from any previous run is saved, the Bulk Data exists in the database in sorted form where it may be modified and reused on restart. Upon restart the Bulk Data entries contained in the Bulk Data are added to the Bulk Data stored in the database. Entries are removed from the database by using a “/” entry.

See the “**Bulk Data Entries**” on page 849 of the *MSC.Nastran Quick Reference Guide*, for a detailed description of the Bulk Data.

See the *MSGMESH Analyst’s Guide* for information on the use of the MSGMESH program for the automatic generation of Bulk Data entries.

1.8 Use of Parameters

Parameters are used extensively in the solution sequences for input of scalar values and for requesting special features. Parameters' values are specified on PARAM Bulk Data entries or PARAM Case Control commands. The PARAM Bulk Data entry is described in the “[Bulk Data Entries](#)” on page 849 of the *MSC.Nastran Quick Reference Guide*. The PARAM Case Control command is described in the “[Case Control Commands](#)” on page 175 of the *MSC.Nastran Quick Reference Guide*.

The manner in which MSC.Nastran processes PARAMeters is described in general terms in the *MSC.Nastran 2005 DMAP Programmer's Guide*. The specification of PARAMeters differs between the rigid formats and the superelement solution sequences and SOL 4. These differences require different user action in placing the PARAM entries in the MSC.Nastran section, and in changing PARAMeter values on restarts.

1.8.1 Parameters in the Superelement Solution Sequences

The program module reads the Case Control Section for PARAM entries. If PARAM entries exist, the VPS table is updated with the values found in the Case Control Section. The PARAM values are updated at the beginning of the generation, assembly, reduction, and the data recovery loops of the superelement solution sequences. This allows the values of some user-supplied parameters to change for each superelement during the course of one run. In the nonlinear solution sequences the PARAM values are updated at the beginning of the iteration loop. This allows the values of some user-set parameters to vary between subcases. These parameters may appear in either the Case Control and/or Bulk Data Section. Other PARAMeters must appear in the Bulk Data Section. If they appear in the Case Control Section, they will not cause a preface error, but will be ignored. Other parameters may be reset to default values during the course of the run. This type should be set only in the subcase of the Case Control Section. See [Table 5-1](#) and [Table 5-2](#) in the , for parameter applicability.

The PARAM Case Control command is treated in the same manner as constraint and output commands. Parameters set above the subcase level are effective in every subcase. If there is also a parameter value set in the subcase, it takes precedence. Only one value of a parameter may be current at any time during the execution of the run. These rules imply an order of precedence if several forms are used in the same run. The rules are illustrated in the following example.

A partial subcase structure is listed on the following page. The subcases are listed in ascending superelement number order. PARAM,GRDPNT is varied between them.

Subcase Structure, Sorted by SEID

```

SUBCASE 1
SUPER = 1
...
SUBCASE 2
SUPER = 2
PARAM,GRDPNT,100
...
SUBCASE 3
SUPER = 3
PARAM,GRDPNT,200
...
SUBCASE 4
SUPER = 4
...
SUBCASE 100 $ RESIDUAL STRUCTURE
PARAM,GRDPNT,-1
...
BEGIN BULK
PARAM,GRDPNT,0

```

Processing Order	Value of GRDPNT
SEID	
1	0
2	100
3	200
4	0
0	-1

The Bulk Data entries are processed at the beginning of the run. Any PARAM entries in the Bulk Data Section set the value of that parameter until it is changed by a Case Control command. Superelement 1 is processed first. Since it has no PARAM,GRDPNT entry the value of 0, as set by the Bulk Data entry PARAM,GRDPNT,0 is used. The value for superelement 2 is set at 100 by a Case Control command. Superelement 4 is processed next. Its subcase contains no PARAM entry, so that the value of the parameter that remains is set by the bulk data. The values for superelement 3 and the residual structure are set by Case Control commands. An alternate method of parameter specification is to set the value most used above the subcase level, and exceptional values within the subcase. For example,

if PARAM,GRDPNT,0 were placed above the subcase level and the values of 100, 200, and -1 placed in subcases 2, 3 and 100, respectively, the value used in each superelement will be the same as described above, regardless of processing order.

CHAPTER

2

Grid Points and Coordinate Systems

- Introduction
- Grid Point and Coordinate System Definition
- Grid Point Properties

2.1 Introduction

Analysis model geometry in MSC.Nastran is represented by grid point locations. The physical behavior of the analysis model is represented by the constraints and their associated coordinate system, and by the elements connecting the grid points (elements are described in “[Structural Elements](#)” on page 47).

2.2 Grid Point and Coordinate System Definition

Geometric grid points are defined on GRID Bulk Data entries by specifying coordinates in either the basic coordinate system or in a local coordinate system. The implicitly defined basic coordinate system is rectangular, except when using axisymmetric elements. Local coordinate systems may be rectangular, cylindrical, or spherical and are related directly or indirectly to the basic coordinate system.

Figure 2-1 shows the coordinate system definition where:

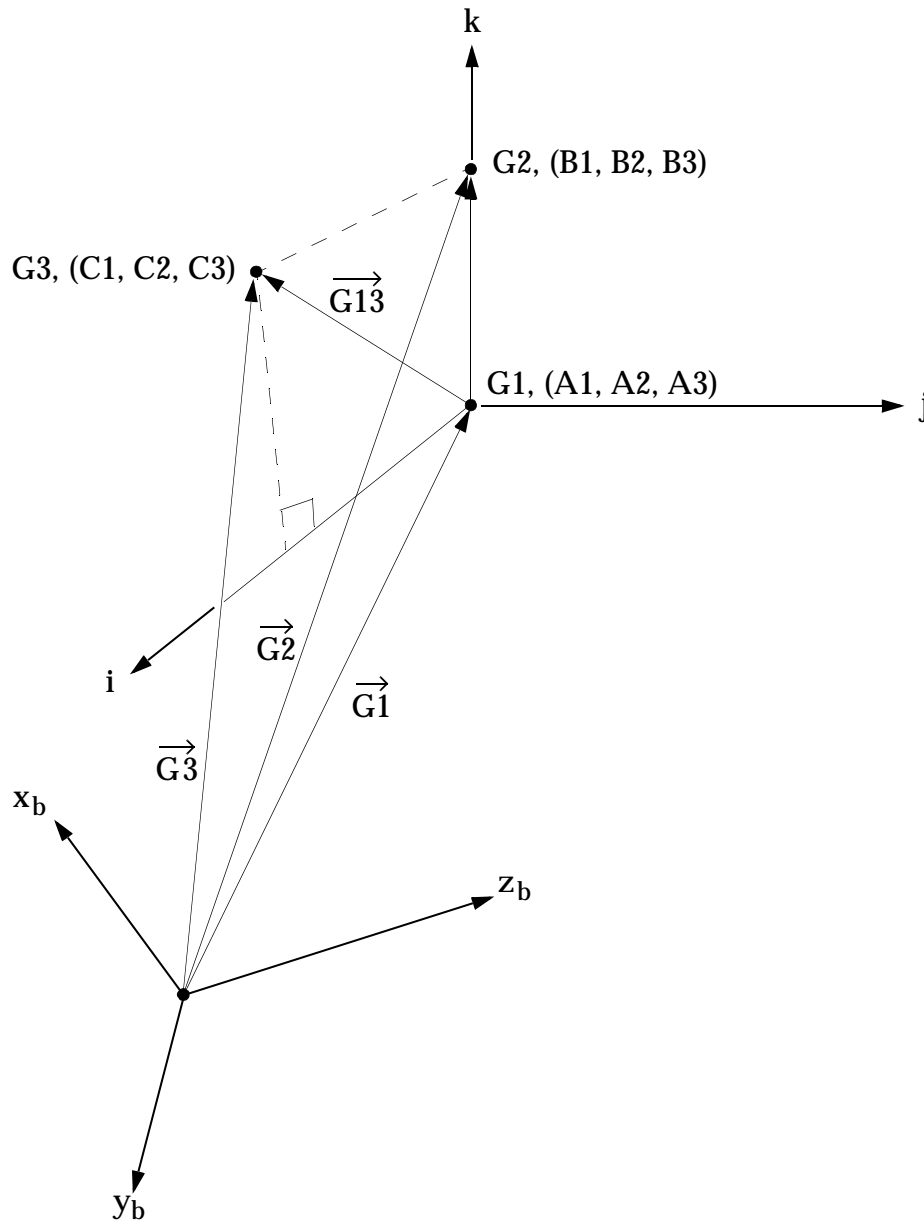


Figure 2-1 Coordinate System Definition

where:

(i, j, k) = user defined coordinate system (x, y, z) or (r, θ, z) or (r, θ, ϕ)

(x_b, y_b, z_b) = basic coordinate system

$$\vec{k} = (\vec{G2} - \vec{G1}) \text{ normalized}$$

$$\vec{G13} = (\vec{G3} - \vec{G1})$$

$$\vec{j} = (\vec{k} \times \vec{G13}) \text{ normalized}$$

$$\vec{i} = (\vec{j} \times \vec{k})$$

The CORD1C, CORD1R and CORD1S entries are used to define cylindrical, rectangular, and spherical local coordinate systems, respectively, in terms of three previously defined geometric grid points. The CORD2C, CORD2R and CORD2S entries are used to define cylindrical, rectangular, and spherical local coordinate systems, respectively, in terms of the coordinates of three points in a previously defined coordinate system.

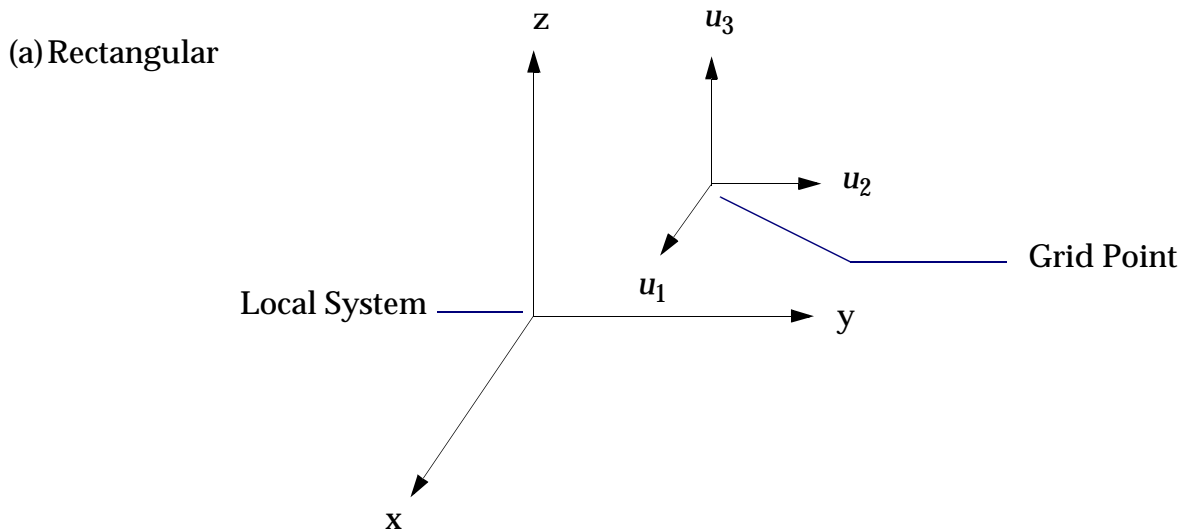
Six rectangular displacement components (three translations and three rotations) are defined at each grid point. The “displacement coordinate system,” which is used to define the directions of motion, may be different from the “location coordinate system,” which is used to locate the grid point. Both the location coordinate system and the displacement coordinate system are specified on the GRID entry for each geometric grid point. The orientation of displacement components depends on the type of local coordinate system used to define the displacement components. If the defining local system is rectangular, the displacement components are parallel to the local system and are independent of the grid point location as indicated in [Figure 2-2\(a\)](#). If the local system is cylindrical, the displacement components are in the radial, tangential, and axial directions as indicated in [Figure 2-2\(b\)](#). If the local system is spherical, the displacement components are in the radial, meridional, and azimuthal directions as indicated in [Figure 2-2\(c\)](#).

If the displacements of a grid point are defined to be in a spherical or cylindrical coordinate system and the grid point is located on, or relatively close to, the polar axis, then it will be replaced by the “defining” rectangular coordinate system. The z-axis of the “defining” rectangular coordinate system is coincident with the polar axis and the x-axis is coincident with $\theta = 0$ in cylindrical system, or $\theta = 0$ and $\phi = 0$ in the spherical system. It is recommended that a rectangular system is explicitly specified by the user.

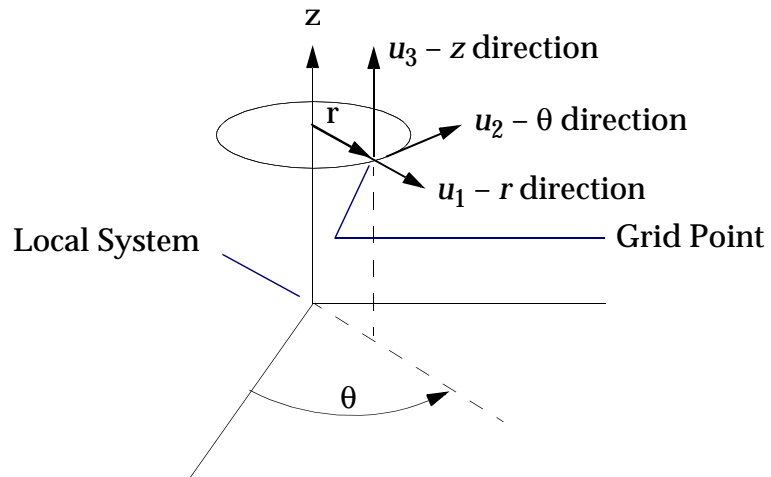
Each geometric grid point may have a unique displacement coordinate system associated with it. The collection of all displacement coordinate systems is known as the “global coordinate system.” All matrices are formed and all displacements are output in the global coordinate system. The symbols T1, T2 and T3 on the printed output indicate translations in the 1, 2, and 3-directions, respectively, for each grid point. The symbols R1, R2, and R3 indicate rotations about the three axes.

Provision is also made on the GRID entry to apply single-point constraints to any of the displacement components. Any constraints specified on the GRID entry will automatically be used for all solutions. Constraints specified on the GRID entry are usually restricted to those degrees-of-freedom that will not be elastically constrained and hence must be removed from the model in order to avoid singularities in the stiffness matrix.

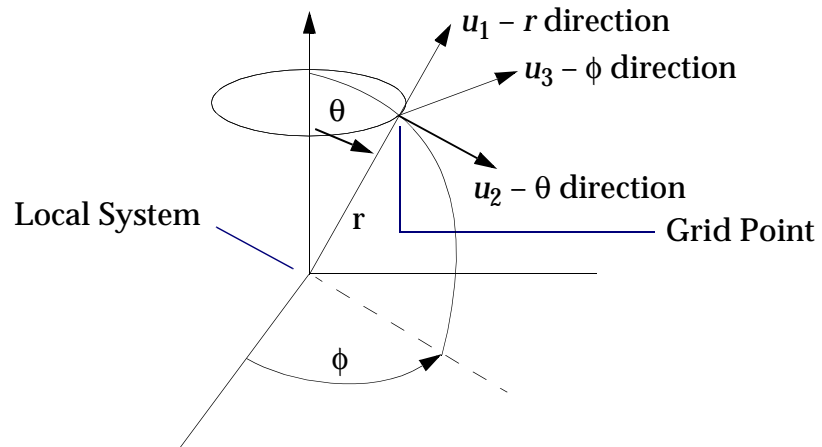
Element offsets are defined by a Cartesian coordinate system, which is also parallel to the displacement coordinate system at the connecting grid point. However, the components of the offsets are always defined in units of translation, even if the displacement coordinate system is cylindrical or spherical.



(b) Cylindrical



(c) Spherical

**Figure 2-2 Displacement Components**

The GRDSET Bulk Data entry is provided to avoid repeating the specification of location coordinate systems, displacement coordinate systems, and single point constraints when many (or all) of the GRID entries have the same entries for these items. When any of the three items are specified on the GRDSET entry, the entries are used to replace blank fields on the GRID entry for these items. This feature is useful in the case of such problems as space trusses where one wishes to remove all of the rotational degrees-of-freedom or in the case of plane structures where one wishes to remove all out-of-plane or all in-plane motions.

Scalar points are defined either on an SPOINT entry or by reference on a connection entry for a scalar element. SPOINT entries are used primarily to define scalar points appearing in constraint equations but to which no structural elements are connected. A scalar point is implicitly defined if it is used as a connection point for any scalar

element. Special scalar points, called “extra points,” may be introduced for dynamic analyses. Extra points are used in connection with transfer functions and other forms of direct matrix input used in dynamic analyses and are defined on EPOINT entries.

GMCORD defines a convective/follower coordinate system on a FEEDGE, GMCURV, FEFACE, or GMSURF Bulk Data entry for use in adaptive (p-element) analysis.

2.3 Grid Point Properties

Some of the characteristics of the model may be specified as properties of grid points rather than as properties of elements.

Mass, damping, and stiffness properties may be provided, in part or entirely, as properties of grid points through the use of direct input matrices. The DMIG entry is used to define direct input matrices. These matrices may be associated with components of geometric grid points, scalar points or extra points introduced for dynamic analysis. The TF entry is used to define transfer functions that are internally converted to direct matrix input. The DMIAX entry is an alternate form of direct matrix input that is used for hydroelastic problems (see “[Coupled Fluid-Structure Interaction](#)” on page 753).

Thermal fields are defined by specifying the temperatures at grid points. The TEMP entry is used to specify the temperature at grid points for use in connection with thermal loading and temperature-dependent material properties. The TEMPD entry is used to specify a default temperature in order to avoid a large number of duplicate entries on a TEMP entry when the temperature is uniform over a large portion of the structure. The TEMPAX entry is used for conical shell problems.

Mass properties may be input as properties of grid points by using the concentrated mass element. The CONM1 entry is used to define a 6×6 matrix of mass coefficients at a geometric grid point in any selected coordinate system. The CONM2 entry is used to define a concentrated mass at a geometric grid point in terms of its mass, the three coordinates of its center of gravity, the three moments of inertia about its center of gravity, and its three products of inertia, referred to any selected coordinate system.

The SNORM Bulk Data entry and/or the PARAM,SNORM assigns the singularity normal of a shell element to a degree-of-freedom. This singularity is then removed via AUTOSPC.

CHAPTER
3

Structural Elements

- Introduction
- Line Elements
- Surface Elements
- Solid Elements (CTETRA, CPENTA, CHEXA)
- Rigid Elements and Multipoint Constraints (R-type, MPC)
- Scalar Elements (CELASi, CMASSi, CDAMPi)
- Axisymmetric Solid Elements (CTRIAX6, CTRIAX, CQUADX)
- Crack Tip Elements (CRAC2D, CRAC3D)
- Gap and Line Contact Elements (CGAP, BCONP, BLSEG)
- Concentrated Mass Elements (CONM1, CONM2)
- General Element Capability (GENEL)
- Direct Matrix Input on Grid Points (DMIG)
- p-Elements
- Hyperelastic Elements
- Bushing Elements
- Interface Elements
- CWELD Element Enhancements
- Element Geometry Checks
- Nonstructural Mass
- ELSUM Case Control Command

3.1 Introduction

Element and Property Definition

Structural elements are defined on Bulk Data connection entries that identify the grid points to which the element is connected. The mnemonics for all such entries have a prefix of the letter “C”, followed by an indication of the type of element, such as CBAR and CROD. The order of the grid point identification defines the positive direction of the axis of a one-dimensional element and the positive surface of a plate element. The connection entries include additional orientation information when required. Some elements allow for offsets between its connecting grid points and the reference plane of the element. The coordinate systems associated with element offsets are defined in terms of the grid point coordinate systems described in “[Grid Points and Coordinate Systems](#)” on page 39. For most elements, each connection entry references a property definition entry. If many elements have the same properties, this system of referencing eliminates a large number of duplicate entries.

The property definition Bulk Data entries define geometric properties such as thicknesses, cross-sectional areas, and moments of inertia. The mnemonics for all such entries have a prefix of the letter “P”, followed by some or all of the characters used on the associated connection entry, such as PBAR and PROD. Other included items are the nonstructural mass and the location of points where stresses will be calculated. For most elements, each property definition entry will reference a material property entry.

In some cases, the same finite element can be defined by using different Bulk Data entries. These alternate entries have been provided for user convenience. In the case of a rod element, the normal definition is accomplished with a connection entry (CROD) which references a property entry (PROD). However, an alternate definition uses a CONROD entry which combines connection and property information on a single entry. This is more convenient if a large number of rod elements all have different properties.

Most of the elements may be used with elements of other types within the limitations of good modeling practice. Exceptions are the axisymmetric elements, which are designed to be used by themselves. There are two types at present, linear and nonlinear. The linear are described in the following section. The nonlinear are described under “[Hyperelastic Elements](#)” on page 221. The conical shell element (“CCONEAX”) describes a thin shell by sweeping a line defined on a plane by two end points through a circular arc. Loads may be varied with azimuth angle through use of

harmonic analysis techniques. This element has a unique set of input entries, which may be used with a limited set of other entries. These unique entries, if mixed with entries for other types of elements, cause a preface error.

Axisymmetric Elements

The axisymmetric elements CTRIAX6, CTRIAX, and CQUADX define a solid ring by sweeping a surface defined on a plane through a circular arc. Loads are constant with azimuth for these elements; that is, only the zeroth harmonic is considered. There may be innovative modeling techniques that allow coupling this class of axisymmetric element with other elements, but there are no features to provide correct automatic coupling.

Although the axisymmetric elements CCONEAX and CTRIAX6 provide useful functions, the user should be aware that their usage precludes the use of many features available for other elements (see [Table 3-1](#)). An alternative modeling technique is to use conventional elements with the cyclic symmetry solution sequences to provide equivalent models. This technique generally provides the features not available with axisymmetric elements. The CTRIAX and CQUADX axisymmetric elements are finite deformation, hyperelastic elements with fully nonlinear characteristics (see “[Hyperelastic Elements](#)” on page 221 and “” on page 568).

Material Properties

The material property definition entries are used to define the properties for each of the materials used in the structural model. The MAT1 entry is used to define the properties for isotropic materials and may be referenced by any of the structural elements. The MATT1 entry specifies table references for isotropic material properties that are temperature dependent. The TABLEM1, TABLEM2, TABLEM3, and TABLEM4 entries define four different types of tabular functions for use in generating temperature dependent material properties. More information is given in “[Material Properties](#)” on page 269.

The MAT2 entry is used to define the properties of anisotropic materials for triangular or quadrilateral membrane and bending elements. The MAT2 entry specifies the relationship between the in-plane stresses and strains. It may also be used for anisotropic transverse shear. The angle between the material coordinate system and the element coordinate system is specified on the connection entries. The MATT2 entry specifies table references for anisotropic material properties that are temperature dependent. This entry may reference any of the TABLEM1, TABLEM2, TABLEM3, or TABLEM4 entries.

The MAT3 entry is used to define the properties for orthotropic materials used in the modeling of axisymmetric shells. This entry may only be referenced by CTRLAX6 entries. The MATT3 entry specifies table references for use in generating temperature-dependent properties for this type of material. More information is given in “[Isotropic Material](#)” on page 272.

The MAT8 entry is used to define the properties of orthotropic materials used in the modeling of quadrilateral and triangular shell elements for composite structures.

The MAT9 entry is used to define the properties of anisotropic materials for the CHEXA, CPENTA, and CTETRA elements. The MATT9 entry specifies table references for use in generating temperature-dependent properties for this type of material. More information is given in “[Orthotropic Material](#)” on page 275.

The MATS1 entry specifies table references for isotropic material properties that are stress and/or temperature dependent. The TABLES1 entry defines a tabular stress-strain function for use in creep analysis (CREEP entry) or material nonlinear analysis. The TABLEST entry specifies TABLES1 references for nonlinear elastic temperature-dependent material properties. The MATS1 entry may be combined with a corresponding MAT1 to define the following types of nonlinear material analysis: (1) plastic (2) nonlinear elastic and (3) creep. More information is given in the *MSC.Nastran Handbook for Nonlinear Analysis*.

The MATHP entry specifies hyperelastic material properties and must be referenced by one of the PLPLANE or PLSOLID property entries, specifying a finite deformation nonlinear analysis, including large strain and large rotation. Experimental data may be input on TABLES1 entries for automatic curve fitting of material parameters (see “” on page 568).

In linear analysis, temperature-dependent material properties are computed once only at the beginning of the analysis. In nonlinear static analysis (SOLution 106), temperature-dependent material properties for linear (MAT1, MAT2, and MAT9 entries) and nonlinear elastic materials (MAT1 and MATS1 entries) may be updated many times during the analysis.

The General Element (GENEL)

The GENEL entry is used to define general elements whose properties are defined in terms of deflection influence coefficients or stiffness matrices which can be connected between any number of grid points. One of the important uses of the general element is the representation of part of a structure by means of experimentally measured data. No output data is prepared for the general element. Detailed information on the general element is given in “[General Element Capability \(GENEL\)](#)” on page 212.

Dummy, Mass, and Damping Elements

Dummy elements are provided in order to allow the user to investigate new structural elements with a minimum expenditure of time and money. A dummy element is defined with a CDUMi ($i = \text{index of element type, } 1 \leq i \leq 9$) entry and its properties are defined with the PDUMi entry. The ADUMi entry is used to define the fields on the connection and property entries.

Mass elements have inertia properties, but do not have stiffness or damping. The CONM2 entry provides convenient input of mass attached to a grid point whereas the CONM1 entry provides a more general description of mass at a grid point.

Damping elements have damping properties only. The CVISC entry provides damping along a line between two grid points.

Constraint Elements

There are two classes of constraint elements. They generate constraint equations that allow elimination of some variables, rather than add coefficients to the stiffness matrix. However, they do have other element-like properties, such as the satisfaction of equilibrium conditions. The RBAR, RBE1, RBE2, RROD, RJOINT, RSSCON, and RTRPLT entries describe rigid elements. The RBE3 and RSPLINE entries describe interpolation elements. (See also “[Rigid Elements and Multipoint Constraints \(R-type, MPC\)](#)” on page 168).

Other Element Types

Other types of elements are described in different subsections of “[Additional Topics](#)” on page 529. They include elements with fluid, acoustics, and nonlinear analysis capability. Heat transfer elements are described in the *[MSC.Nastran Thermal Analysis User's Guide](#)*.

Details of the finite elements in MSC.Nastran are given in the following sections. A summary of the finite elements and their characteristics is given in [Table 3-1](#). An X in the table indicates the existence of an item.

Element identification numbers must be unique across all element types.

Table 3-1 Element Summary -- Small Strain Elements

Element Type	Structural Matrices							Materials			Static Load				Heat Transfer			
	Stiffness	Mass	Differential Stiffness	Geometric Nonlinear	Material Nonlinear	Viscous Damping	Axisymmetric	p-Adaptivity	Isotropic	Anisotropic	Orthotropic	Thermal	Pressure	Gravity	Element Deformation	Heat Conduction	Heat Capacity	Thermal Load
CAXIFi							X											
CBAR	X	L C	X						X			EB	X	X	X	X	X	X
CBEAM	X	L C	X	X	X			X	X			EB	X	X	X	X	X	X
CBUSH	F D				X	F D												
CBUSH1D	X	X	X	X	X				X									
CBEND	X	C	X						X			EB	X	X		X	X	X
CCONEA X	X	L					X		X	X	X	E	X	X				
CONMi		L C												X				
CONROD	CS	L C	X	X	X				X			E		X	X	X	X	X
CRAC2D	I	L C							X	X		E		X				
CRAC3D	I	L C							X	X		E		X				
CDAMPi						X											X	
CELASi	X								X							X		
CFLUIDi							X											
CGAP	X				X													
CHBDYi															X	X	X	
CHEXA	I	L C	X	X*	X*			X	X	X		E	X	X	X	X	X	X
CMASSi		L												X				

Table 3-1 Element Summary -- Small Strain Elements (continued)

Element Type	Structural Matrices							Materials			Static Load				Heat Transfer			
	Stiffness	Mass	Differential Stiffness	Geometric Nonlinear	Material Nonlinear	Viscous Damping	Axisymmetric	p-Adaptivity	Isotropic	Anisotropic	Orthotropic	Thermal	Pressure	Gravity	Element Deformation	Heat Conduction	Heat Capacity	Thermal Load
CPENTA	I	L C	X	X*	X*			X	X	X		E	X	X		X	X	X
CQUAD4	I	L C	X	X	X			X	X	X	X	EB				X	X	X
CQUAD8	I	L C	X						X	X	X	EB	X	X		X	X	X
CQUADR	I	L C	X						X	X	X	EB	X	X		X	X	X
CROD	CS	L C	X	X	X				X			E		X	X	X	X	X
CSHEAR	CS	L	X						X			E	X	X				
CSLOT†							X											
CTETRA	I	L C	X	X	X			X	X	X		E	X	X		X	X	X
CTRIA3	I	L C	X	X	X			X	X	X	X	EB	X	X		X	X	X
CTRIA6	I	L C	X						X	X	X	EB	X	X		X	X	X
CTRIAR	I	L C	X						X	X	X	EB	X	X		X	X	X
CTRIAX6	I	L C					X		X		X	E	X	X		X	X	X
CTUBE	CS	L C	X	X	X				X			E		X	X	X	X	X
CVISC						X												
CWELD	X	X							X	X								

† For the fully nonlinear hyperelastic elements, see [Table 13-2](#), (Sec.).

* With the exception of hyperelastic elements, no midside grid points may be defined with the nonlinear stiffness formulation.

Table 3-1 Element Summary (continued)

Element Type	Data Recovery								
	Stress Real*	Stress Complex*	Force Real*	Force Complex*	Force Sum Global	Force Sum Element Edges	Structure Plot	Contour Plot	Grid Point Stresses
CAXIFi	12	23							
CBAR	16	19	9	17	X	X	X		
CBEAM	89	111	89	177	X		X		
CBEND	23	23	17	31	X		X		
CBUSH	7	13	7	13	X				
CBUSH1D	8				X				
CCONEAX**	18		7				X		
CONMi									
CONROD	5	5	3	5	X	X	X		
CRAC2D	8						X		
CRAC3D	10								
CDAMPi			2	3					
CELASi	2	3	2	3	X				
CFLUIDi							X		
CGAP	8						X		
CHBDYP, CHBDYG							X		
CHEXA	193	121			X	X	X		X
CMASSi									
CPENTA	151	95			X	X	X		X
CQUAD4	17	15	9	17	X	X	X	X	X
CQUAD8	87	77	47	87	X		X		X
CQUADR	87	77	47	87	X		X		X
CROD	5	5	3	5	X	X	X		
CSHEAR	8	5	17	33	X	X	X	X	

Table 3-1 Element Summary (continued)

Element Type	Data Recovery								
	Stress Real*	Stress Complex*	Force Real*	Force Complex*	Force Sum Global	Force Sum Element Edges	Structure Plot	Contour Plot	Grid Point Stresses
CSLOT1	7	13					X		
CTETRA	109	69			X		X		X
CTRIA3	17	15	9	17	X	X	X	X	X
CTRIA6	70	62	38	70	X		X		X
CTRIAR	70	62	38	70	X	X	X		X
CTRIAX6	30	34			X		X	X	
CTUBE	5	5	3	5	X	X	X	X	
CVISC				5			X		
CWELD			9	17					

* The integers represent the number of output words per element, useful for storage requirement calculations.

** This requires the presence of an AXISYM Case Control command.

Table 3-1 Element Summary (continued)

Element Type	Data Recovery									
	Strain Energy Density	Strain Energy	SORT2	MSGSTRESS	Heat Transfer	MSGMESH	Composite* Failure Indices	Composite* Stresses	Strain Real*	Strain Complex*
CAXIFi			X							
CBAR	X	X	X		X	X				
CBEAM	X	X	X		X	X				
CBEND	X	X	X	X	X					
CCONEAX**										
CBUSH		X	X							
CBUSH1D		X	X						8	
CONMi										
CONROD	X	X	X		X		X			
CRAC2D										
CRAC3D										
CDAMPi			X							
CELASi		X	X		X					
CFLUIDi							X			
CGAP		X	X							
CHBDYi			X		X	X				
CHEXA	X	X	X	X	X	X			193	121
CMASSi										
CPENTA	X	X	X	X	X	X			151	95
CQUAD4	X	X	X	X	X	X	9	11	17	15
CQUAD8	X	X	X		X	X	9	11	87	77
CQUADR	X	X	X		X	X	9	11	87	77
CROD	X	X	X		X	X				
CSHEAR	X	X	X			X				

Table 3-1 Element Summary (continued)

Element Type	Data Recovery									
	Strain Energy Density	Strain Energy	SORT2	MSGSTRESS	Heat Transfer	MSGMESH	Composite* Failure Indices	Composite* Stresses	Strain Real*	Strain Complex*
CSLOTi			X							
CTETRA	X	X	X		X				109	69
CTRIA3	X	X	X	X	X	X	9	11	17	15
CTRIA6	X	X	X		X	X	9	11	70	62
CTRIAR	X	X	X		X	X	9	11	70	62
CTRIAX6	X	X	X	X	X	X				
CTUBE	X	X	X		X	X				
CVISC										
CWELD	X	X	X							

* The integers represent the number of output words per element, useful for storage requirement calculations.

** This requires the presence of an AXISYM Case Control command.

Stiffness Matrix

Mass Matrix

CS-- Constant strain element

L -- Lumped mass only

LS -- Linear strain element

C --Coupled mass only

I -- Modified isoparametric element

LC -- Lumped mass or coupled mass

FD -- Frequency dependent

Thermal Load

E --Extension load only

EB -- Both extension and bending load

Integer values indicate number of items output.

3.2 Line Elements

The line elements consist of the following:

- CBEAM
- CBAR
- CBEND
- CROD
- CONROD
- CTUBE

3.2.1 Beam Element (CBEAM)

The beam element is defined with a CBEAM entry and its properties are defined with a PBEAM, PBCOMP, or PBEAML entry. The beam element includes extension, torsion, bending in two perpendicular planes, the associated shears, and the features listed below:

1. Distributed mass polar moment of inertia.
2. Separate shear center, neutral axis, and nonstructural mass center of gravity.
3. Arbitrary variation of the section properties (A, I1, I2, I12, J) and of the nonstructural mass (NSM) along the beam (PBEAM only).
4. Shear relief due to taper (PBEAM only).
5. The ability to apply either concentrated or distributed loads along the beam, using the PLOAD1 entry.
6. The effect of cross-sectional warping on torsional stiffness (PBEAM only).
7. The ability to model a beam made up of offset rods, using the PBCOMP entry.
8. Nonlinear material properties: elastic-perfectly plastic only (see TYPE = PLASTIC on MATS1 entry).

Any five of the six forces at either end of the element may be set equal to zero by using the pin flags on the CBEAM entry. The integers 1 to 6 represent the axial force, shearing force in Plane 1, shearing force in Plane 2, axial torque, moment in Plane 2 and moment in Plane 1, respectively. The structural and nonstructural mass of the beam are lumped at the ends of the elements, unless coupled mass is requested with the PARAM entry, COUPMASS (see “[Parameters](#)” on page 601 of the *MSC.Nastran Quick Reference Guide*).

The beam element coordinate system and nomenclature are shown in **Figure 3-1**. End A is offset from grid point GA an amount measured by vector \vec{w}_a , and end B is offset from grid point GB an amount measured by vector \vec{w}_b . The vectors \vec{w}_a and \vec{w}_b are measured in the global coordinates of the connected grid point. The x -axis of the element coordinate system is defined by a line connecting the shear center of end A to that at end B of the beam element. The orientation of the beam element is described in terms of two reference planes. The reference planes are defined with the aid of vector \vec{v} . This vector may be defined directly with three components in the global system at end A of the beam, or by a line drawn from end A parallel to the line from GA to a third referenced grid point. The first reference plane (Plane 1) is defined by the x -axis and the vector \vec{v} . The second reference plane (Plane 2) is defined by the vector cross product $\hat{x} \times \vec{v}$ and the x -axis. The subscripts 1 and 2 refer to forces and geometric properties associated with bending in Planes 1 and 2, respectively. The reference planes are not necessarily principal planes. The coincidence of the reference planes and the principal planes is indicated by a zero product of inertia (I_{12}) on the PBEAM entry. When pin flags and offsets are used, the effect of the pin is to free the force at the end of the element x -axis of the beam, not at the grid point.

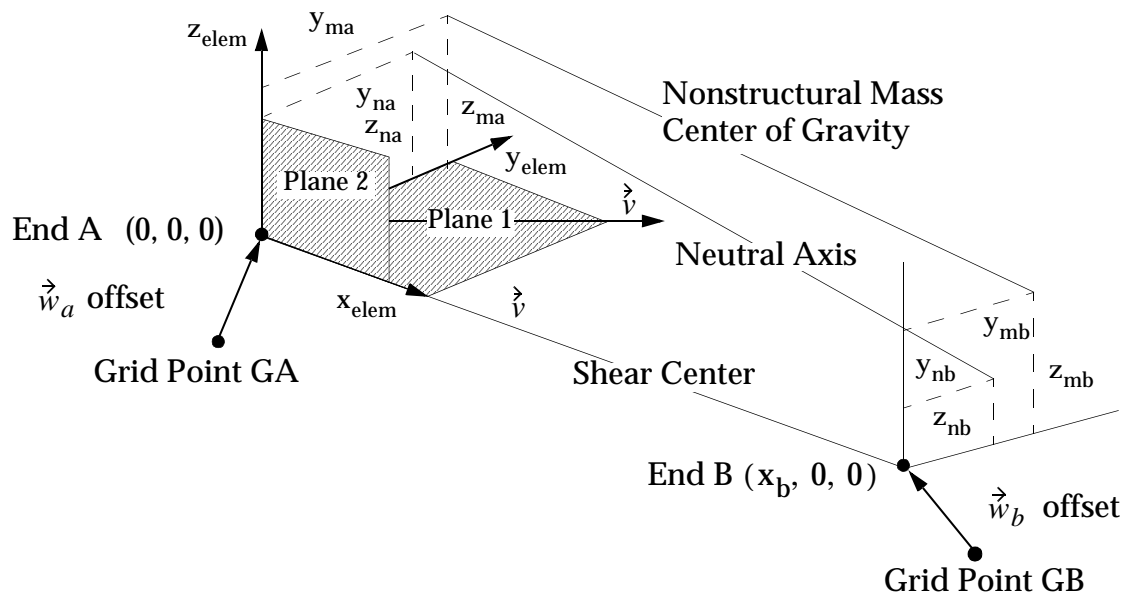


Figure 3-1 CBEAM Element Coordinate System

The positive directions for element forces are shown in **Figure 3-2**. The following element forces, either real or complex (depending on the solution sequence), are output on request at both ends and at intermediate locations defined on the PBEAM entry:

Beam element internal forces and moments

- Bending moments in the two reference planes at the neutral axis.
- Shear forces in the two reference planes at the shear center.
- Axial force at the neutral axis.
- Total torque about the beam shear center axis.
- Component of torque due to warping.

The following real element stress data are output on request:

- Real longitudinal stress at the four points prescribed for each cross section defined along the length of the beam on the PBEAM entry.
- Maximum and minimum longitudinal stresses.
- Margins of safety in tension and compression for the element if the user enters stress limits on the MAT1 entry.

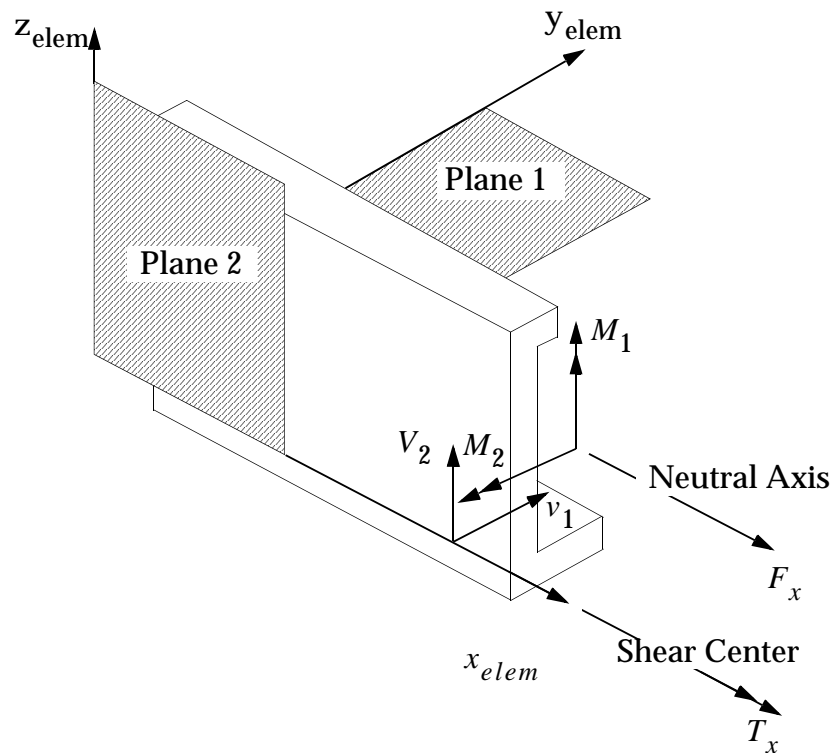


Figure 3-2 CBEAM Element Internal Forces and Moments

Tensile stresses are given a positive sign and compressive stresses a negative sign. Only the longitudinal stresses are available as complex stresses. The stress recovery coefficients on the PBEAM entry are used to locate points on the cross section for stress

recovery. The subscript 1 is associated with the distance of a stress recovery point from Plane 2. The subscript 2 is associated with the distance from Plane 1. Note that if zero value stress recovery coefficients are used, the axial stress is output.

The PBCOMP entry allows the input of offset rods to define the beam's section properties. A program automatically converts the data to an equivalent PBEAM entry. The input options that allow efficient descriptions of various symmetric cross sections are shown in **Figure 3-3**.

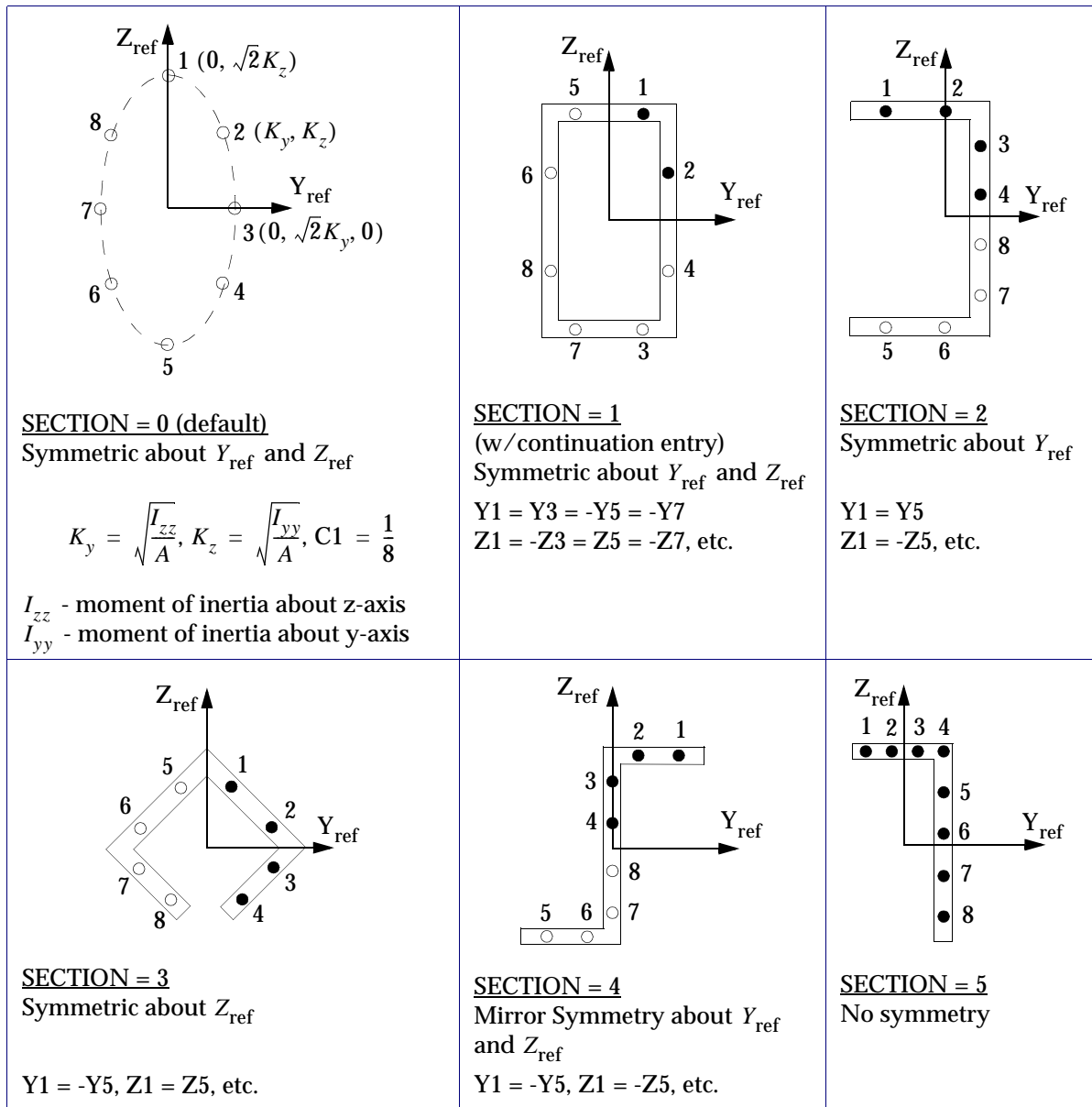


Figure Notes:

1. Integration points (lumped area) are numbered 1 through 8.
2. User-specified points are denoted by ● and the program default point denoted by ○.

Figure 3-3 PBCOMP Entry SECTION Types

Element Properties on the PBEAM Entry

The cross-sectional properties of the element which may be defined by the user and the manner in which these properties are interpolated along the x-axis of the element are listed in the following table:

Quantity	Definition	Locations at Which User Specifies Properties	Method of Interpolation
A	Cross-sectional Area	Ends, Interior points	Linear between points
I1,I2,I12	Moments and product of inertia about neutral axis for planes 1 and 2	Ends, Interior points	Linear between points
J	Torsional stiffness parameter	Ends, Interior points	Linear between points
NSM	Nonstructural mass per unit length	Ends, Interior points	Linear between points
K1,K2	Shear factors in $(K)AG$ for planes 1 and 2	One value	Constant
S1,S2	Shear relief factors for planes 1 and 2	One Value	Constant
CW	Warping stiffness parameter	Ends	Linear between ends
NSI	Nonstructural mass moment of inertia per unit length	Ends	Linear between ends

Mass Matrix

The inertia properties of the CBEAM element include the following terms:

- Structural mass per unit length, $RHO \cdot A$, on the neutral axis.
- Nonstructural mass per unit length, NSM.
- Moment of inertia of structural mass per unit length $RHO \cdot (I1 + I2)$, about neutral axis.
- Moment of inertia of nonstructural mass, NSI.

where RHO is the density defined on the MAT1 entry.

Cross Sectional Warping

Open section members such as channels will undergo torsion as well as bending when transverse loads act anywhere except at the shear center of a cross-section. This torsion produces warping of the cross-section so that plane sections do not remain plane and, as a result, axial stresses are produced. This situation can be represented in the following differential equation for the torsion of a beam about the axis of shear centers:

$$G \frac{d}{dx} \left(J \frac{d\theta}{dx} \right) - E \frac{d^2}{dx^2} \left(C_w \frac{d^2\theta}{dx^2} \right) = m \quad \text{Eq. 3-1}$$

where:

E = Young's modulus of elasticity

C_w = Warping constant

G = Shear modulus

J = Torsion constant

θ = Angle of rotation at any cross-section

m = Applied torsional moment per unit length

Note that C_w , the warping constant, has units of (length)⁶. The development of the above differential equation and methods for the numerical evaluations of the warping constant are available in the literature. (See, for example, Timoshenko and Gere, *Theory of Elastic Stability*, McGraw Hill Book Company, 1961.) An example that demonstrates the use of the warping constant is the section titled "Example Problem of Channel Section."

Shear Relief

The shear relief factor accounts for the fact that in a tapered flanged beam the flanges sustain a portion of the transverse shear load. This situation is illustrated in [Figure 3-4](#):

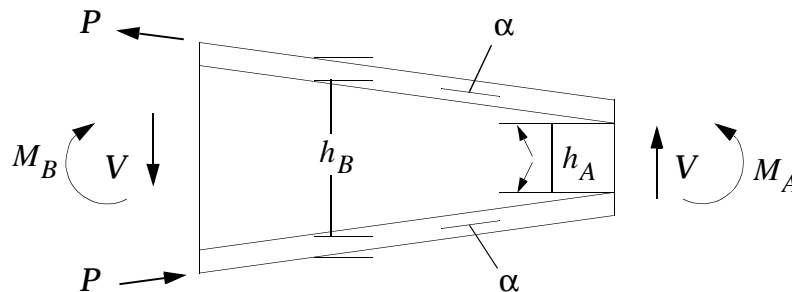


Figure 3-4

Here, the net transverse shear, Q , at the cross-section of depth h_B is

$$Q = V - 2P \sin \alpha \quad \text{Eq. 3-2}$$

and, if the entire bending moment is carried by the flanges,

$$Q = V \frac{2 \tan \alpha}{h_r} M_B \quad \text{Eq. 3-3}$$

Eq. 3-3 is represented in MSC.Nastran as:

$$Q_1 = V_1 - \frac{S_1}{l} M_1 \quad \text{Eq. 3-4}$$

$$Q_2 = V_2 - \frac{S_2}{l} M_2$$

where l is the length of the CBEAM element in question and the subscripts refer to plane 1 and plane 2, respectively. The terms s_1 and s_2 are denoted as the shear relief coefficients. The value of the shear coefficient for a tapered beam with heavy flanges that sustain the entire moment load may then be written as:

$$S_1 = \frac{2(h_A - h_B)}{(h_A + h_B)}$$

where:

h_A = depth at end A of CBEAM element (GA on CBEAM entry)

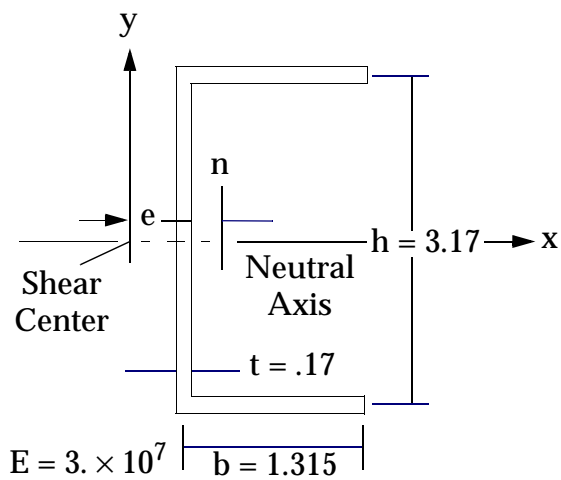
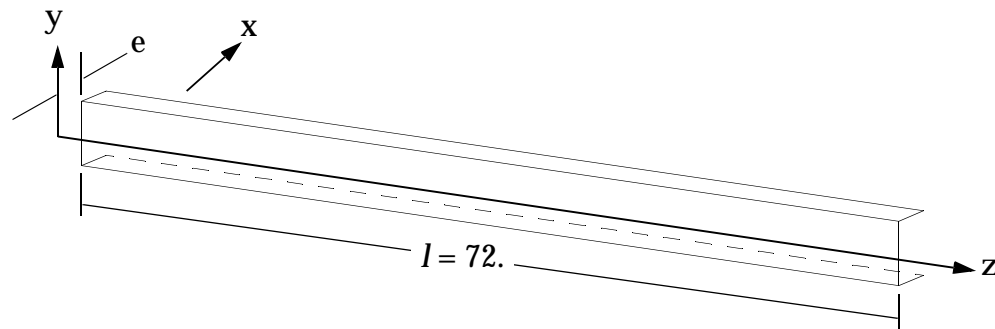
h_B = depth at end B of CBEAM element (GB on CBEAM entry)

Example Problem of Channel Section

The simply supported beam illustrated in [Figure 3-5](#) is modeled with five CBEAM elements. Since the beam is an open section channel with a single plane of symmetry, buckling failure can occur either through a combination of torsion and bending about the element x-axis or the lateral bending about the y-axis. The effect of cross-sectional warping coefficients, $CW(A)$ and $CW(B)$, on the PBEAM entry are necessary to capture these effects. Since the column is of uniform cross-section only the PBEAM entry illustrated below is required.

1	2	3	4	5	6	7	8	9	10
PBEAM	1	1	.986	1.578465	.1720965		.0094985		
	NO	1.							
							.3010320	.3010320	
						.7659450		.769450	

Warping requires 7 DOF for the beam element. Thus, for warping, each grid must have an associated scalar point.



Section Properties

$$e = \frac{3b^2}{6b + h} = .4690484$$

$$J = \frac{t^3}{3}(2b + h) = .0094985$$

$$C_w = \frac{tb^3h^2}{12} \frac{3b + 2h}{6b + h} = .3010320$$

$$A = .986$$

$$n = .2968966$$

$$I_x = 1.579465$$

$$I_y = .1720965$$

$$X_o = (e + n) = .7659450$$

Figure 3-5 Simply-Supported Channel

Example of Tapered Beam

The cantilevered tapered flanged beam illustrated in **Figure 3-6** is utilized to demonstrate the use of shear relief factors, S1 and S2, with the CBEAM element. As noted earlier in the section, shear relief provides for the fact that in a tapered flanged beam some of the transverse shear load is carried by the flanges. All of the bending load is carried by the flanges. The shear relief coefficient for the tip cell is computed by the following formula:

$$S_1 = \frac{2(h_A - h_B)}{(h_A + h_B)} = \frac{2(10. - 20.)}{(10. + 20.)} = -.66667$$

The PBEAM entry for this tip cell is illustrated below:

	1	2	3	4	5	6	7	8	9	10
PBEAM	1	1	1.	1.	60.	1.				
	5.			-5.						
	YES	1.	2.	240.						
	10.			-10.						
				-.666667						

Note: A unique PBEAM entry is required for each CBEAM element as the shear relief factor among other properties vary from element to element. Also note that the default values of 1. are accepted for the shear stiffness factors.

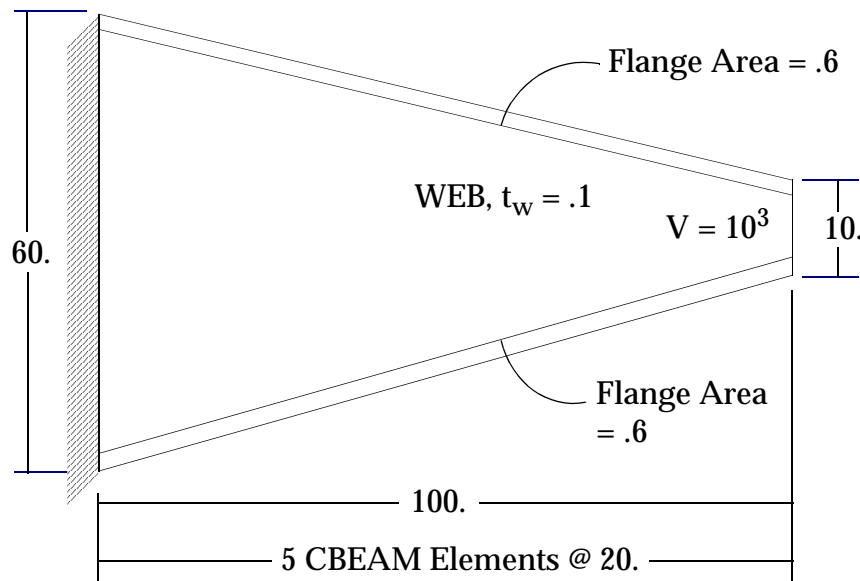


Figure 3-6 Tapered Cantilevered Beam With Shear Relief

3.2.2 Bar Element (CBAR)

The bar element is defined with a CBAR entry and its properties are defined with a PBAR entry. The bar element is a one-dimensional bending element which is prismatic, and for which the elastic axis, gravity axis, and shear center all coincide. The use of pin flags and the orientation of the bar element are similar to the beam element as shown in [Figure 3-7](#). If shearing deformations are included in the bar element, the reference axes (Planes 1 and 2) and the principal axes must coincide.

The positive directions for element forces are shown in [Figure 3-7](#). The following element forces, either real or complex (depending on the rigid format), are output on request:

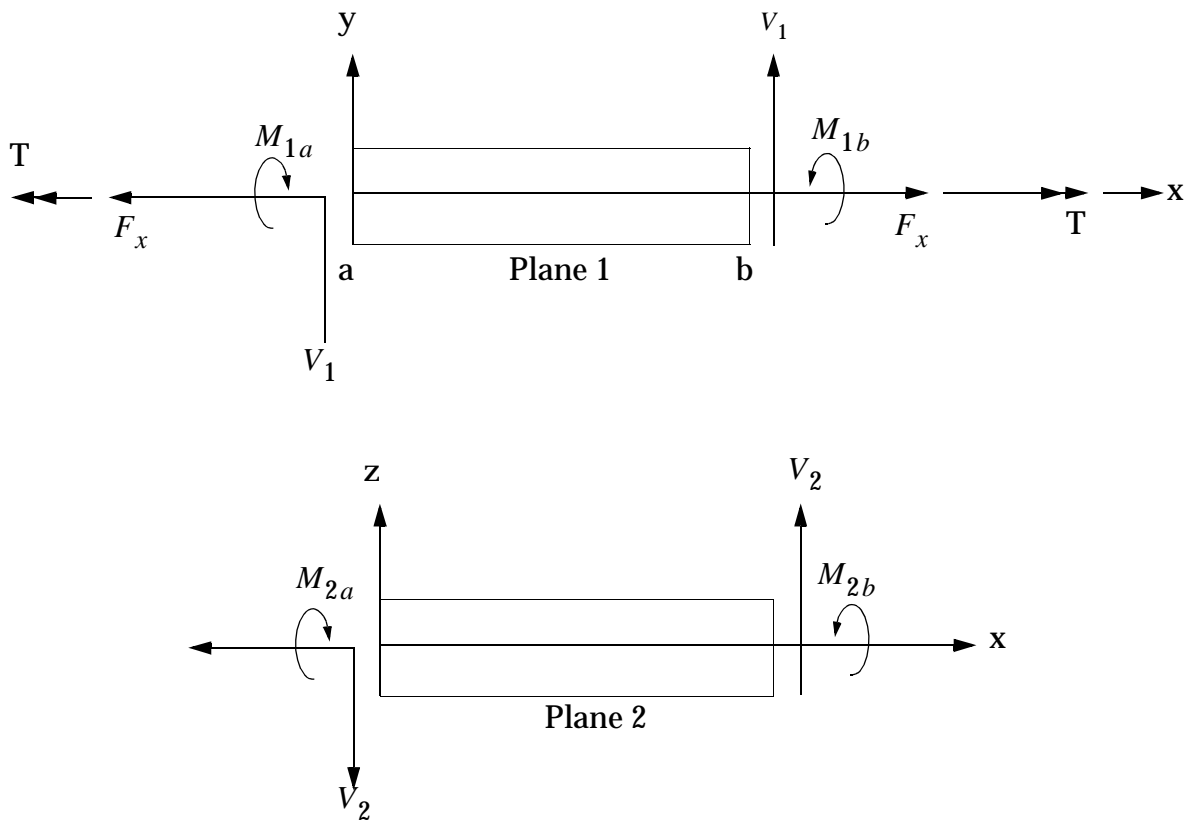


Figure 3-7 CBAR Element Forces

where M_{1a} , M_{1b} , M_{2a} , and M_{2b} are the bending moments at both ends in the two reference planes, V_1 and V_2 are the shear forces in the two reference planes, F_x is the average axial force, and T is the torque about the x -axis.

The following real element stresses are output on request:

- Average axial stress.

- Extensional stress due to bending at four points on the cross section at both ends. (Optional, calculated only if the user enters stress recovery points on the PBAR entry.)
- Maximum and minimum extensional stresses at both ends.
- Margins of safety in tension and compression for the whole element. (Optional, calculated only if the user enters stress limits on the MAT1 entry.)

Tensile stresses are given a positive sign and compressive stresses a negative sign. Only the average axial stress and the extensional stresses due to bending are available as complex stresses. The stress recovery coefficients on the PBAR entry are used to locate points on the cross section for stress recovery. The subscript 1 is associated with the distance of a stress recovery point from Plane 2. The subscript 2 is associated with the distance from Plane 1.

CBAR element force and stress data recovery with distributed loads (PLOAD1) and distributed mass (coupled mass) effects included may be obtained at intermediate as well as end points from the dynamic solution sequences. The following items must be in the input data:

- A LOADSET in Case Control which selects an LSEQ entry referencing PLOAD1 entries in the Bulk Data Section.
- Use PARAM,COUPMASS to select the coupled mass option for all elements.

The use of the BAROR Bulk Data entry avoids unnecessary repetition of input when a large number of bar elements either have the same property identification number or have their reference axes oriented in the same manner. BAROR defines default values on the CBAR entry for the property identification number and the orientation vector for the reference axes. The default values are used only when the corresponding fields on the CBAR entry are blank.

3.2.3 Bend Element (CBEND)

The bend element is defined with a CBEND entry and its properties are defined with a PBEND entry. The BEND element is a one-dimensional bending element with a constant radius of curvature. The CBEND element may be used to analyze either curved beams or pipe elbows. The bend element includes extension, torsion, bending in two perpendicular planes, and the associated transverse shear. The bend element cross section properties are assumed to be constant along the length of the element and the principal axes coincide with the element coordinates shown in [Figure 3-8](#). Structural and nonstructural mass for the bend element is obtained only as consistent mass. The lumped mass option is not available.

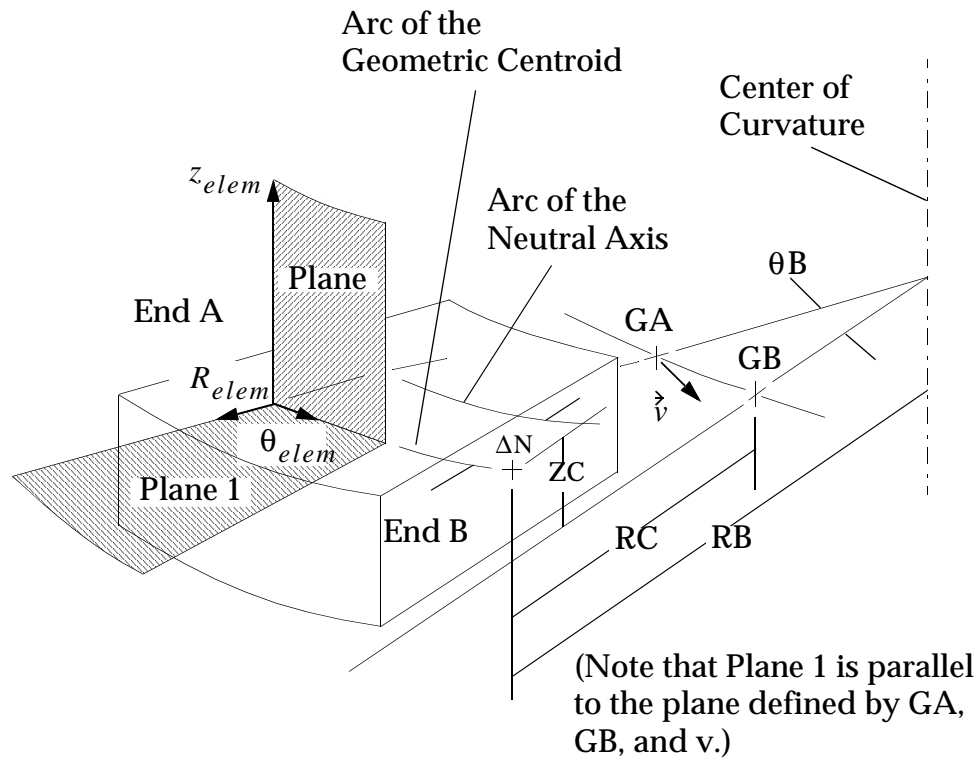


Figure 3-8 CBEND Element Coordinate System

Offsets of the ends of the element from the grid points are the same at both ends. The offsets are measured in the element coordinate system as shown in [Figure 3-8](#). The element coordinate system is defined by one of four methods that the user specifies in the GEOM field on the CBEND entry. The Z-direction of the element coordinate system is defined by the cross product $\vec{v}_{ab} \times \vec{v}$ of the vector \vec{v}_{ab} connecting grid point GA to grid point GB and the vector \vec{v} for GEOM = 1. For GEOM = 2, 3, or 4, the Z-direction is defined by the cross product $\vec{v} \times \vec{v}_{ab}$. The center of curvature and intersection of the tangent lines from end A and end B are located using the data required for each of the four options. The R-direction is obtained by the vector extending from the center of curvature to end A. The θ -direction is the cross product of $\vec{Z} \times \vec{R}$. When $\theta = 0$, end A of the element is indicated and $\theta = \theta_B$ represents end B. Plane 1 of the element lies in the $R\theta$ plane of the element coordinates. Plane 1 is parallel to the plane defined by GA, GB and the vector \vec{v} , but it is offset by ZC in the Z-direction. Plane 2 lies in the θZ plane and is offset from GA and GB by RC in the R-direction. The subscripts 1 and 2 refer to forces and geometric properties associated with bending in Planes 1 and 2, respectively. These reference planes are the principal planes of the element cross section.

The neutral axis radial offset shown in [Figure 3-8](#) from the geometric centroid due to bending of a curved beam with a constant radius of curvature is defined as follows:

$$\Delta N = \frac{R_B}{1 + \frac{AR_R^2}{Z}} \quad \text{Eq. 3-5}$$

where:

R_B = the bend radius

A = the cross section area

$$Z = \int \frac{r^2 dA}{1 + \frac{R}{R_B}}$$

r = a local variable aligned with R_{elem} direction

The user can use the default provided with the general format or he can calculate and input a value using the above formula. For the circular section format, the neutral axis offset is automatically calculated with analytical expressions for hollow and solid circular cross-sectional elements.

The flexibility factors which multiply the bending terms of the flexibility matrix are selected by the FSI field on the hollow circular section format of the property entry. The options available are

for FSI = 1:

$$k_p = 1.0$$

for FSI = 2:

$$k_p = \frac{1.65r^2}{R_B t} \left[\frac{1}{1 + 6 \frac{Pr}{Et} \left(\frac{r}{t}\right)^{4/3} \left(\frac{R_B}{r}\right)^{1/3}} \right] \quad \text{Eq. 3-6}$$

for FSI = 3:

$$k_p = \frac{1.66}{R_B t} \left[\frac{1}{1 + 1.75\lambda^{4/3} \exp(-1.5\Psi^{-1/4})} \right] \quad \text{Eq. 3-7}$$

where:

$$\lambda = \frac{R_B t}{r^2 \sqrt{1 - \nu^2}}$$

$$\Psi = \frac{P R_B^2}{E r t}$$

r = mean cross-sectional radius

t = wall thickness

P = internal pressure

E = Young's modulus

ν = Poisson's ratio

The positive sign conventions for internal element forces are shown in **Figure 3-9**. The following element forces, either real or complex (depending on the rigid format), are output on request at both ends:

- Bending moments in the two reference planes, M_1 and M_2 .
- Shears in the two reference planes, V_1 and V_2 .
- Average axial force, F_θ .
- Torque about the bend axis, M_θ .

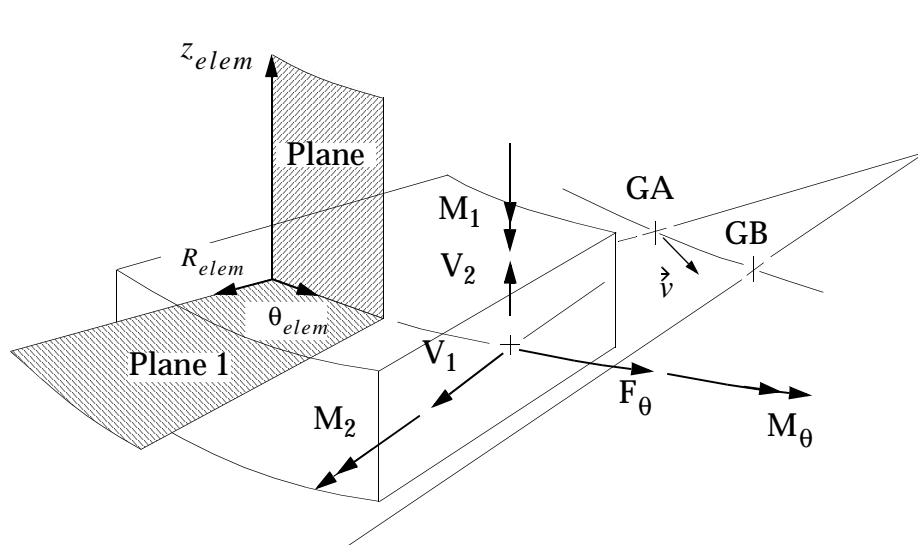
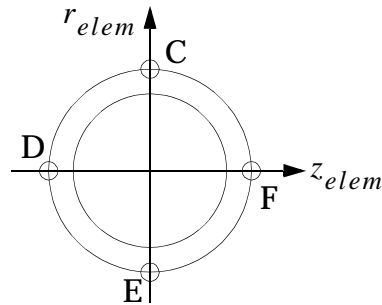


Figure 3-9 CBEND Element Internal Forces and Moments

The following real element stress data are output on request:

- Real longitudinal stress at the four points which are the same at both ends for the general cross-sectional property entry format. If the circular cross-sectional property and format is used, the stress points are automatically located at the points indicated on the following figure.



- Maximum and minimum longitudinal stresses.
- Margins of safety in tension and compression for the element if the user enters stress limits on the MAT1 entry.
- Whenever the pipe format is used, the stress data are modified to account for stress intensification resulting from internal pressurization and curvature of the element. The internal pressure is prescribed on the property entry. The methods used to calculate the stress intensification factor are selected through the FSI parameters; see the Remarks under the “**PBEND**” on page 1810 of the .

Tensile stresses are given a positive sign and compressive stresses a negative sign. Only the longitudinal stresses are available as complex stresses. The stress recovery coefficients on the general form of the PBEND entry are used to locate points on the cross section for stress recovery. The subscript 1 is associated with the distance of a stress recovery point from Plane 2. The subscript 2 is associated with the distance from Plane 1. If zero-value stress recovery coefficients are used, the axial stress is output.

3.2.4 Rod Elements (CROD, CONROD, CTUBE)

The rod element is defined with a CROD entry and its properties with a PROD entry. The rod element includes extensional and torsional properties. The CONROD entry is an alternate form that includes both the connection and property information on a single entry. The tube element is a specialized form that is assumed to have a circular cross section. The tube element is defined with a CTUBE entry, and its properties with a PTUBE entry. The structural and nonstructural mass of the rod are lumped at the adjacent grid points unless coupled mass is requested with the PARAM,COUPMASS

(see “**Parameters**” on page 601 of the *MSC.Nastran Quick Reference Guide*). Theoretical aspects of the rod element are treated in Section 5.2 of *The NASTRAN Theoretical Manual*.

The x -axis of the element coordinate system is defined by a line connecting end a to end b, as shown in **Figure 3-10**. The axial force and torque are output on request in either real or complex form. The positive directions for these forces are indicated in **Figure 3-10**.

The following real element stresses are output on request:

- Axial stress.
- Torsional stress.
- Margin of safety for axial stress.
- Margin of safety for torsional stress.

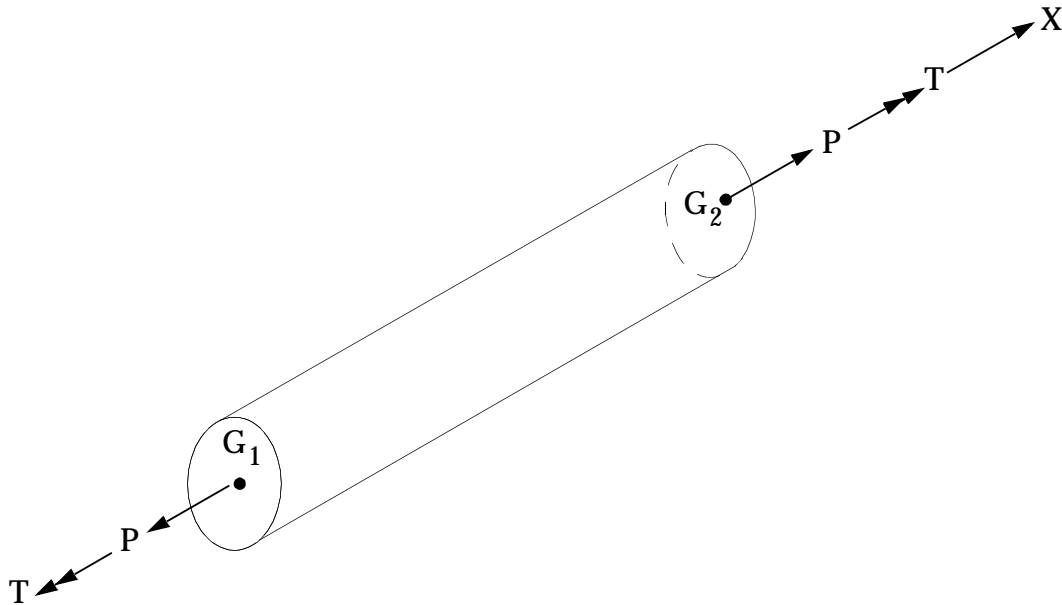


Figure 3-10 Rod Element Coordinate System and Element Forces

Positive directions are the same as those indicated in **Figure 3-10** for element forces. Only the axial stress and the torsional stress are available as complex stresses.

Another kind of rod element is the viscous damper. It has extensional and torsional viscous damping properties rather than stiffness properties. The viscous damper element is defined with a CVISC entry and its properties with a PVISC entry. This element is used in the formulation of dynamic matrices. The mode displacement method (PARAM,DDRMM,-1) must be selected for element force output.

3.2.5 Supplied Beam and Bar Libraries

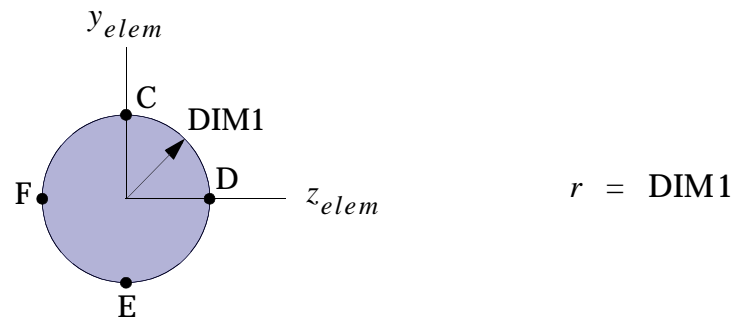
The Bulk Data entries PBARL and PBEAML provide libraries of cross sections. For open sections and the HAT1, thin-walled theory is assumed.

The following pages provide equations that define the PBEAM and PBAR geometric property entries in terms of entries on the PBEAML or PBARL. Symbols absent for a particular cross section are normally set to zero.

A	Cross-sectional area.
y_c	Distance to centroid along Y element axis.
z_c	Distance to centroid along Z element axis.
y_s	Distance to shear center along Y element axis.
z_s	Distance to shear center along Z element axis.
I_1	Moment of inertia about the Z element axis at the centroid. $I_1 = I_{(zz)_{elem}}$
I_2	Moment of inertia about the Y element axis at the centroid. $I_2 = I_{(yy)_{elem}}$
I_{12}	Product moment of inertia at the centroid. $I_{12} = I_{(zy)_{elem}}$
J	Torsional Stiffness Constant.
C, D, E, F	Location of the stress recovery points in the element coordinate system relative to the shear center. For the PBARL the locations must be changed to be relative to the centroid. This can be done by adding y_{na} , z_{na} to the listed equations.
K_1, K_2	Shear stiffness factor for plane 1 and plane 2.
I_w	Warping coefficient for the cross section relative to the shear center.
y_{na}, z_{na}	Coordinates of the centroid relative to the shear center.

$$A = \pi r^2$$

$$\left. \begin{array}{l} I_1 \\ I_2 \end{array} \right\} = \frac{\pi}{4} r^4$$



TYPE = "ROD"

$$J = \frac{\pi}{2} r^4$$

$$C = r, 0$$

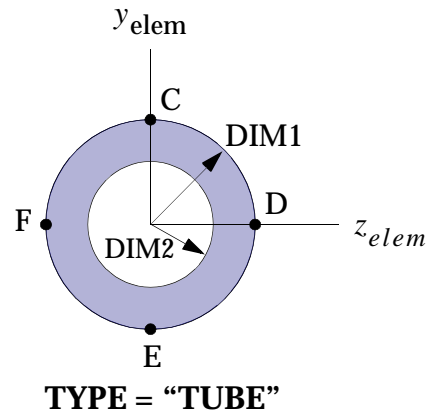
$$D = 0, r$$

$$E = -r, 0$$

$$F = 0, -r$$

$$\left. \begin{array}{l} K_1 \\ K_2 \end{array} \right\} = 0.9$$

Figure 3-11 Geometric Property Formulas for a ROD



$$r_0 = \text{DIM1}$$

$$r_i = \text{DIM2}$$

$$A = \pi(r_0^2 - r_i^2)$$

$$\left. \begin{matrix} I_1 \\ I_2 \end{matrix} \right\} = \frac{\pi}{4}(r_0^4 - r_i^4)$$

$$J = \frac{\pi}{2}(r_0^4 - r_i^4)$$

$$C = r_0, 0$$

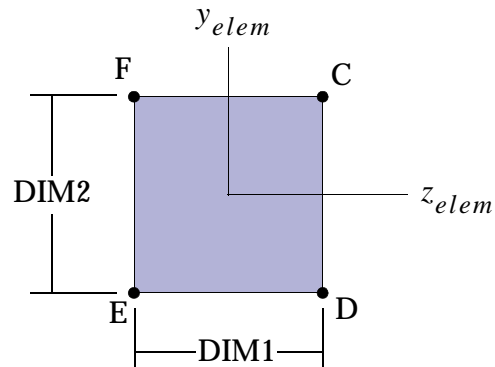
$$D = 0, r_0$$

$$E = -r_0, 0$$

$$F = 0, -r_0$$

$$\left. \begin{matrix} K_1 \\ K_2 \end{matrix} \right\} = 0.5$$

Figure 3-12 Geometric Property Formulas for a TUBE Cross Section



$$b = \text{DIM1}$$

$$h = \text{DIM2}$$

TYPE = "BAR"

$$A = bh$$

$$I_1 = \frac{bh^3}{12}$$

$$I_2 = \frac{b^3h}{12}$$

$$I_{12} = 0.$$

$$J = bh^3 \left[\frac{1}{3} - .21 \frac{h}{b} \left(1 - \frac{h^4}{12b^4} \right) \right]$$

$$C = \frac{h}{2}, \frac{b}{2}$$

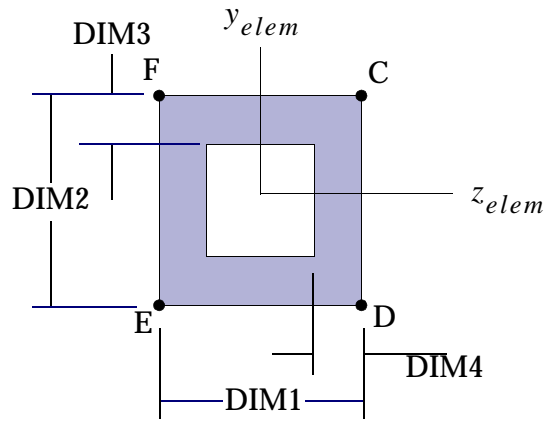
$$D = -\frac{h}{2}, \frac{b}{2}$$

$$E = -\frac{h}{2}, -\frac{b}{2}$$

$$F = \frac{h}{2}, -\frac{b}{2}$$

$$\left. \begin{array}{l} K_1 \\ K_2 \end{array} \right\} = \frac{5}{6}$$

Figure 3-13 Geometric Property Formulas for a BAR



$$\begin{aligned} b &= \text{DIM1} \\ h &= \text{DIM2} \\ t_1 &= \text{DIM3} \\ t_2 &= \text{DIM4} \\ b_i &= b - 2t_2 \\ h_i &= h - 2t_1 \end{aligned}$$

TYPE = "BOX"

$$A = bh - b_i h_i$$

$$I_1 = \frac{bh^3}{12} - \frac{b_i h_i^3}{12}$$

$$I_2 = \frac{hb^3}{12} - \frac{h_i b_i^3}{12}$$

$$I_{12} = 0.$$

$$J = \frac{2t_2 t_1 (b - t_2)^2 (h - t_1)^2}{bt_2 + ht_1 - t_2^2 - t_1^2}$$

$$C = \frac{h}{2}, \frac{b}{2}$$

$$D = -\frac{h}{2}, \frac{b}{2}$$

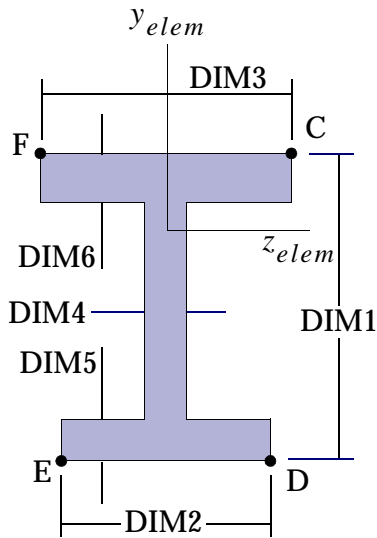
$$E = -\frac{h}{2}, -\frac{b}{2}$$

$$F = \frac{h}{2}, -\frac{b}{2}$$

$$K_1 = \frac{h_i t_2}{A} \cdot 2$$

$$K_2 = \frac{b_i t_1}{A} \cdot 2$$

Figure 3-14 Geometric Property Formulas for a BOX



TYPE = "I"

$$a = \text{DIM2}$$

$$b = \text{DIM3}$$

$$h = \text{DIM1}$$

$$t_a = \text{DIM5}$$

$$t_b = \text{DIM6}$$

$$t_w = \text{DIM4}$$

$$h_w = h - (t_a + t_b)$$

$$h_f = h - 0.5(t_a + t_b)$$

$$A = t_a a + h_w t_w + b t_b$$

$$y_c = \frac{1}{A} \left(\frac{1}{2} h_w (h_w + t_a) t_w + h_f t_b b \right)$$

$$y_s = \frac{t_b h_f b^3}{(t_b b^3 + t_a a^3)}$$

$$y_{na} = y_c - y_s$$

$$I_1 = \frac{b t_b^3}{12} + \frac{a t_a^3}{12} + \frac{t_w h_w^3}{12} + (h_f - y_c)^2 b t_b + y_c^2 a t_a + (y_c - 0.5(h_w + t_a))^2 h_w t_w$$

$$I_2 = \frac{b^3 t_b}{12} + \frac{t_a a^3}{12} + \frac{h_w t_w^3}{12}$$

$$I_{12} = 0.$$

$$J = \frac{1}{3} (t_b^3 b + t_a^3 a + t_w^3 h_f)$$

(continued)

Note that I_c and y_s are relative to the center of flange defined by Dim1 and Dim4.

$$C = h_f - y_s + .5t_b, .5b$$

$$D = -y - .5t_a, .5a$$

$$E = -y - .5t_a, -.5a$$

$$F = h_f - y_s + .5t_b, -.5a$$

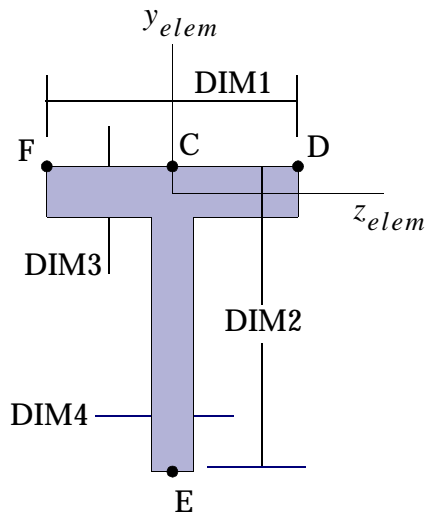
$$K_1 = \frac{t_w h_w}{A}$$

$$K_2 = \frac{5(at_a + bt_b)}{6A}$$

$$I_w = \frac{h^2 t_b t_a b^3 a^3}{12(t_b b^3 + t_a a^3)}$$

Note that I_w and y_s are based on thin walled formulations.

Figure 3-15 Geometric Property Formulas for an I Section



$$\begin{aligned} d &= \text{DIM1} \\ h &= \text{DIM2} - 0.5t_f \\ t_f &= \text{DIM3} \\ t_w &= \text{DIM4} \\ h_w &= \text{DIM2} - t_f \end{aligned}$$

TYPE = "T"

$$A = dt_f + h_w t_w$$

(continued)

$$y_{na} = -\frac{h_w t_w (h_w + t_f)}{2A}$$

$$I_1 = \frac{dt_f^3}{12} + \frac{t_w h_w^3}{12} + h_w t_w (y_{na} + .5(h_w + t_f))^2 + dt_f y_{na}^2$$

$$I_2 = \frac{t_f d^3}{12} + \frac{h_w t_w^3}{12}$$

$$I_{12} = 0.$$

$$J = \frac{1}{3}(t_f^3 d + t_w^3 h)$$

$$C = .5t_f, 0$$

$$D = .5t_f, .5d$$

$$E = -h, 0$$

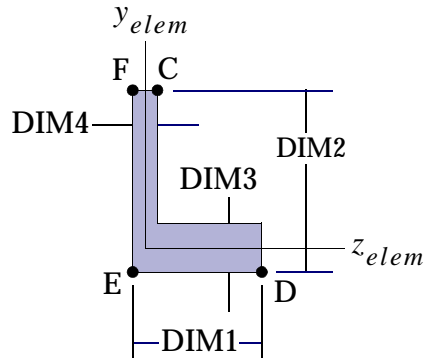
$$F = .5t_f, -.5d$$

(continued)

$$K_1 = \frac{h_w t_w}{A}$$

$$K_2 = \frac{t_f d}{A}$$

Figure 3-16 Geometric Property Formulas for a T Section



$$\begin{aligned}
 b &= \text{DIM1} - 0.5t_2 \\
 h &= \text{DIM2} - 0.5t_1 \\
 t_1 &= \text{DIM3} \\
 t_2 &= \text{DIM4} \\
 h_2 &= \text{DIM2} - t_1 \\
 b_1 &= \text{DIM1} - t_2
 \end{aligned}$$

TYPE = "L"

$$A = (b + .5t_2)t_1 + h_2t_2$$

$$y_c = \frac{t_2h_2(h_2 + t_1)}{2A}$$

$$z_c = \frac{t_1b_1(b_1 + t_2)}{2A}$$

$$I_1 = \frac{t_1^3(b + .5t_2)}{12} + t_1(b + .5t_2)y_c^2 + \frac{t_2h_2^3}{12} + h_2t_2(0.5(h_2 + t_1) - y_c)^2$$

$$I_2 = \frac{t_2^3h_2}{12} + t_2h_2z_c^2 + \frac{t_1(b + .5t_2)^3}{12} + t_1(b + .5t_2)(.5b_1 - z_c)^2$$

$$I_{12} = z_c y_c t_1 t_2 - b_1 t_1 y_c (.5(b_1 + t_2) - z_c) - h_2 t_2 z_c (.5(h_2 + t_1) - y_c)$$

$$J = \frac{1}{3}(t_1^3 b + t_2^3 h)$$

$$C = h, .5t_2$$

$$D = -.5t_1, b$$

$$E = .5t_1, -.5t_2$$

$$F = h, -.5t_2$$

(continued)

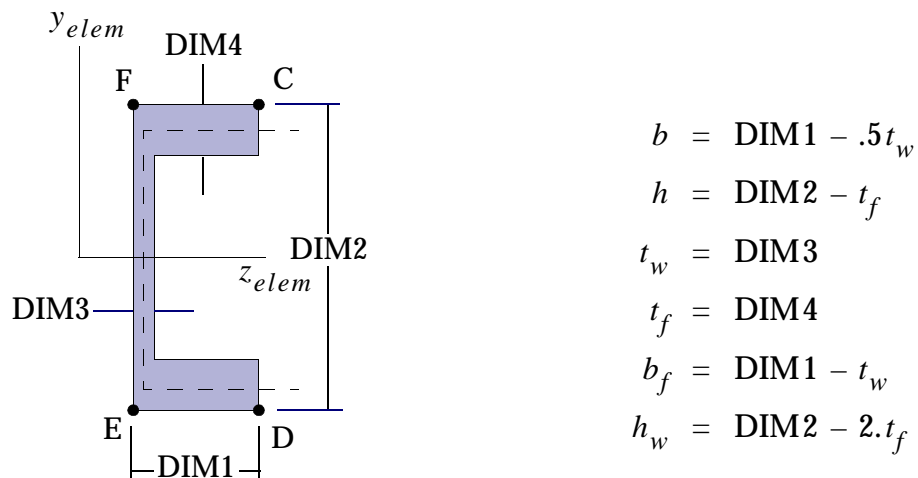
$$K_1 = \frac{h_2 t_2}{A}$$

$$K_2 = \frac{b_1 t_1}{A}$$

$$y_{na} = y_c$$

$$z_{na} = z_c$$

Figure 3-17 Geometric Property Formulas for an L Section



$$b = \text{DIM1} - .5t_w$$

$$h = \text{DIM2} - t_f$$

$$t_w = \text{DIM3}$$

$$t_f = \text{DIM4}$$

$$b_f = \text{DIM1} - t_w$$

$$h_w = \text{DIM2} - 2.t_f$$

TYPE = "CHAN"

$$A = 2t_f b_f + (h + t_t)t_w$$

$$z_c = \frac{b_f t_f (b_f + t_w)}{A}$$

$$z_s = \frac{b^2 t_f}{\left(2bt_w + \frac{1}{3}ht_f\right)}$$

$$I_1 = \frac{h^2 t_f b_f}{2} + \frac{b_f t_f^3}{6} + \frac{(h + t_f)^3 t_w}{12}$$

(continued)

$$I_2 = \frac{(h + t_f)t_w^3}{12} + \frac{b_f^3 t_f}{6} + .5(b_f + t_w)^2 b_f t_f - z_c^2 A$$

$$J = \frac{1}{3}(2b t_f^3 + h t_w^3)$$

$$I_w = \frac{t_f b^3 h^2}{12} \left(\frac{2t_w h + 3t_f b}{t_w h + 6t_f b} \right)$$

$$C = .5(h + .5t_f), z_s + b$$

$$D = -.5(h + .5t_f), z_s + b$$

$$E = -.5(h + .5t_f), z_s - .5t_w$$

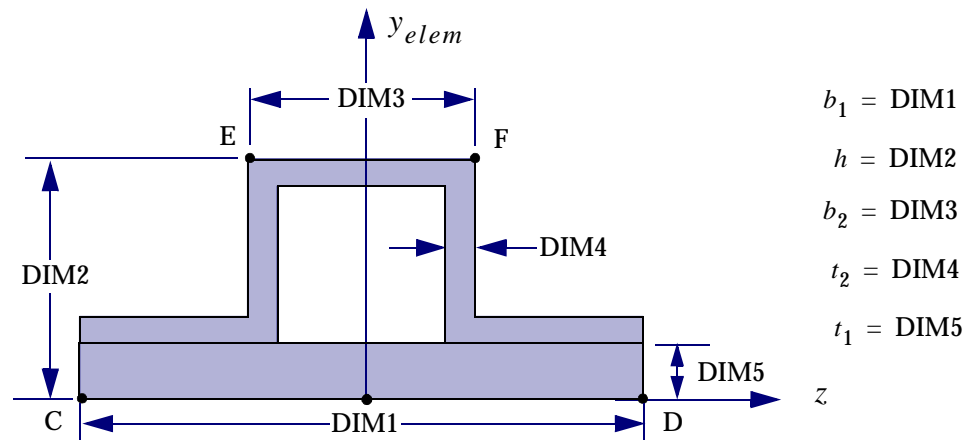
$$F = .5(h + .5t_f), z_s - .5t_w$$

$$K_1 = \frac{t_w h_w}{A}$$

$$K_2 = \frac{2t_f b_f}{A}$$

$$z_{na} = z_c + z_s$$

Note that z_c, z_s are distances measured relative to an origin positioned at the center of the web.



- $b_1 = \text{DIM1}$
- $h = \text{DIM2}$
- $b_2 = \text{DIM3}$
- $t_2 = \text{DIM4}$
- $t_1 = \text{DIM5}$

TYPE="HAT1"

Figure 3-18 Geometric Property Formulas for a Closed-hat Section

$$A = b_1 t_1 + t_2(b_1 - b_2) + 2t_2(h - t_1) + t_2(b_2 - 2t_2)$$

$$y_c = \frac{(b_1 t_1^2 / 2) + t_2(b_1 - b_2)(t_1 + t_2 / 2) + t_2(h^2 - t_1^2) + t_2(b_2 - 2t_2)(h - t_2 / 2)}{A}$$

$$y_s = \frac{\Psi}{\Psi_1} + \frac{t_2}{2}$$

where:

$$\Psi_1 = \frac{t_2(b_1 - b_2 + t_2)^3}{6} + \frac{t_2(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)^2}{16} + \frac{t_2(h - t_1 - t_2)(b_2 - t_2)^2}{2} + \frac{t_2(b_2 - t_2)^3}{12} + \frac{t_1 b_1^3}{12}$$

$$\begin{aligned} \Psi = & (b_2 - t_2)(h - t_1 / 2 - t_2 / 2) \left[\frac{t_1(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)}{24} + \frac{t_1 b_1^2}{12} + \Psi_3 \right] \\ & + (b_2 - t_2)(h - t_1 - t_2) \left[\frac{t_2(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)}{8} + \frac{t_2(h - t_1 - t_2)(b_2 - t_2)}{4} + \Psi_3 \right] + \frac{t_1 t_2 b_1^3}{12} \\ & - \frac{t_1 t_2 (b_1 - b_2 + t_2)(b_1 + b_2 - t_2)(b_2 - t_2)}{8} + \frac{t_2^2 b_1^3}{12} - \frac{t_1 t_2 b_1^2 (b_2 - t_2)}{12} \\ & + \frac{t_1 t_2 (b_2 - t_2)(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)}{12} - \frac{t_2^2 (b_2 - t_2)(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)}{8} - \frac{t_2^2 b_1^2 (b_2 - t_2)}{12} \\ & + \frac{t_2^2 (b_2 - t_2)(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)}{12} \end{aligned}$$

and

$$\Psi_3 = \frac{t_1 t_2 \Psi_2}{2t_1(h - t_1 - t_2) + (b_2 - t_2)(t_1 + t_2)}$$

where:

$$\begin{aligned} \Psi_2 = & \frac{(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)(h - t_1 - t_2)}{4} + \frac{(h - t_1 - t_2)^2 (b_2 - t_2)}{2} + \frac{(b_2 - t_2)^3}{12} \\ & + \frac{(b_1 - b_2 + t_2)(b_1 + b_2 - t_2)(b_2 - t_2)}{6} + \frac{(h - t_1 - t_2)(b_2 - t_2)^2}{2} + \frac{b_1^2 (b_2 - t_2)}{12} \end{aligned}$$

$$y_{na} = y_s - (h - y_c)$$

$$\begin{aligned}
 I_1 = & \frac{b_1 t_1^3}{12} + b_1 t_1 (y_c - t_1/2)^2 + \frac{(b_1 - b_2) t_2^3}{12} \\
 & + (b_1 - b_2) t_2 (y_c - (t_1 + t_2/2))^2 \\
 & + \frac{t_2 (h - t_1)^3}{6} + 2 t_2 (h - t_1) \left(y_c - \left(\frac{h - t_1}{2} \right) + t_1 \right)^2 \\
 & + \frac{(b_2 - 2 t_2) t_2^3}{12} + (b_2 - 2 t_2) t_2 (y_c - (h - t_2/2))^2
 \end{aligned}$$

$$\begin{aligned}
 I_2 = & \frac{t_1 b_1^3}{12} + \frac{t_2 (b_1 - b_2)^3}{48} + \frac{t_2 (b_1 - b_2) (b_1 + b_2)^2}{16} \\
 & + \frac{(h - t_1) t_2^3}{6} + \frac{(h - t_1) t_2 (b_2 - t_2)^2}{2} \\
 & + \frac{2 t_3 (b_2/2 - t_2)^3}{3}
 \end{aligned}$$

$$I_{12} = 0$$

$$J = \frac{4 t_1 t_2 (b_2 - t_2)^2 (h - t_1/2 - t_2/2)^2}{(b_2 - t_2)(t_1 + t_2) + t_1 (2h - t_1 - t_2)} + (b_1 - b_2)(t_1 + t_3)^3 / 3$$

$$C = y_s - h, -b_1/2$$

$$D = y_s - h, b_1/2$$

$$E = y_s, -b_2/2$$

$$F = y_s, b_2/2$$

$$K_1 = \frac{1}{n} \sum_{i=1}^n \frac{2(h - t_1) t_2}{m A}$$

$$K_2 = \frac{1}{n} \sum_{i=1}^n \frac{1 - 2(h - t_1) t_2}{m A}$$

where n is the number of sections and $m = 2$ for end A or end B and $m = 1$ for any intermediate station.

3.2.6 Adding Your Own Beam Cross Section Library

The standard cross sections provided by MSC should be adequate in the majority of cases. If these standard sections are not adequate for your purposes, you can add your own library of cross sections to suit your needs. To add your own library, you need to write few simple subroutines in FORTRAN and link them to MSC.Nastran through inter-process communications.

The *MSC.Nastran 2004 Installation and Operations Guide* will describe to you the current server requirements and provide you with the location of starter subroutines as described below.

This process requires writing and/or modifying up to eight basic subroutines:

1. BSCON -- Defines the number of dimensions for each of the section types.
2. BSBRPD -- Calculates section properties based on section dimensions.
3. BSGRQ -- Defines NSECT, the number of section types, and NDIMAX, the maximum number of dimensions (including nonstructural mass) required by any of the sections.
4. BSBRT -- Provides the name, number of dimensions and number of design constraints for each section type.
5. BSBRID -- Provides information for the calculation of gradients of section properties with respect to section dimensions.
6. BSBRGD -- Calculates any nonlinear gradients of section properties with respect to section dimensions.
7. BSBRC D -- Defines constraints in the design of section dimensions.
8. BSMS -- A utility routine; handles errors that occur in the beam library.

BSCON and BSBRPD are always required. BSGRQ, BSBRT, and BSBRID are required if you wish to perform sensitivity and/or optimization tasks using the beam library. BSBRGD is required if you are providing nonlinear analytical sensitivities in the design task, and BSBRC D is an optional routine that can be provided to help the optimizer to stay within physical design constraints. BSMSG handles any error messages you feel are appropriate.

This section describes each of these basic routines in turn and provides MSC's version of each. Routines that are called by these basic routines are also described with adequate examples to allow you to construct your own library. All the example routines shown are for the 32-bit machines, such as IBM RISC/System 6000.

For 64-bit machines such as CRAY C90, all the routine names that end with "D" should be changed to end with "S," and all real variables must be designated as single precision instead of double precision. Therefore, the naming convention for routines on 64-bit machines are: BSCON, BSBRPS, BSGRQ, BSBRT, BSPRIS, BSBRGS, BSBRCS, and BSMSG.

BSCON SUBROUTINE

This routine provides the number of fields in the continuation lines to be read from the Bulk Data entries PBARL and PBEAML for each cross section in the library. The value of the ENTYP variable may be 0, 1, or 2. When ENTYP = 0, the value returned is the number of DIMi. When ENTYP = 1, the value returned includes both the DIMi and NSM fields. The value of 1 is used for PBARL only. When ENTYP = 2, the value returned includes the DIMi, NSM, SO, and XIXB fields for 11 different stations. The value of 2 applies to PBEAML only.

The calling sequence and example routine for the standard MSC library is given below.

```

SUBROUTINE BSCON( GRPID , TYPE , ENTYP , NDIMI , ERROR )
C -----
C Purpose
C To get the number of maximum fields in continuation entries for
C each section in the library.
C
C Arguments:
C
C GRPID input integer Integer id of this group or group name.
C Not used, reserved for future use.
C TYPE input character*8 Name of cross section
C ENTYP input integer 0: dimensions only without NSM
C 1: PBARL, total # of data items for 2:PBEAML
C NDIMI output integer Number of dimi fields for the 'ENTYP'
C section
C ERROR output integer Error code
C
C Called by BCCON
C -----
C=== Argument Type Declaration
INTEGER GRPID, ENTYP, NDIMI, ERROR
CHARACTER*8 TYPE
C=== Default to 'nothing wrong'
```

```

        ERROR = 0

C=== Dimensions vary with section type
        IF      ( TYPE.EQ.'ROD      ' ) THEN
            NDIMI = 1
        ELSEIF( TYPE.EQ.'TUBE     ' .OR. TYPE.EQ.'BAR      ' ) THEN
            NDIMI = 2
        ELSEIF( TYPE.EQ.'HEXA     ' ) THEN
            NDIMI = 3
        ELSEIF( TYPE.EQ.'BOX      ' .OR. TYPE.EQ.'T        ' .OR.
+           TYPE.EQ.'L          ' .OR.
+           TYPE.EQ.'CHAN       ' .OR. TYPE.EQ.'CROSS     ' .OR.
+           TYPE.EQ.'H          ' .OR.
+           TYPE.EQ.'I1         ' .OR. TYPE.EQ.'T1         ' .OR.
+           TYPE.EQ.'CHAN1      ' .OR.
+           TYPE.EQ.'Z          ' .OR. TYPE.EQ.'CHAN2     ' .OR.
+           TYPE.EQ.'T2         ' .OR. TYPE.EQ.'HAT        ' ) THEN
            NDIMI = 4
        ELSEIF( TYPE.EQ.'I        ' .OR. TYPE.EQ.'BOX1     ' ) THEN
            NDIMI = 6
        ELSE
C=== Set error code if invalid name for the section
            ERROR = 5150
            RETURN
        ENDIF

C=== Number of data items to be read on PBARL entry is DIMI
C=== plus the NSM field

        IF ( ENTYP.EQ.1 ) NDIMI = NDIMI+1

C=== Number of data items to be read on PBEAML entry is DIMI
C=== plus the NSM, SO and X/XB fields for eleven different
C=== stations.

        IF ( ENTYP.EQ.2 ) NDIMI = (NDIMI+3)*11

C-----
        RETURN
        END

```

BSBRPD Subroutine

Finite element analysis requires section properties such as area, moment of inertia, etc., instead of section dimension. Therefore, the dimensions specified on PBARL and PBEAML need to be converted to equivalent properties usually specified on PBAR and PBEAM entries. The images of all these entries are stored in EPT datablock as records.


```

C           the PEBAM EPT record.
C           DIMI   input   real       Array containing the floating words
C                                           in the PBARL or PBEAML EPT record
C                                           for the 'TYPE' section
C           NDIMI  input   integer    Dimension of the DIMI array.
C           DIMO   output  real       Array conatining the real words for
C                                           the PBAR or PBEAM EPT record.
C           NDIMO  input   integer    Dimension of the DIMO array.  It is
C                                           equal to 17 for the PBAR and 193 for
C                                           the PBEAM EPT record.
C           ERROR  output  integer    Error code
C
C-----
C   CALLED BY:
C           BCBRP
C-----
C   CALLS:
C           MEVBRD ,MEVBMD
C-----
C   IMPLICIT DECLARATIONS
C           IMPLICIT INTEGER (I-N)
C           IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C.....
C   EXPLICIT DECLARATIONS
C           INTEGER ENTYP ,ERROR ,GRPID
C           CHARACTER*8 TYPE
C-----
C   DIMENSION STATEMENTS
C           INTEGER IDI(NID) , IDO(NIDO)
C           DOUBLE PRECISION DIMI(NDIMI) , DIMO(NDIMO)
C=====
C=== ENTYP=1, FOR PBAR1; 2, FOR PBEAM1
C           IF (ENTYP.EQ.1) THEN
C               CALL MEVBRD(GRPID,TYPE,IDI,NID,IDO,NIDO,DIMI,NDIMI,DIMO,NDIMO
C +                   ,ERROR)
C           ELSE IF (ENTYP.EQ.2) THEN
C               CALL MEVBMD(GRPID,TYPE,IDI,NID,IDO,NIDO,DIMI,NDIMI,DIMO,NDIMO
C +                   ,ERROR)
C           END IF
C-----
C           RETURN
C           END

```

MEVBRD and MEVBMD Subroutines

MEVBRD and MEVBMD are the branched routines for the various sections, and convert the section dimensions to section properties for Bar and Beam elements. You may rename these routines as you like or move the function of these routines to BSBRPD. These routines call the BRXXPD routines where XX is the two-letter keyword for various section types. The MEVBRD routine for the MSC standard library is given in [Listing 3-2](#).

Listing 3-2 MEVBRD Subroutine

```

SUBROUTINE MEVBRD( GRPID, TYPE, ID, NID, IDO, NIDO, DIMI, NDIMI, DIMO,
+                NDIMO, ERROR)

C=====
C Purpose
C           Call the default type subroutine to convert PBAR1 to PBAR
C
C Arguments
C
C GRPID input   int       ID of group
C TYPE  input   char      Type of cross section
C ID    input   int       Array of values PID, MID contained in PBAR1
C                               entries
C NID   input   int       Size of ID array, NID=2 for PBAR1 entry
C IDO   output  int       Array of integer values contained in PBAR
C                               entries
C NIDO  output  int       Size of IDO array, NIDO=2 for PBAR entry
C DIMI  input   flt       Dimension values of cross section
C NDIMI input   int       Size of DIMI array
C DIMO  output  flt       Properties of cross section
C NDIMO output  flt       Size of DIMO array
C ERROR output  int       Type of error
C
C Method
C           Call the subroutine with respect to the section type
C
C Called by
C           BSRBPD
C
C CALLS
C           BRRDPD, BRTUPD, BRBRPD, BRBXP, BRIIPD, BRTTPD, BRLLPD, BRCHPD
C-----
      IMPLICIT INTEGER (I-N)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

C Calling sequence arguments
      INTEGER          ERROR, GRPID, ID(NID), IDO(NIDO)
      CHARACTER*8      TYPE
      DOUBLE PRECISION DIMI(NDIMI), DIMO(NDIMO)

C=====

C Clear the output array before usage
      CALL ZEROD ( DIMO, NDIMO )
      CALL ZEROI ( IDO, NIDO )
      ERROR = 0

C
      IF ( TYPE.EQ.'ROD      ' ) THEN
          CALL BRRDPD( ID, NID, IDO, NIDO, DIMI, NDIMI, DIMO, NDIMO, ERROR)
      ELSEIF( TYPE.EQ.'TUBE  ' ) THEN
          CALL BRTUPD( ID, NID, IDO, NIDO, DIMI, NDIMI, DIMO, NDIMO, ERROR)
      ELSEIF( TYPE.EQ.'BAR   ' ) THEN

```

```

        CALL BRBRPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'BOX      ') THEN
        CALL BRBXP( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'I        ') THEN
        CALL BRIIPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'T        ') THEN
        CALL BRTTPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'L        ') THEN
        CALL BRLLPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'CHAN     ') THEN
        CALL BRCHPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'CROSS    ') THEN
        CALL BRCRPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'H        ') THEN
        CALL BRHHPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'T1       ') THEN
        CALL BRT1PD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'I1       ') THEN
        CALL BRI1PD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'CHAN1    ') THEN
        CALL BRC1PD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'Z        ') THEN
        CALL BRZZPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'CHAN2    ') THEN
        CALL BRC2PD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'T2       ') THEN
        CALL BRT2PD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'BOX1     ') THEN
        CALL BRB1PD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'HEXA     ') THEN
        CALL BRHXP( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSEIF( TYPE.EQ. 'HAT      ') THEN
        CALL BRHTPD( ID,NID,IDO,NIDO, DIMI,NDIMI, DIMO,NDIMO, ERROR)
ELSE
        ERROR = 5150
END IF
C-----
RETURN
END

```

BRXXPD and BMXXPD Subroutines

The purpose of the BRXXPD and BMXXPD routines is to calculate the properties from the section dimensions. For each cross section, subroutines are required to convert the images of PBARL and PBEAML records to the images of PBAR and PBEAM records in the EPT datablock.

BRTUPD Subroutine

BRTUPD is an example routine that shows how to convert PBARL EPT record to PBAR EPT record for the Tube section. First, the details of the PBARL and the PBAR record are shown, and then the routine itself is given.

PBARL Record

The PBARL record in the EPT datablock is derived from the PBARL Bulk Data entry and is given below.

Table 3-2 PBARL (9102, 91, 52)

Word	Name	Type	Description
1	PID	I	Property identification number.
2	MID	I	Material identification number.
3	Group	Char	Group Name.
4	Group	Char	Group Name.
5	TYPE	Char4	Cross-section Type.
6	TYPE	Char4	Cross-section Type.
7	Dim1	RS	Dimension 1.
8	Dim2	RS	Dimension 2.
n+7-1	Dim n	RS	Dimension n (note that the final dimension is the nonstructural mass).
n+7	Flag	I	-1. Flag indicating end of cross-section dimensions.

PBAR Record

The PBAR record in the EPT datablock is derived from the PBAR Bulk Data entry and consists of 19 words. It is a replica of the Bulk Data entry, starting with PID field. The word 8 in the record is set to 0.0 since the field 9 in the first line of the PBAR Bulk Data entry is not used. The details of the PBAR record are given in [Table 3-3](#).

Table 3-3 PBAR (52, 20, 181)

Word	Name	Type	Description
1	PID	I	Property identification number.
2	MID	I	Material identification number.
3	A	RS	Area of cross-section.
4	I1	RS	Area moment of inertia for bending in plane 1.
5	I2	RS	Area moment of inertia for bending in plane 2.
6	J	RS	Torsional constant.

Table 3-3 PBAR (52, 20, 181) (continued)

Word	Name	Type	Description
7	NSM	RS	Nonstructural mass per unit length.
8	FE	RS	Not used. Set to 0.0.
9	C1	RS	Stress recovery location.
10	C2	RS	Stress recovery location.
11	D1	RS	Stress recovery location.
12	D2	RS	Stress recovery location.
13	E1	RS	Stress recovery location.
14	E2	RS	Stress recovery location.
15	F1	RS	Stress recovery location.
16	F2	RS	Stress recovery location.
17	K1	RS	Area factor of shear for plane 1.
18	K2	RS	Area factor of shear for plane 2.
19	I12	RS	Area product of inertia.

Listing 3-3 BRTUPD Subroutine

```

SUBROUTINE BRTUPD( ID , NID , IDO , NIDO , DIMI , NDIMI , DIMO , NDIMO , ERROR )
C=====
C Purpose:
C Convert PBAR1(entity type : TUBE) to PBAR
C
C Arguments:
C
C DIMI input flt Array of dimension values for cross-section
C (SEE FIG. 5 IN MEMO SSW-25, REV. 4, DATE 8/16/94)
C NDIM input int Size of DIMI array
C DIMO output flt Array of property values for cross-section
C NDIMO output int Size of DIMO array
C ERROR output int Type of error
C
C DISCRIPTION FOR DIMO ARRAY:
C DIMO ( 1 ) = A
C DIMO ( 2 ) = I1
C DIMO ( 3 ) = I2
C DIMO ( 4 ) = J
C DIMO ( 5 ) = NSM
C DIMO ( 6 ) = FE
C DIMO ( 7 ) = C1

```

```
C      DIMO ( 8 ) = C2
C      DIMO ( 9 ) = D1
C      DIMO (10) = D2
C      DIMO (11) = E1
C      DIMO (12) = E2
C      DIMO (13) = F1
C      DIMO (14) = F2
C      DIMO (15) = K1
C      DIMO (16) = K2
C      DIMO (17) = I12
C
C      Method
C
C      Called by:
C          MEVBRD
C-----
C      IMPLICIT INTEGER ( I-N )
C      IMPLICIT DOUBLE PRECISION ( A-H,O-Z )
C
C      Calling sequence arguments
C      INTEGER          ID(NID) , IDO(NIDO) , ERROR
C      DOUBLE PRECISION DIMI(NDIMI) , DIMO(NDIMO)
C=====
C
C=== WRITE THE PART OF INTEGER
C      DO 30 II = 1,NID
C          IDO(II) = ID(II)
C 30 CONTINUE
C
C      DIM1 = DIMI(1)
C      DIM2 = DIMI(2)
C      DIMO(1) = PI*(DIM1*DIM1-DIM2*DIM2)
C      DIMO(2) = PI*(DIM1**4-DIM2**4)/4.D0
C      DIMO(3) = DIMO(2)
C      DIMO(4) = PI*(DIM1**4-DIM2**4)/2.D0
C      DIMO(5) = DIMI(3)
C      DIMO(6) = 0.D0
C      DIMO(7) = DIM1
C      DIMO(8) = 0.D0
C      DIMO(9) = 0.D0
C      DIMO(10) = DIM1
C      DIMO(11) = -DIM1
C      DIMO(12) = 0.D0
C      DIMO(13) = 0.D0
C      DIMO(14) = -DIM1
C      DIMO(15) = 0.5D0
C      DIMO(16) = 0.5D0
C      DIMO(17) = 0.D0
C
C      IF ( DIMI(1).LE.DIMI(2) ) ERROR = 5102
C-----
C      RETURN
C      END
```

BMTUPD Subroutine

BMTUPD is an example routine that shows how to convert PBEAML EPT record to PBEAM EPT record for the Tube section. First, the details of the PBEAML and the PBEAM records are shown, and then the routine itself is given.

PBEAML Record

The PBEAML record in the EPT datablock is a derived from the PBEAML Bulk Data entry and is given in [Table 3-4](#).

Table 3-4 PBEAML (9202, 92, 53)

Word	Name	Type	Description
1	PID	I	Property ID.
2	MID	I	Material ID.
3	Group	Char	Group Name.
4	Group	Char	Group Name.
5	TYPE	Char4	Cross-section Type.
6	TYPE	Char4	Cross-section Type.
7	SO	RS	Stress output request flag, 1=yes,=no.
8	XXB	RS	X/XB - parametric location of the station.
9	Dim1	RS	Dimension 1.
10	Dim2	RS	Dimension 2.
n+9-1	Dim n	RS	Dimension n (note that the final dimension is the nonstructural mass).
n+9	Flag	I	-1. Flag indicating end of cross-section dimensions.
Words 7 through ndim+11 repeat 11 times.			

PBEAM Record

The PBEAM record in EPT datablock consists of 197 words. The first five words and the last 16 words are common to all the 11 stations. Each of the 11 stations have their own 21 unique words. The details of the PBEAM record are given in [Table 3-5](#).

Table 3-5 PBEAM (5402, 54, 262)

Word	Name	Type	Description
1	PID	I	Property identification number.
2	MID	I	Material identification number.
3	N	I	Number of intermediate stations.
4	CCF	I	Constant cross-section flag. 1 = constant, 2 = variable.
5	X	RS	Unused.
6	SO	RS	Stress output request. 1.0 = yes, 0.0 = no.
7	XXB	RS	Parametric location of the station. Varies between 0. and 1.0.
8	A	RS	Area.
9	I1	RS	Moment of inertia for bending in plane 1.
10	I2	RS	Moment of inertia for bending in plane 2.
11	I12	RS	Area product of inertia.
12	J	RS	Torsional constant.
13	NSM	RS	Nonstructural mass.
14	C1	RS	Stress recovery location.
15	C2	RS	Stress recovery location.
16	D1	RS	Stress recovery location.
17	D2	RS	Stress recovery location.
18	E1	RS	Stress recovery location.
19	E2	RS	Stress recovery location.
20	F1	RS	Stress recovery location.
21	F2	RS	Stress recovery location.
Words 6 through 21 repeat 11 times.			
182	K1	RS	Area factor for shear for plane 1.
183	K2	RS	Area factor for shear for plane 2.
184	S1	RS	Shear-relief coefficient for plane 1.
185	S2	RS	Shear-relief coefficient for plane 2.

Table 3-5 PBEAM (5402, 54, 262) (continued)

Word	Name	Type	Description
186	NSIA	RS	Nonstructural mass moment of inertia at end A.
187	NSIB	RS	Nonstructural mass moment of inertia at end B.
188	CWA	RS	Warping coefficient for end A.
189	CWB	RS	Warping coefficient for end B.
190	M1A	RS	Y-coordinate of center of gravity for nonstructural mass at end A.
191	M2A	RS	Z-coordinate of center of gravity for nonstructural mass at end A.
192	M1B	RS	Y-coordinate of center of gravity for nonstructural mass at end B.
193	M2B	RS	Z-coordinate of center of gravity for nonstructural mass at end B.
194	N1A	RS	Y-coordinate for neutral axis at end A.
195	N2A	RS	Z-coordinate for neutral axis at end A.
196	N1B	RS	Y-coordinate for neutral axis at end B.
197	N2B	RS	Z-coordinate for neutral axis at end B.

Listing 3-4 BMTUPD Subroutine

```

SUBROUTINE BMTUPD( ID ,NID , IDO ,NIDO , DIMI ,NDIMI , DIMO ,NDIMO , ERROR )
C=====
C  Purpose
C    Convert PBEAM1(entity type : TUBE) to PBEAM
C
C  Arguments
C
C  ID    input  int  Contain the integer information PID, MID
C  NID   input  int  Size of ID array, NID = 2
C  DIMI  input  flt  Dimension values of cross section
C          ( See FIG.5 IN MEMO SSW-25, REV. 4, DATE 8/16/94)
C  NDIMI input  int  Size of DIMI array
C  IDO   output int  Contain the integer information PID,MID,N,CCF
C  NIDO  output int  Size of IDO array, NIDO = 4
C  DIMO  output flt  Properties of cross section
C  NDIMO output int  Size of DIMO array
C  ERROR output int  Type of error
C
C  Description for DIMO array

```

```
C      DIMO (1)  = X
C      DIMO (2)  = SO
C      DIMO (3)  = XXB
C      DIMO (4)  = A
C      DIMO (5)  = I1
C      DIMO (6)  = I2
C      DIMO (7)  = I12
C      DIMO (8)  = J
C      DIMO (9)  = NSM
C      DIMO (10) = C1
C      DIMO (11) = C2
C      DIMO (12) = D1
C      DIMO (13) = D2
C      DIMO (14) = E1
C      DIMO (15) = E2
C      DIMO (16) = F1
C      DIMO (17) = F2
C      DIMO(2) thru DIMO(17) repeat 11 times
C      DIMO (178) = K1
C      DIMO (179) = K2
C      DIMO (180) = S1
C      DIMO (181) = S2
C      DIMO (182) = NSIA
C      DIMO (183) = NSIB
C      DIMO (184) = CWA
C      DIMO (185) = CWB
C      DIMO (186) = M1A
C      DIMO (187) = M2A
C      DIMO (188) = M1B
C      DIMO (189) = M2B
C      DIMO (190) = N1A
C      DIMO (191) = N2A
C      DIMO (192) = N1B
C      DIMO (193) = N2B
C
C      Method
C          Simply calculate the properties and locate that data
C          to the image of PBEAM entries
C      Called by
C          MEVBMD
C-----
C      IMPLICIT INTEGER (I-N)
C      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C      Calling sequence arguments
C      INTEGER          ERROR, ID(NID), IDO(NIDO)
C      DOUBLE PRECISION DIMI(NDIMI), DIMO(NDIMO)
C
C      Local variables
C      INTEGER NAM(2)
C
C      NASTRAN common blocks
C      COMMON /CONDAD/PI
```

```

C      Local data
      DATA NAM/4HBMTU,4HPD /
C-----
C
C=== WRITE THE PART OF INTEGER
      DO 30 II = 1,NID
          IDO(II) = ID(II)
      30  CONTINUE
C-----
C=== DETECT HOW MANY STATION , CONSTANT OR LINEAR BEAM.
C-----
      ISTATC = NDIMI/11
      DO 35 II = 0,10
          NW = II*ISTATC
          IF (DIMI(3+NW).EQ.0.D0) THEN
              IDO(3) = II-2
              IDO(4) = 1
              IF (DIMI(3).NE.DIMI(3+NW-ISTATC)) IDO(4)=2
              GO TO 40
          END IF
      35  CONTINUE
      40  DIMO(1) = 0.D0
          DO 100 L1 = 0,10
              LC = 16*L1
              NW = L1*ISTATC
              IF (DIMI(3+NW+ISTATC).EQ.0.D0) LC = 160
              DIM1 = DIMI(3+NW)
              DIM2 = DIMI(4+NW)

              IF ( DIM1.LE.DIM2 ) ERROR = 5102

              DIMO(2+LC) = DIMI(1+NW)
              DIMO(3+LC) = DIMI(2+NW)
              DIMO(4+LC) = PI*(DIM1*DIM1-DIM2*DIM2)
              DIMO(5+LC) = PI*(DIM1**4-DIM2**4)/4.D0
              DIMO(6+LC) = DIMO(5+LC)
              DIMO(7+LC) = 0.D0
              DIMO(8+LC) = PI*(DIM1**4-DIM2**4)/2.D0
              DIMO(9+LC) = DIMI(5+NW)
              DIMO(10+LC) = DIM1
              DIMO(11+LC) = 0.D0
              DIMO(12+LC) = 0.D0
              DIMO(13+LC) = DIM1
              DIMO(14+LC) = -DIM1
              DIMO(15+LC) = 0.D0
              DIMO(16+LC) = 0.D0
              DIMO(17+LC) = -DIM1
              IF (LC.EQ.160) GO TO 110
          100 CONTINUE
          110 DIMO(178) = 0.5D0
              DIMO(179) = 0.5D0
C-----
      300 RETURN
      END

```


BSGRQ Subroutine

For optimization of PBARL entries, you need to provide overall information such as number of cross sections in your library and the maximum number of fields in the continuation lines. BSGRQ provides the information and is required only if you wish to perform sensitivity or optimization with the section dimensions. The calling sequence and example routine is given in [Listing 3-5](#).

Listing 3-5 BSGRQ Routine

```

SUBROUTINE BSGRQ (GRPID, NSECT, NDIMAX, ERROR)
C
C -----
C   PURPOSE:
C     PROVIDE OVERALL CHARACTERISTICS OF A BEAM SECTION LIBRARY
C -----
C   ARGUMENTS:
C
C     GRPID  input  integer  Group name.  Not used, reserved for future
C                                     use.
C     NSECT  output integer  Number of different section types
C     NDIMAX output integer  Maximum number of dimension for any
C                                     section type
C     ERROR  output integer  Indicates if an error has occurred. The
C                                     code returned indicates the type of error
C -----
C   CALLED BY:
C     BCGRQ routine
C -----
C
C===  Set number of section in the library
C      nsect = 19
C===  Set the maximum number of DIMi fields (including 1 for nonstructured
C      mass) required by any one section
C      ndimax = 7
C -----
C
C     RETURN
C     END

```

Note that the arguments GRPID and ERROR are not used. GRPID is reserved for future use. You may use the ERROR argument to send an error code which could later be used to print an error message.

BSBRT Subroutine

The BSBRT routine provides the name, number of fields in the continuation line in the PBARL Bulk Data entry, and the number of constraints for each section in the library. As an example, the name of the tube section, shown in [Figure 3-20](#), in the MSC standard library is “TUBE”, the number of dimensions for the tube section is three (OUTER RADIUS, INNER RADIUS, and NSM), and there is one physical constraint.

The physical constraint is that the inner radius (DIM2) can not be greater than outer radius (DIM1). It is necessary to specify the constraints so that the optimization of the section dimension in SOL200 does not result into an inconsistent shape.

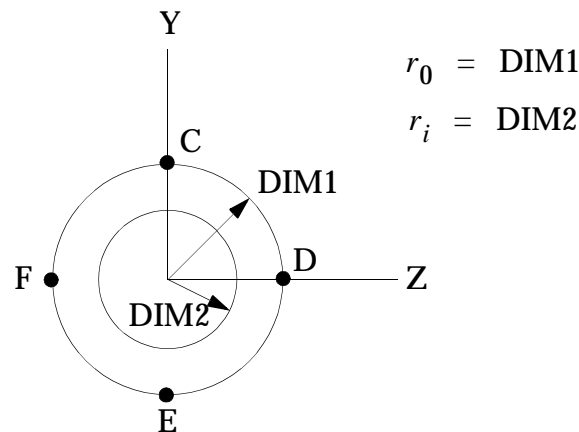


Figure 3-20 Tube Section

This routine is called during optimization process. The calling sequence and example routine is given in [Listing 3-6](#).

Listing 3-6 BSBRT Subroutine

```

SUBROUTINE BSBRT( GRPID, ENTYP, TYPE, NDIM, NCONST, NSECT, ERROR)
C =====
C Purpose
C     Get the name, number of fields in the continuation line and
C     number of constraints
C     to be used by optimization routines
C
C Arguments
C
C     GRPID  input  integer      The ID of group name
C     ENTYP  input  integer      1: PBARL, 2: PBEAML
C     TYPE   output character*8  Arrays for cross section types
C     NDIM   output integer      Number of dimensions for isect
C                                     section type, including 1 for
C                                     nonstructural mass
C     NCONST output integer      Number of dimensional constraints
C                                     imposed by isect section type
C     NSECT  input  integer      Number of sections
C     ERROR  output integer      Error code
C
C Called by
C     BCBRT routine
C -----
C=== Argument Type Declaration
      INTEGER      GRPID, ENTYP, NDIM( NSECT ), NCONST( NSECT )
      CHARACTER*8  TYPE( NSECT )

```

```
C=== Local variables
      INTEGER NAM(2)
C =====
C=== Currently, only PBARL is supported for optimization. Based on
C=== ENTYP the library may have different number of DIMi fields and
C=== number of constraints. Currently, GRPID and ENTYP are not being
C=== used. So just set them to default values, even though they are
C=== input type.
      GRPID = 1
      ENTYP = 1

C -----
C=== Set the name, number of fields in the continuation line for the PBARL
entry,
C=== and number of constraints in the TYPE, DIMI and NCONST arrays,
      respectively .
C=== Note that the value in DIMI is one more (for NSM field) than DIMI
fields
C=== Make sure names are all capitals, even if they are lower case in the
input data file.
      TYPE(1) = 'ROD'
      NDIM(1) = 2
      NCONST(1) = 0
C
      TYPE(2) = 'TUBE'
      NDIM(2) = 3
      NCONST(2) = 1
C
      TYPE(3) = 'BAR'
      NDIM(3) = 3
      NCONST(3) = 0
C
      TYPE(4) = 'BOX'
      NDIM(4) = 5
      NCONST(4) = 2
C
      TYPE(5) = 'I'
      NDIM(5) = 7
      NCONST(5) = 3
C
      TYPE(6) = 'T'
      NDIM(6) = 5
      NCONST(6) = 2
C
      TYPE(7) = 'L'
      NDIM(7) = 5
      NCONST(7) = 2
C
      TYPE(8) = 'CHAN'
      NDIM(8) = 5
      NCONST(8) = 2
C
      TYPE(9) = 'CROSS'
      NDIM(9) = 5
```

```

      NCONST(9) = 1
C
      TYPE(10) = 'H'
      NDIM(10) = 5
      NCONST(10) = 1
C
      TYPE(11) = 'T1'
      NDIM(11) = 5
      NCONST(11) = 1
C
      TYPE(12) = 'I1'
      NDIM(12) = 5
      NCONST(12) = 1
C
      TYPE(13) = 'CHAN1'
      NDIM(13) = 5
      NCONST(13) = 1
C
      TYPE(14) = 'Z'
      NDIM(14) = 5
      NCONST(14) = 1
C
      TYPE(15) = 'CHAN2'
      NDIM(15) = 5
      NCONST(15) = 2
C
      TYPE(16) = 'T2'
      NDIM(16) = 5
      NCONST(16) = 2
C
      TYPE(17) = 'BOX1'
      NDIM(17) = 7
      NCONST(17) = 2
C
      TYPE(18) = 'HEXA'
      NDIM(18) = 4
      NCONST(18) = 1
C
      TYPE(19) = 'HAT'
      NDIM(19) = 5
      NCONST(19) = 2

C
C =====
      RETURN
      END

```

BSBRID Subroutine

The BSBRID subroutine is required if optimization is to be performed. Its function is to provide information required in the calculation of the sensitivities (gradients) of the bar properties with respect to the bar dimensions.

Two basic types of information are provided. The first is the SENTYP array, which indicates how each section property varies as a function of each dimension. Values in the SENTYP array can be either: 0 for no variation; 1 for a linear variation; 2 for a nonlinear variation; or 3 for an unknown variation. The SENTYP = 3 option is to be used when you know that the property is a function of the design dimension, but analytical gradient information is not being provided using the BSBRGD subroutine. In this case, MSC.Nastran will calculate the gradients for you using central differencing techniques.

The second piece of information is the ALIN array. This array provides any linear sensitivity data. For example, the C1 stress recovery location for the TUBE section is $1.0 \cdot \text{DIM1}$ so that this sensitivity of this stress recovery point with respect to the first dimension is 1.0.

You may use your own naming convention for the subroutines that specify the section sensitivity data. The calling tree used for the MSC standard library is shown in [Figure 3-21](#).

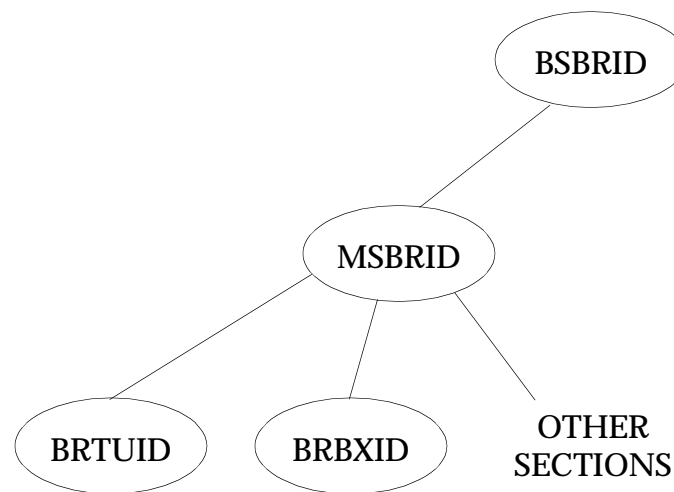


Figure 3-21 Calling Tree to Generate Sensitivity Data

BSBRID calls the bar evaluator routine MSBRID/S for the BAR element while the corresponding beam evaluator routine is absent since the PBEAM1 dimensions cannot be designed. The evaluator in turn calls the routines for each section. The routines are named BRXXID, where XX is a two-letter identifier for the section. For example, the routine for the TUBE section is called BRTUID. The details for various routines are given in [Listing 3-7](#).

Listing 3-7 BSBRID Subroutine

```

SUBROUTINE BSBRID(GRPID,ENTYP,SECTON,SENTYP,ALIN,NDIM,NPROP,
1          ERROR)

C
C =====
C Purpose
C   set up section dependent information for a particular cross
C   section type
C
C Arguments
C
C   GRPID  input  integer      ID of the group
C   ENTYP  input  integer      1: PBAR1, 2: PBEAM1
C   SECTON input  character*8   Section type
C   SENTYP output integer      Type of sensitivity, 0: invariant,
C                               1: linear, 2: nonlinear, 3: calculated by
C                               finite difference
C   ALIN   output double       Matrix providing the linear
C                               factors for sensitive relationships
C   NDIM   input  integer      Number of dimensions
C   NPROP  input  integer      Number of properties
C   ERROR  output integer      Type of error
C
C Method
C   Simply transfer control based on entry type
C
C Called by
C   BCBRID
C
C Calls
C   MSBRID, MSBMID
C -----
C   IMPLICIT INTEGER (I-N)
C   IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C   Calling sequence arguments
C   INTEGER          ENTYP,ERROR,GRPID,SENTYP(NPROP,NDIM)
C   CHARACTER*8     SECTON
C   DOUBLE PRECISION ALIN(NPROP,NDIM)
C =====
C   GRPID = 1
C   IF(ENTYP.EQ.1) THEN
C     CALL MSBRID(SECTON,SENTYP,ALIN,NDIM,NPROP,ERROR)
C   ELSE
C     ERROR = 5400
C   END IF
C
C   RETURN
C   END

```

MSBRID Subroutine

MSBRID is a branched routine for providing information on the calculation of sensitivities for each of the bar types. You may rename this routine as you like or move its function to BSBRID. MSBRID calls the BRXXID routines, where XX is the two-letter keyword for various section types. The routine MSBRID for the MSC standard library is given in [Listing 3-8](#).

Listing 3-8 MSBRID Subroutine

```

SUBROUTINE MSBRID( SECTON , SENTYP , ALIN , NDIM , NPROP , ERROR )
C
C =====
C Purpose
C   To set up section dependent information for PBAR1 cross section
C
C Arguments
C
C   SECTON input  character*8 Name of section type
C   SENTYP output integer      Type of sensitivity, 0: invariant,
C                               1: linear, 2: nonlinear
C   ALIN    output double      Matrix providing the linear factors
C                               for sensitive relationships
C   NDIM   input  integer      No. of dimensions
C   NPROP  input  integer      No. of properties in EPT datablock
C   ERROR  output integer      Type of error
C
C Method
C   Simply transfer the section dependent information of
C   the 19 kinds
C Called by
C   BSBRID
C
C Calls
C   BRRDID, BRBRID, BRTUID, BRBXID, BRIIID, BRLLID, BRTTID,
C   BRCHID, BRCRID, BRHHID, BRT1ID, BRI1ID, BRC1ID, BRZZID,
C   BRC2ID, BRT2ID, BRB1ID, BRHXID, BRHTID
C   ZEROI, ZEROD (Nastran utility)
C -----
C
C IMPLICIT INTEGER (I-N)
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C Calling sequence arguments
C CHARACTER*8      SECTON
C INTEGER          ERROR, SENTYP(NPROP,NDIM)
C DOUBLE PRECISION ALIN(NPROP,NDIM)
C =====
C
C CALL ZEROI( SENTYP(1,1), NPROP*NDIM )
C CALL ZEROD( ALIN(1,1), NPROP*NDIM )
C ERROR = 0

```

C

```

IF(SECTON.EQ.'ROD') THEN
  CALL BRRDID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'BAR') THEN
  CALL BRBRID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'TUBE') THEN
  CALL BRTUID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'BOX') THEN
  CALL BRBXID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'I') THEN
  CALL BRIIID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'L') THEN
  CALL BRLLID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'T') THEN
  CALL BRTTID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'CHAN') THEN
  CALL BRCHID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'CROSS') THEN
  CALL BRCRID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'H') THEN
  CALL BRHHID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'T1') THEN
  CALL BRT1ID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'I1') THEN
  CALL BRI1ID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'CHAN1') THEN
  CALL BRC1ID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'Z') THEN
  CALL BRZZID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'CHAN2') THEN
  CALL BRC2ID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'T2') THEN
  CALL BRT2ID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'BOX1') THEN
  CALL BRB1ID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'HEXA') THEN
  CALL BRHXID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE IF(SECTON.EQ.'HAT') THEN
  CALL BRHTID(SENTYP,ALIN,NDIM,NPROP,ERROR)
ELSE
  ERROR = 5410
END IF

```

C

```

RETURN
END

```

BRTUID Subroutine

BRTUID is an example routine that shows how to define the sensitivity type of each of the bar properties for the tube and the subset of the sensitivities that are linear.

Listing 3-9 BRTUID Subroutine

```
      SUBROUTINE BRTUID(SENTYP,ALIN,NDIM,NPROP,ERROR)
C
C =====
C   Purpose
C     To set up section dependent information for rod cross section
C
C   Arguments
C
C     SENTYP output integer  Type of sensitivity, 0: invariant,
C                          1: linear, 2: nonlinear
C     ALIN   output double   Matrix providing the linear factors for
C                          sensitive relationships
C     NDIM   input  integer  No. of dimensions
C     NPROP  input  integer  No. of properties in EPT datablock
C     ERROR  output integer  Type of error
C
C   Method
C     Simply provides the information
C
C   Called by
C     MSBRID
C -----
C     IMPLICIT INTEGER (I-N)
C     IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C   Calling sequence arguments
C     INTEGER          ERROR,SENTYP(NPROP,NDIM)
C     DOUBLE PRECISION ALIN(NPROP,NDIM)
C
C   Local variables
C     INTEGER NAM(2)
C
C   Local data
C     DATA NAM/4HBRTU,4HID /
C =====
C
C     ALIN( 7,1) =  1.0D0
C     ALIN(10,1) =  1.0D0
C     ALIN(11,1) = -1.0D0
C     ALIN(14,1) = -1.0D0
C     ALIN( 5,3) =  1.0D0
C
C     SENTYP( 1,1) =  2
C     SENTYP( 1,2) =  2
C     SENTYP( 2,1) =  2
C     SENTYP( 2,2) =  2
C     SENTYP( 3,1) =  2
C     SENTYP( 3,2) =  2
C     SENTYP( 4,1) =  2
C     SENTYP( 4,2) =  2
C     SENTYP( 7,1) =  1
C     SENTYP(10,1) =  1
```

```

SENTYP(11,1) = 1
SENTYP(14,1) = 1
SENTYP( 5,3) = 1

```

C

```

RETURN
END

```

BSBRGD Subroutine

The BSBRGD subroutine is required if optimization is to be performed and analytical sensitivities are needed (SENTYP = 2 in subroutine BSBRID). Its function is to provide the nonlinear gradients of the bar properties with respect to the bar dimensions. You may use your own naming convention for the subroutines that calculate the section gradients from the dimensions. The calling tree used for the MSC standard library is shown in [Figure 3-22](#).

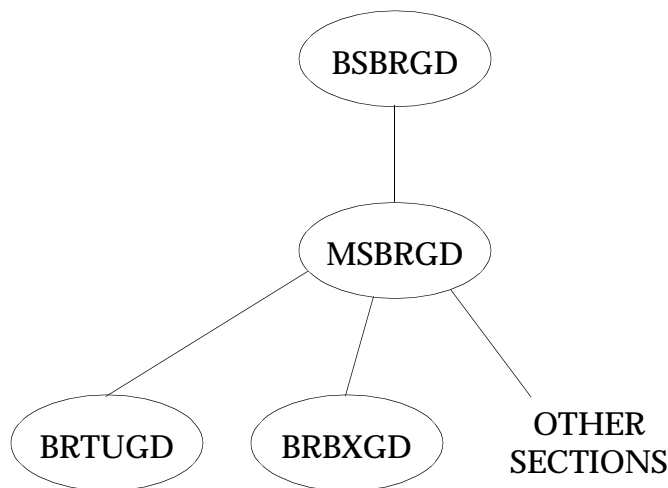


Figure 3-22 Calling Tree to Generate Nonlinear Sensitivity Data

BSBRGD calls the bar evaluator routine MSBRGD/S for the BAR element. The evaluator, in turn, calls the routines for each section. The routines are named BRXXGD, where XX is a two-letter identifier for the section. For example, the routine for the TUBE section is called BRTUGD. The details for various routines are given in [Listing 3-10](#).

Listing 3-10 BSBRGD Subroutine

```

SUBROUTINE BSBRGD(GRPID, ENTYP, TYPE, DIMI, NDIMI, ANONL, NPROP, ERROR)
C
C =====
C Purpose
C To get the nonlinear factors of sensitivities for default sections
C

```

```

C Arguments
C
C     GRPID input integer      ID of group name
C     ENTYP input integer      1: PBAR1, 2: PBEAM1
C     TYPE  input character*8  Type name of cross-section
C     DIMI  input double       Array from EPT record
C     NDIMI input integer      Number of dimensions (Plus NSM)
C     ANONL output double      Array providing the nonlinear factors
C                               for sensitivity relationships
C     NPROP input integer      Number of properties in PBAR entries
C     ERROR output integer     Type of error
C
C Method
C     Simply transfer control based on entry types
C Called by
C     BCBRGD
C
C Calls
C     MSBRGD
C -----
C     IMPLICIT INTEGER (I-N)
C     IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C Calling sequence arguments
C     INTEGER          GRPID, ENTYP, NDIMI, NPROP, ERROR
C     CHARACTER*8     TYPE
C     DOUBLE PRECISION ANONL(NPROP, NDIMI), DIMI(NDIMI)
C
C Local variables
C     INTEGER NAM(2)
C
C Local data
C     DATA NAM/4HBSBR, 4HGD /
C =====
C
C     GRPID = 1
C     IF(ENTYP .EQ. 1) THEN
C         CALL MSBRGD(TYPE, DIMI, NDIMI, ANONL, NPROP, ERROR)
C     ELSE
C         ERROR = 5300
C     END IF
C -----
C     RETURN
C     END

```

MSBRGD Subroutine

MSBRGD is a branched routine for providing information on the calculation of nonlinear gradients for each of the bar types. You may rename this routine as you like or move its function to BSBRGD. MSBRGD calls the BRXXGD routines, where XX is the two-letter keyword for various section types. The routine MSBRGD for the MSC standard library is given in [Listing 3-11](#).

Listing 3-11 MSBRGD Subroutine

```

SUBROUTINE MSBRGD(TYPE, DIMI, NDIMI, ANONL, NPROP, ERROR)
C
C =====
C Purpose
C   To get the nonlinear factors of sensitivities for PBAR1 entries
C
C Arguments
C
C   TYPE  input  character*8  Type name of cross-section
C   DIMI  input  double       Array from EPT record
C   NDIMI input  integer      Number of dimensions (Plus NSM)
C   ANONL output double      Array providing the nonlinear factors
C                               for sensitivity relationships
C   NPROP input  integer      Number of properties in PBAR entries
C   ERROR output integer      Type of error
C
C Method
C   Simply transfer information based on cross-section type
C Called by
C   BSBRGD
C
C Calls
C   BRRDGD, BRBRGD, BRBXGD, BRTUGD, BRIIGD, BRTTGD, BRLIGD, BRCHGD
C   BRCRGD, BRHHGD, BRT1GD, BRT2GD, BRI1GD, BRC1GD, BRC2GD, BRZZGD
C   BRHXGD, BRB1GD, BRHTGD, ZEROD(Nastran utility)
C -----
C   IMPLICIT INTEGER (I-N)
C   IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C Calling sequence arguments
C   INTEGER          NDIMI, NPROP, ERROR
C   CHARACTER*8     TYPE
C   DOUBLE PRECISION ANONL(NPROP, NDIMI), DIMI(NDIMI)
C =====

CALL ZEROD( ANONL(1,1), NPROP*NDIMI )
ERROR = 0

IF(TYPE.EQ.'ROD') THEN
    CALL BRRDGD(DIMI, NDIMI, ANONL, NPROP, ERROR)
ELSE IF(TYPE.EQ.'BAR') THEN
    CALL BRBRGD(DIMI, NDIMI, ANONL, NPROP, ERROR)
ELSE IF(TYPE.EQ.'BOX') THEN
    CALL BRBXGD(DIMI, NDIMI, ANONL, NPROP, ERROR)
ELSE IF(TYPE.EQ.'TUBE') THEN
    CALL BRTUGD(DIMI, NDIMI, ANONL, NPROP, ERROR)
ELSE IF(TYPE.EQ.'I') THEN
    CALL BRIIGD(DIMI, NDIMI, ANONL, NPROP, ERROR)
ELSE IF(TYPE.EQ.'L') THEN
    CALL BRLIGD(DIMI, NDIMI, ANONL, NPROP, ERROR)
ELSE IF(TYPE.EQ.'T') THEN
    CALL BRTTGD(DIMI, NDIMI, ANONL, NPROP, ERROR)

```

```

ELSE IF(TYPE.EQ.'CHAN') THEN
    CALL BRCHGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'CROSS') THEN
    CALL BRCRGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'H') THEN
    CALL BRHHGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'T1') THEN
    CALL BRT1GD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'I1') THEN
    CALL BRI1GD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'CHAN1') THEN
    CALL BRC1GD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'Z') THEN
    CALL BRZZGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'CHAN2') THEN
    CALL BRC2GD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'T2') THEN
    CALL BRT2GD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'BOX1') THEN
    CALL BRB1GD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'HEXA') THEN
    CALL BRHXGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE IF(TYPE.EQ.'HAT') THEN
    CALL BRHTGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
ELSE
    ERROR = 5310
ENDIF

C
RETURN
END

```

BRTUGD Subroutine

BRTUGD is an example routine that shows how to define the nonlinear gradients of the TUBE section as a function of the outer and inner radius of the tube.

Listing 3-12 BRTUGD Subroutine

```

SUBROUTINE BRTUGD(DIMI,NDIMI,ANONL,NPROP,ERROR)
C
C =====
C Purpose
C To get the nonlinear factors of sensitivities for TUBE section
C
C Arguments
C
C DIMI input double Array of EPT records (Dimi+NSM)
C NDIMI input integer Number of dimensions (Plus NSM)
C ANONL output double Array providing the nonlinear factors
C for sensitivity relationships
C NPROP input integer Number of properties in PBAR entries
C ERROR output integer Type of error

```

```

C
C   Method
C           Simply calculates the nonlinear factors
C   Called by
C           MSBRGD
C -----
C           IMPLICIT INTEGER (I-N)
C           IMPLICIT DOUBLE PRECISION (A-H,O-Z)

C   Calling sequence arguments
C           INTEGER           NDIMI,NPROP,ERROR
C           DOUBLE PRECISION ANONL(NPROP,NDIMI), DIMI(NDIMI)

C   Nastran common blocks
C           COMMON /CONDAD/ PI
C =====

C           DIM1           = DIMI(1)
C           DIM2           = DIMI(2)
C           PDIM13         = PI*DIM1**3
C           PDIM23         = PI*DIM2**3

C           ANONL(1,1) = 2*PI*DIM1
C           ANONL(1,2) = -2*PI*DIM2
C           ANONL(2,1) = PDIM13
C           ANONL(2,2) = -PDIM23
C           ANONL(3,1) = PDIM13
C           ANONL(3,2) = -PDIM23
C           ANONL(4,1) = 2*PDIM13
C           ANONL(4,2) = -2*PDIM23

C
C           RETURN
C           END

```

BSBRCD Subroutine

The BSBRCD subroutine allows you to place constraints on values the beam dimensions can take during a design task. It is not needed unless optimization is used and, even then, is available only to impose conditions on the dimensions to keep the optimization process from selecting physically meaningless dimensions.

For example, the optimizer might select a TUBE design with the inner radius greater than the outer radius because this allows for a negative area and therefore a negative weight (something a weight minimization algorithm loves!) These constraints are not the same as the PMIN and PMAX property limits that are imposed on the DVPREL1 entry. Instead, these are constraints that occur between or among section dimensions. A DRESP2 entry could be used to develop the same design constraints, but the

subroutine reduces the burden on the user interface, the primary goal of the beam library project. The calling tree used for the MSC standard library, which is shown in [Figure 3-23](#).

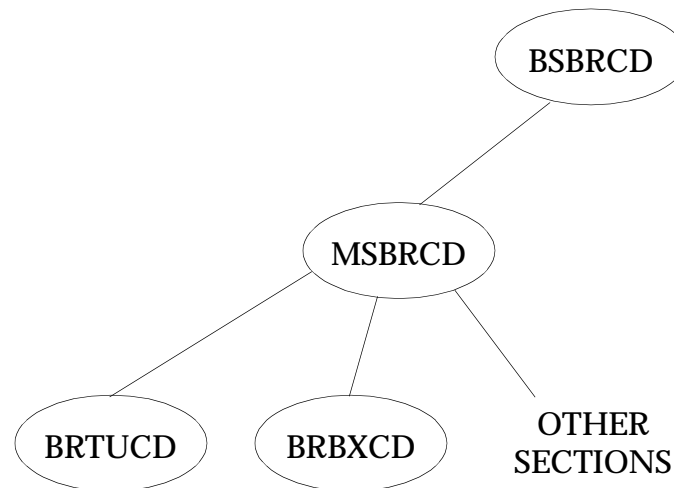


Figure 3-23 Calling Tree to Generate Beam Dimension Constraints

BSBRCD calls the bar evaluator routine MSBRCD. The evaluator in turn calls the routines for each section that requires constraints. The routines are named BRXXCD, where XX is a two-letter identifier for the section. For example the routine for the TUBE section is called BRTUCD. The details for various routines are given in [Listing 3-13](#).

Listing 3-13 BSBRCD Subroutine

```

SUBROUTINE BSBRCD(GRPID, ENTYP, TYPE, AFACT, NCONST, NDIMI, ERROR)
C
C =====
C Purpose
C     To get constraint information for default types
C
C Arguments
C
C     GRPID  input integer    ID of the group name
C     ENTYP  input integer    1: PBAR1, 2: PBEAM1
C     TYPE   input character*8 Section type
C     NCONST input integer    Number of constraints
C                                     for the section type
C     NDIMI  input integer    Number of dimensions
C                                     for the section type
C     AFACT  output double    The factor for the NDIMI dimension in
C                                     the constraint relation. Dimensions are
C                                     NCONST by NDIMI.
C     ERROR  output integer    type of error
C

```



```
C      ERROR  output integer      type of error
C
C      Method
C          Simply transfers control based on group name
C
C      Called by
C          BSBRCT
C
C      Calls
C          BRTUCD, BRBXCD, BRIICD, BRTTCD, BRLLCD, BRCHCD,
C          BRCRCD, BRHHCD, BRT1CD, BRI1CD, BRC1CD, BRZZCD,
C          BRC2CD, BRT2CD, BRB1CD, BRHXCD, BRHTCD, ZEROD
C      (Nastran utility)
C -----
C          IMPLICIT INTEGER (I-N)
C          IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C      Calling sequence arguments
C          CHARACTER*8          TYPE
C          INTEGER              NCONST,NDIMI,ERROR
C          DOUBLE PRECISION     AFACT(NCONST,NDIMI)
C =====
C
C          CALL ZEROD( AFACT(1,1), NCONST*NDIMI )
C          ERROR = 0
C
C          IF(TYPE.EQ.'TUBE') THEN
C              CALL BRTUCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'BOX') THEN
C              CALL BRBXCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'I') THEN
C              CALL BRIICD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'L') THEN
C              CALL BRLLCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'T') THEN
C              CALL BRTTCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'CHAN') THEN
C              CALL BRCHCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'CROSS') THEN
C              CALL BRCRCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'H') THEN
C              CALL BRHHCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'T1') THEN
C              CALL BRT1CD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'I1') THEN
C              CALL BRI1CD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'CHAN1') THEN
C              CALL BRC1CD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'Z') THEN
C              CALL BRZZCD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'CHAN2') THEN
C              CALL BRC2CD(AFACT,NCONST,NDIMI,ERROR)
C          ELSE IF(TYPE.EQ.'T2') THEN
```

```

        CALL BRT2CD(AFACT,NCONST,NDIMI,ERROR)
    ELSE IF(TYPE.EQ.'BOX1') THEN
        CALL BRB1CD(AFACT,NCONST,NDIMI,ERROR)
    ELSE IF(TYPE.EQ.'HEXA') THEN
        CALL BRHXCD(AFACT,NCONST,NDIMI,ERROR)
    ELSE IF(TYPE.EQ.'HAT') THEN
        CALL BRHTCD(AFACT,NCONST,NDIMI,ERROR)
    ELSE
        ERROR = 5500
    END IF

```

```

C
    RETURN
    END

```

BRTUCD Subroutine

BRTUCD is an example routine that shows how to define the constraints for a bar section. This example routine is for the TUBE section and imposes a single constraint that $-DIM1 + DIM2 < 0.0$, where DIM1 is the outer radius and DIM2 is the inner radius of the tube. The constraints should always be specified so that the specified linear combination of dimensions is less or equal to zero when the constraint is satisfied.

Listing 3-15 BRTUCD Subroutine

```

        SUBROUTINE BRTUCD(AFACT,NCONST,NDIMI,ERROR)
C
C =====
C Purpose
C     To get constraint information for TUBE type
C
C Arguments
C
C     AFACT   output double   The factor for the NDIMI dimension in the
C                                     constraint relation. Dimensions are NCONST
C                                     by NDIMI.
C     NCONST  input  integer   Number of constraints for the section type
C     NDIMI   input  integer   Number of dimensions for the section type
C     ERROR   output integer   type of error
C
C Method
C     Simply transfers constraint information
C
C Called by
C     MSBRCD
C -----
        IMPLICIT INTEGER (I-N)
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
C Calling sequence arguments
        INTEGER          NCONST,NDIMI,ERROR
        DOUBLE PRECISION AFACT(NCONST,NDIMI)

```

```

C =====
C
C     IF(NCONST.NE.1) THEN
C         ERROR = 5502
C         RETURN
C     END IF
C
C     AFACT(1,1) = -1.0D0
C     AFACT(1,2) =  1.0D0
C
C
C     RETURN
C     END

```

BSMSG Subroutine

The Error handling is performed by a subroutine called `bsmsg.f`. This routine has the following parameters:

```

SUBROUTINE BSMSG(GRPID,ERRCOD,MXLEN,Z,ERROR)
C =====
C   Purpose
C     To handle the error messages for User Defined Group.
C -----
C   Arguments
C
C   GRPID  input  int    ID of group or group name - not used
C   ERRCOD input  int    Error message number if any found
C   MXLEN  input  int    Maximum length of the message that can be
C   passed
C   Z      output char  Array to contain the message return
C   ERROR  output int    The code returned indicates the type of error
C
C   Called by
C           BCMSG routine
C -----

```

The purpose of this subroutine is to return an error String associated with an error code. The error codes are to be returned by the other 7 “BS...” routines and as such the “BSMSG” routine is used as the repository of all of the error messages for the Beam Library applications.

For example, suppose that “BSBRPD” application returns an error code of 5103 when a certain error condition occurs. Then, the Beam Library Client routines will expect that there will be a String returned from the “BSMSG” routine, which corresponds to this error code. These error messages will be printed in the “*.f06” file to guide the user as to what the error could have been and how to fix it. The string may be as long as 160 characters for this release of MSC.Nastran.

The following is an example of BSMSG code construct:

```

      IF(ERRCOD .EQ. 5103) THEN
      Z(1:MXLEN) = 'This is a User Specified Error Message
      ....'//
      &          'Messages could be 160 characters long ...'
      ERROR = ERRCOD
      .....
```

Again, it is highly recommended that you use the example Beam Server files as a template to generate your BSMSG routines.

Linking Your Library to MSC.Nastran

Once you have created the eight “BS...” routines, these routines may be linked with MSC.Nastran Beam Server Library to build a beam server executable. An example user-defined beam server has been delivered with Version 69 of MSC.Nastran. It is highly recommended that you study, build, and use the example “Beam Server” before you build your own version of the beam server.

The MSC.Nastran special library contains a main routine as well as the communications routines that allow MSC.Nastran to communicate with the user-defined beam server.

The MSC.Nastran user may connect up to 10 beam servers in a single job execution. This connection is made using the concept of evaluator groups described in the remainder of this section. For each group, the user specifies on the PBEAML/PBARL entry referring to an external beam server, MSC.Nastran will start and communicate with the beam server.

You may define as many beam evaluators as required using the “CONNECT” FMS commands. Only 10 of these evaluators, however, may be referenced in the groups on the PBARL or PBEAML Bulk Data entries.

The PBARL/PBEAML entries specifies a “Group” name on the fourth field of the first entry. This group name is associated with an “Evaluator” class using a “Connect” command in the FMS section. Finally, the “Evaluator” class is associated with an executable using the MSC.Nastran configuration file specified via the “gmconn” key word on the nastran command line.

The following example shows the mappings mentioned:

1. Group is referenced on the PBARL/PBEAML entry (or entries).
 PBARL,39,6,LOCSERV,I_SECTION (Specify the “Group” name.)

2. The “Group” is associated with an “Evaluator” class.
CONNECT,BEAMEVAL,LOCSERV,EXTBML (Associate the “Group” name with an “Evaluator” class.)
3. The external evaluator connection file associates the “Evaluator” class with a server executable. The following statement must be specified in the connection file:
EXTBML,-,*beam_server_pathname*
4. Refer to the external evaluator connection file on the command line using the “gmconn” keyword
nastran myjob ... gmconn=*external_evaluator_pathname*.
5. In the example Beam Library section that standard output (FORTRAN unit 5) or standard output (FORTRAN unit 6) are not used as these I/O channels are reserved by the Inter-Process Communications (IPC) subsystem.

Example of Building and Linking a Beam Server

As an example of building and linking a beam server executable, the sample beam server will be modified. Complete instructions on building and using a beam server are provided in the [MSC.Nastran 2004 Installation and Operations Guide](#) for your system.

- Make a copy of the beam server sample source.
- Edit the source for the BRTUPD subroutine; this routine describes the equations that convert the PBARL dimensions into the standard PBAR dimensions for a tube cross section.
- Add an extra multiplication of 3.0 to the DIMO(2) equation to increase the calculated moments of inertia.

Since the formulation of this bar section has been changed, the sensitivities for optimization will also change. Rather than calculate what the new sensitivities should be, the MSC.Nastran can calculate them using central differencing techniques. To permit this, edit the source file for the BRTUID subroutine and change all occurrences of SENTYP = 2 to SENTYP = 3.

Build your new beam server using the instructions detailed in the [MSC.Nastran 2004 Installation and Operations Guide](#). Once you have built the beam server executable, you must create an external evaluator connection to point to your executable. Typically, this file would be kept in the user's home directory, but for this example it will remain in the current directory. Edit the new file `bmconfig.fil`. Put the following line in the file:

`LOCBMLS, -, pathname`

where LOCBMLS is the evaluator referenced in the SAMPLE data file included with the beam server. Remember, this file can contain references to any number of beam servers.

To run the sample job, type in the following command:

```
nastran sample scr=yes bat=no gmconn=bmconfig.fil
```

Common problems which may occur when attempting to run an external beam library job are generally indicated in the F06 by USER FATAL MESSAGE 6498. If this message includes the text “No such group defined,” the PBARL/PBEAML selected a group not defined on a CONNECT entry. If UFM 6498 includes the text “No such evaluator class,” either the “gmconn” keyword was not specified or the CONNECT entry selected an evaluator not defined in the configuration file.

If the job was successful, you can look at the Design Variable History and see that the results for the variable mytubeor are different than the results for tubeor. These variables refer to the outer radius of tube sections from equivalent models. One model used the MSC tube section while the other used the tube section in your modified beam server.

3.2.7 Support for Offsets in Element Coordinates for CBAR & CBEAM Elements

Introduction

Bar elements (CBAR and CBEAM Bulk Data entries) have been used for many years to model structural members that can transmit axial loads, shear forces and bending moments. The offset feature of these elements allows the analyst to position the beam axis relative to the line connecting the two grid points used in the definition of the element.

Theory

Structural stiffeners such as I-beams, wide flange beams, channel sections, z-sections and hat sections are typically modeled using the bar and beam finite elements available in MSC.Nastran. When these members are used to support flat or curved panel structures, it is often desirable to include the offset effects of the member neutral/shear axis location when modeling the structure. These effects are most generally prevalent when the grid points are located at the mid-plane or outer surface of the panels. Under these circumstances, when a beam element is attached to these points, the line connecting the beam element end grid points does not correctly

represent the element neutral axis location relative to the panel surface. To handle this situation, the CBAR and CBEAM Bulk Data entries provide for the specification of offset vectors for each end of the element that locate the positions of the element end points with respect to the locations of the element connection grid points.

Until now, the components of the offset vectors had to be specified in the principal directions of the output coordinate systems of the connection grid points. The program would then transform each of the offset vectors so that the components were measured in the basic coordinate reference system of the model. The locations of the end grid points are also converted to this basic system if necessary. Adding the offsets to the coordinates of the grid points produces the element end point locations. The vector between the two end points together with the orientation vector can be used to generate the element coordinate system. Thus, when defining element offset vectors, the analyst must keep in mind the fact that the vectors must be specified in the output coordinate system directions of the connection grid points. For example, in the following figure, the local x direction at grid point GA is in the opposite direction of the local y direction at grid point GB and the local y at grid point GA and the local x at grid point GB are in the same direction. Let us assume that the element x-axis is offset 0.5 units in the local x direction at grid point GA from the line connecting grid points GA and GB. Then, in order to define the neutral axis of the element properly, the analyst must define the End A offset vector as $W_a = \{0.5, 0.0, 0.0\}$ and at the same time specify the End B offset vector as $W_b = \{0.0, -0.5, 0.0\}$.

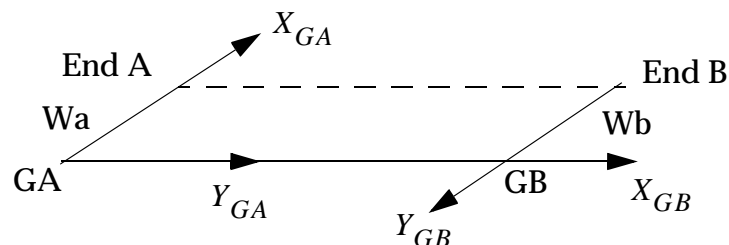


Figure 3-24

If we use the concept of the local element coordinate system, then no discussion of the local output coordinate systems employed at the connection grid points is necessary. For this case, the element x-axis is from GA to GB and the orientation vector is in the direction of X_{GA} , defining the direction of the element- y-axis and the x-y plane. The offset vector components will be the same at both ends and will be specified as $W_a = W_b = \{0.0, 0.5, 0.0\}$. The program will take care of any necessary transformations between the element system, the basic system and the local systems at the end grid points.

Inputs

The specification of the offset vectors for the two ends of the bar and beam elements is still accomplished using fields four through nine of the first continuation entry for the CBAR and CBEAM Bulk Data entries respectively. The orientation (reference) vector is specified using fields 6, 7 and 8 of the bulk data entries. How MSC.Nastran interprets the contents of these fields depends upon the offset and reference vector specification codes supplied in field nine of the main (parent) bulk data entry. Each of the entries is described by one character. There must be three characters in the string if it is present. The first character is associated with the end A offset vector. The second character is associated with the end B offset vector. The third character is associated with the orientation vector. The character “G” signifies that the input is specified in the Global coordinate reference system. The character “E” signifies that the input is specified in the Element local coordinate reference system. The character “B” signifies that the input is specified in the Basic coordinate reference system. Offset vectors can be defined in Global (default) or Element systems. The orientation vector can be defined in Global (default) or Basic systems. The following table describes the various valid character combinations that can be entered in this field.

String	End A Offset	End B Offset	Orientation Vector
G G G	Global	Global	Global
G G B	Global	Global	Basic
G E G	Global	Element	Global
G E B	Global	Element	Basic
E G G	Element	Global	Global
E G B	Element	Global	Basic
E E G	Element	Element	Global
E E B	Element	Element	Basic

The default entry for this field is “GGG”. This indicates that all vectors are defined in the global reference system. This is the historical specification for the vector inputs. Any attempt to specify invalid combinations results in a bulk data entry input error message. For example, a value of EEE indicates offset vectors for end A and end B, as well as the orientation vector, are all specified in the local Element reference system. This usage results in a fatal error because the orientation vector can be specified only in Global or Basic systems but not the Element local system.

Outputs

There are no special output considerations when the offset vectors are specified in the local element coordinate system.

3.2.8 Guidelines and Limitations

This feature is not applicable to the CBEAM entry when it is acting as a p-element beam.

3.2.9 Example

A simple model is presented to demonstrate the specification of the offset vectors in the element coordinate system. The input file contains two CBAR elements that use offsets. One of the CBARs uses the original method of specifying the offsets in the global system components at the connection grid points. The second of the CBARs uses the new technique of specifying the offsets in the element coordinate system. The displacements and element forces should be the same for the two elements if the offset vectors are “equivalent”.

Model Description

Two CBAR elements are used to demonstrate the use of the offset vectors in the element coordinate system. The grid points that define the ends of the elements are situated along the basic y-axis. The output coordinate system at each grid point is the basic coordinate system. The global system is equivalent to the basic system. The orientation vector for both elements places the element x-y plane at a thirty degree cant to the basic y-z plane toward the negative x-direction. Element 101 defines the offsets to be -0.50 units in the global (basic) x-direction and 0.866 units in the global (basic) z-direction. This is standard legacy usage of the offset vectors. Element 111 defines the offsets to be 1.0 units in the y-direction of the local element coordinate system. One end of each bar is completely constrained resulting in a cantilever beam model. Three subcases are defined to apply unit loads to the free end of the beam in each of the basic directions. Element forces are recovered for the CBAR elements for each subcase.

Input File

```
id test,cbar
sol 101
cend
$
title = v2004 release notes demonstration problem
subtitle = test cbar offset vector specification options
```

```

label= two equivalent cbars
disp=all
spcf=all
elfo=all
spc = 100
$
$ unit load in basic x-direction
$
subcase 100
load=100
$
$ unit load in basic y-direction
$
subcase 200
load = 200
$
$ unit load in basic z-direction
$
subcase 300
load = 300
begin bulk
$
$=====
$ offset bar example.
$ grid output cstm is basic
$ offset in element coordinate system is {0.0,1.0,0.0}(t)
$ which is equivalent to {-0.5,0.0,0.866025}(t) in basic
$ element force results should be the same for both elements
$
grid,1001,,0.0,11.0,0.0
grid,1002,,0.0,20.0,0.0
$
$ Standard legacy input.
$ Element 101 specifies offsets and orientation vector in the
$ global coordinate system.
$
cbar,101,1101,1001,1002,-0.50,0.0,.866025,,+br101
+br101,,-0.50,0.0,0.866025,-0.50,0.0,0.86602
$
$ New feature option input.
$ Element 111 specifies offsets in the local element system and
$ orientation vector in the global coordinate system.
$
cbar,111,1101,1001,1002,-0.50,0.0,.866025,eeg,+br111
+br111,,,0.0,1.0,0.0,0.0,1.0,0.0
$
force,100,1002,,1000.0,1.0,0.0,0.0
force,200,1002,,1000.0,0.0,1.0,0.0
force,300,1002,,1000.0,0.0,0.0,1.0
spc1,100,123456,1001
$
$=====

```

```
$ common data  
$  
pbar,1101,1,1.0,1.0,1.0,1.0  
mat1,1,10.7+6,,.33  
enddata
```

3.3 Surface Elements

The surface elements consist of the following:

- Shear panel
- Shell
- Conical shell

3.3.1 Shear Panel Element (CSHEAR)

The shear panel is defined with a CSHEAR entry and its properties with a PSHEAR entry. A shear panel is a two-dimensional structural element that resists the action of tangential forces applied to its edges, and the action of normal forces if effectiveness factors are used on the alternate form of the PSHEAR Bulk Data entry. The structural and nonstructural mass of the shear panel is lumped at the connected grid points. Details of the shear panel element are discussed in Section 3.0 of *The NASTRAN Theoretical Manual*.

The element coordinate system for a shear panel is shown in [Figure 3-25](#). The labels G1, G2, G3, and G4 refer to the order of the connected grid points on the CSHEAR entry. The element forces are output on request in either real or complex form.

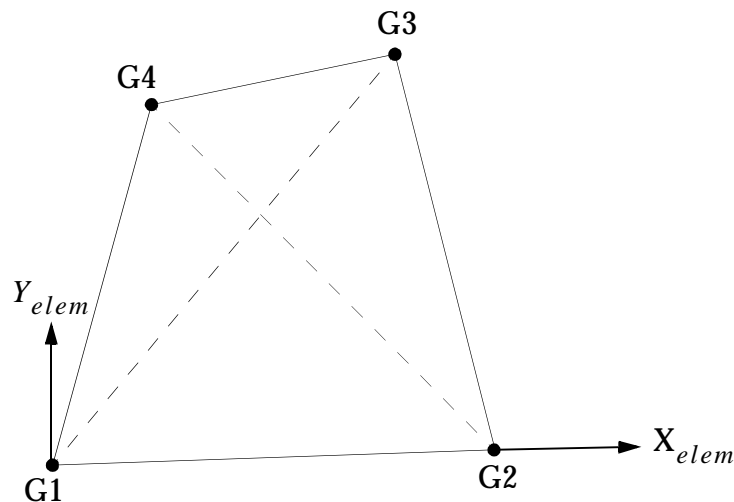


Figure 3-25 Shear Panel Connection and Coordinate System

The positive directions for these forces are indicated in [Figure 3-26](#). These forces consist of the forces applied to the element at the corners in the direction of the sides, kick forces at the corners in a direction normal to the plane formed by the two adjacent edges, and shear flows (force per unit length) along the four edges. The shear stresses are calculated at the corners in skewed coordinates parallel to the exterior edges. The

average of the four corner stresses and the maximum stress are output on request in either the real or complex form. A margin of safety is also output when the stresses are real.

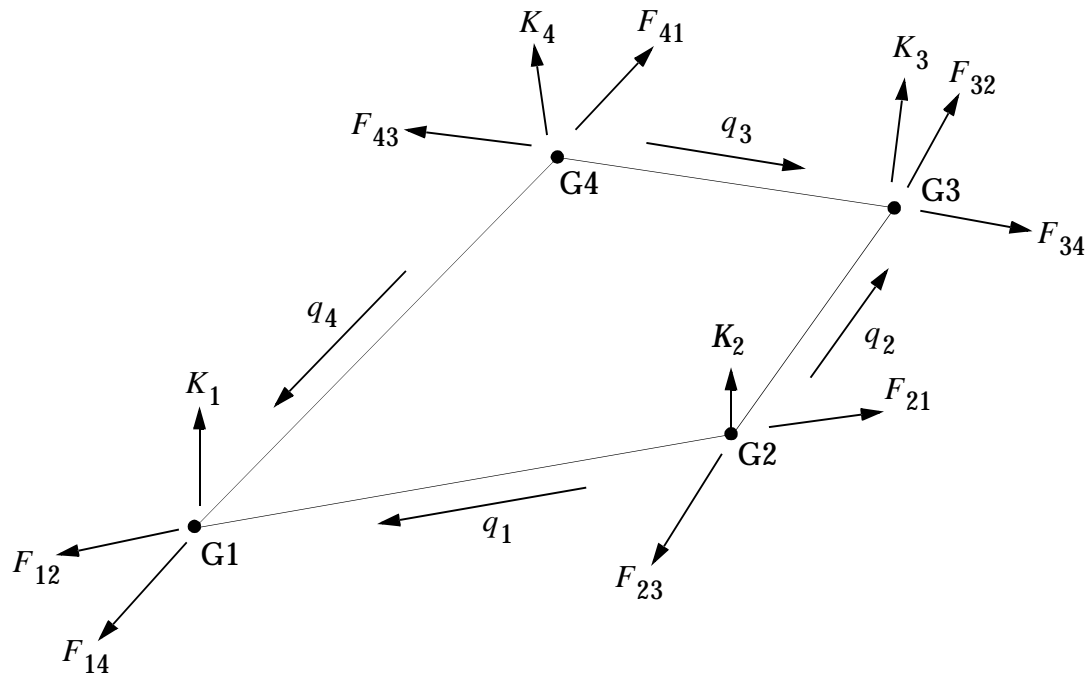


Figure 3-26 CSHEAR Element Corner Forces and Shear Flows

3.3.2 Shell Elements (CTRIA3, CTRIA6, CTRIAR, CQUAD4, CQUAD8, CQUADR)

MSC.Nastran includes two different shapes of isoparametric shell elements (triangular and quadrilateral) and two different stress systems (membrane and bending). There are in all a total of six different forms of shell elements that are defined by connection entries as follows:

- CTRIA3 -- Isoparametric triangular element with optional coupling of bending and membrane stiffness.
- CTRIA6 -- Isoparametric triangular element with optional coupling of bending and membrane stiffness and optional midside nodes.
- CTRIAR -- Isoparametric triangular element with no coupling of bending and membrane stiffness; the membrane stiffness formulation includes rotation about the normal to the plane of the element. See Section 3.3.3.
- CQUAD4 -- Isoparametric quadrilateral element with optional coupling of bending and membrane stiffnesses.

- CQUAD8 -- Isoparametric quadrilateral element with optional coupling of bending and membrane stiffness and optional midside nodes.
- CQUADR -- Isoparametric quadrilateral element with no coupling of bending and membrane stiffnesses; the membrane stiffness formulation includes rotation about the normal to the plane of the element. See Section 3.3.3.

The CQUAD4, CQUAD8, CTRIA6, and CTRIA3 elements may be used to model hyperelastic materials; see “[CQUAD4 and CQUADR Coordinate Systems](#)” on page 133.

The properties for the above elements are defined on the PSHELL entry. Anisotropic material may be specified for all shell elements. Transverse shear flexibility may be included for all bending elements on an optional basis. Structural mass is calculated from the membrane density and thickness. Nonstructural mass can be specified for all shell elements. Lumped mass procedures are used unless coupled mass is requested with the parameter COUPMASS. Differential stiffness matrices are generated for all shell elements. Plane strain analysis may be requested for all shell elements.

The element coordinate systems for triangular and quadrilateral shell elements are shown in [Figure 3-27](#) through [Figure 3-30](#). The integers 1 through 8 refer to the order of the connected grid points on the connection entries defining the elements. The angle, THETA, is the orientation angle for material properties. The element coordinate system for the CQUAD8 and CTRIA6 elements is a Cartesian system defined locally for each point (ξ , η) with the following orientation:

1. The plane containing x_{elem} and y_{elem} is tangent to the surface of the element.
2. For the CQUAD8 element, x_{elem} and y_{elem} are obtained by doubly bisecting the lines of constant ξ and η .
3. For the CTRIA6 element, x_{elem} is tangent to the line of constant η .
4. x_{elem} increases in the general direction of increasing ξ and y_{elem} of η .

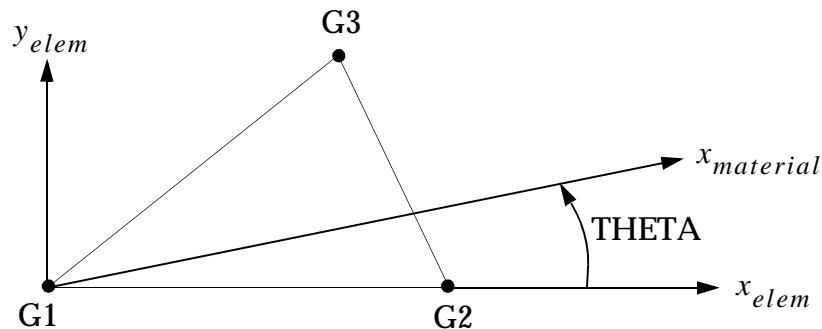


Figure 3-27 CTRIA3 and CTRIAR Coordinate Systems

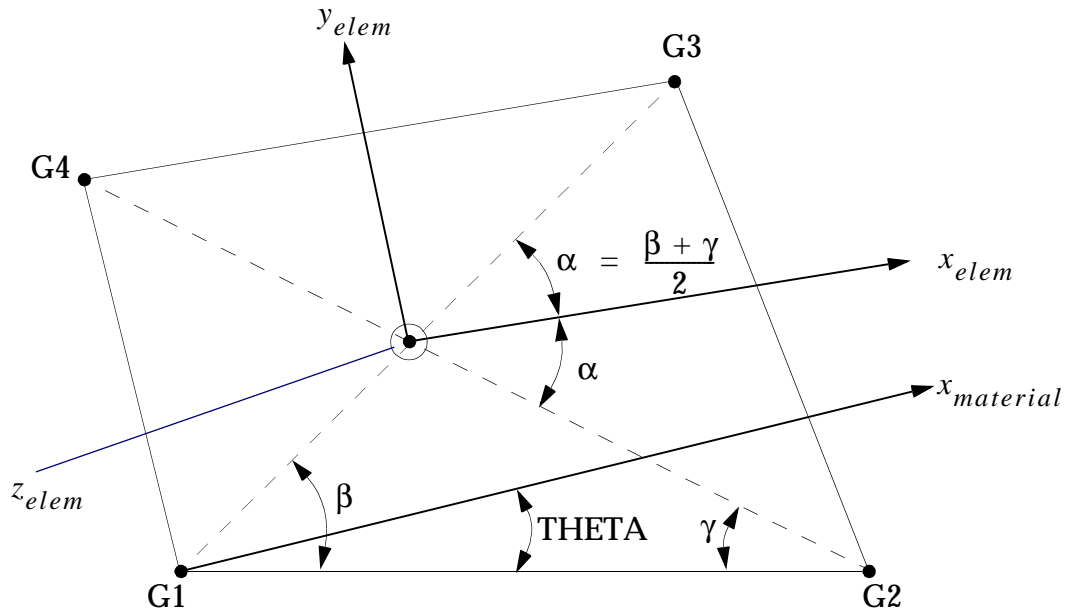


Figure 3-28 CQUAD4 and CQUADR Coordinate Systems

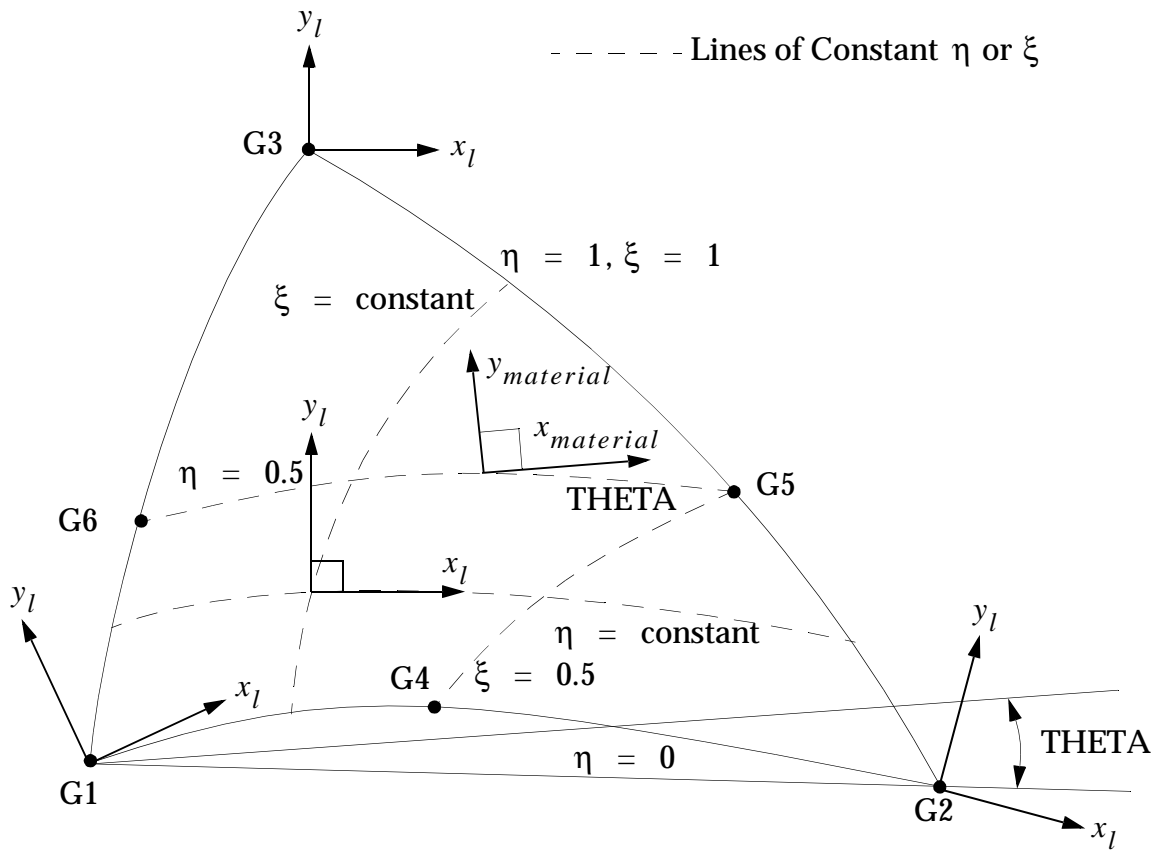


Figure 3-29 CTRIA6 Coordinate System

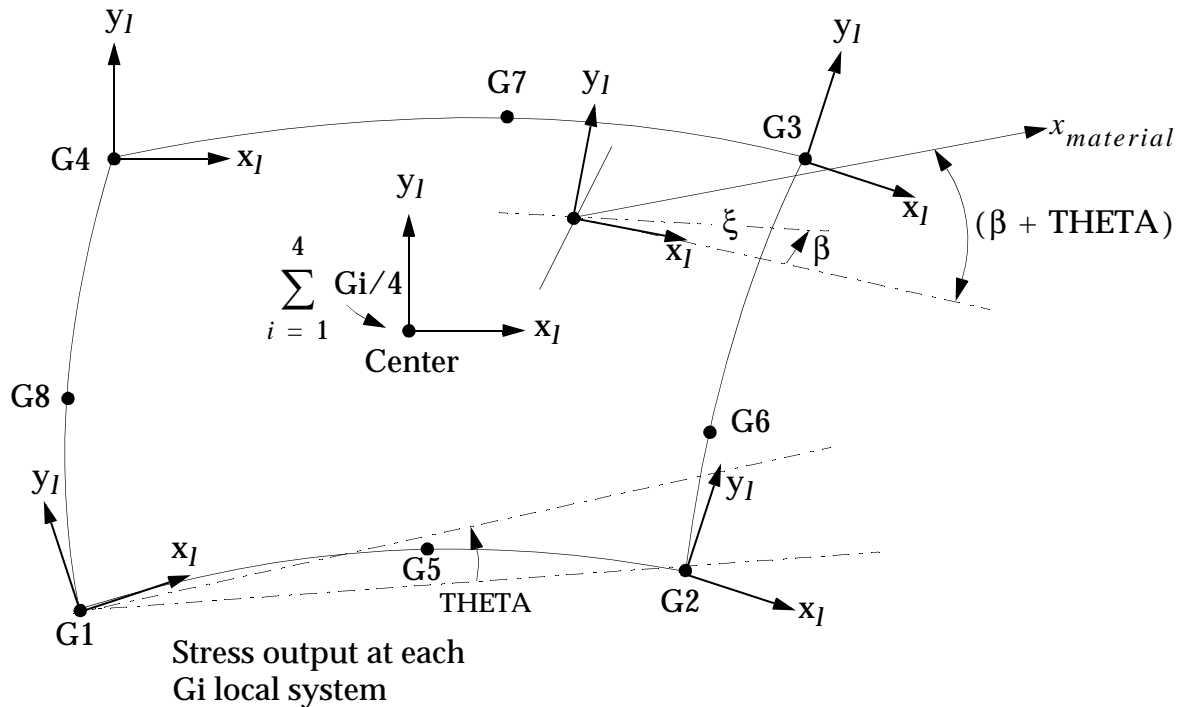


Figure 3-30 CQUAD8 Coordinate System

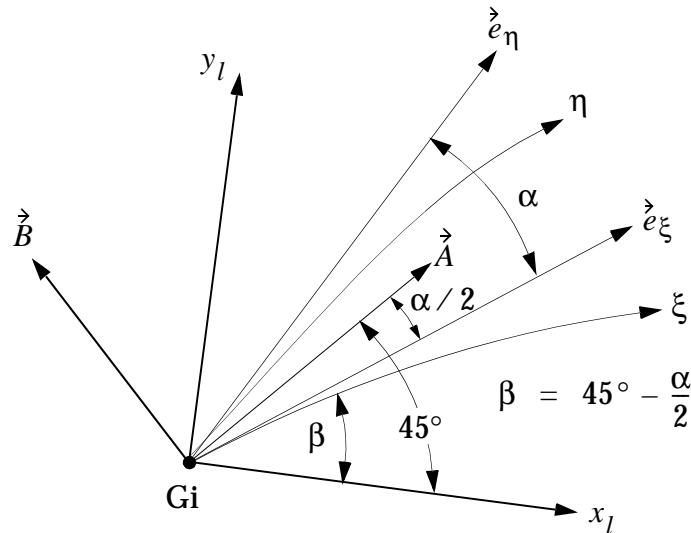


Figure 3-29 CQUAD8 Coordinate System (continued)

The CQUAD4, CQUAD8, CTRIA3, and CTRIA6 elements may be offset relative to the mean plane of their connected grid points. There are three commonly used techniques to define these offsets in MSC.Nastran. These are generically denoted as the ZOFFS, MID4, and RBAR methods and are described below.

It is generally recommended the ZOFFS be used if one has a sufficiently fine mesh in the region in which offsets are to be defined. In regions of a finite element model in which the analyst has determined that a coarse mesh is satisfactory, the use of RBARs to define offsets is generally more accurate.

Irrespective of which of the three methods is used to define the offset for the element, it is required that values be specified for both MID1 and MID2 on the PSHELL entry referenced by the offset element. The mass properties of an offset element are not modified to reflect the existence of the offset when the ZOFFS and MID4 methods are used. Therefore, if the weight or mass properties of an offset element are to be used in an analysis, the RBAR method must be used to represent the offset.

The CQUADR and CTRIAR elements may be offset by the RBAR method only, but no membrane-bending coupling is computed.

For the CQUAD4 and the CTRIA3, the forces are evaluated at the centroid of the element. For the CQUAD8, CTRIA6, CQUADR, and CTRIAR, the forces are evaluated at the centroid and at the vertices. The positive directions for shell element forces in the element coordinate system are shown in [Figure 3-31](#) and [Figure 3-32](#). The following element forces per unit of length, either real or complex, are output on request:

- Bending moments on the x and y faces.
- Twisting moment.
- Shear forces on the x and y faces.
- Normal forces on the x and y faces.

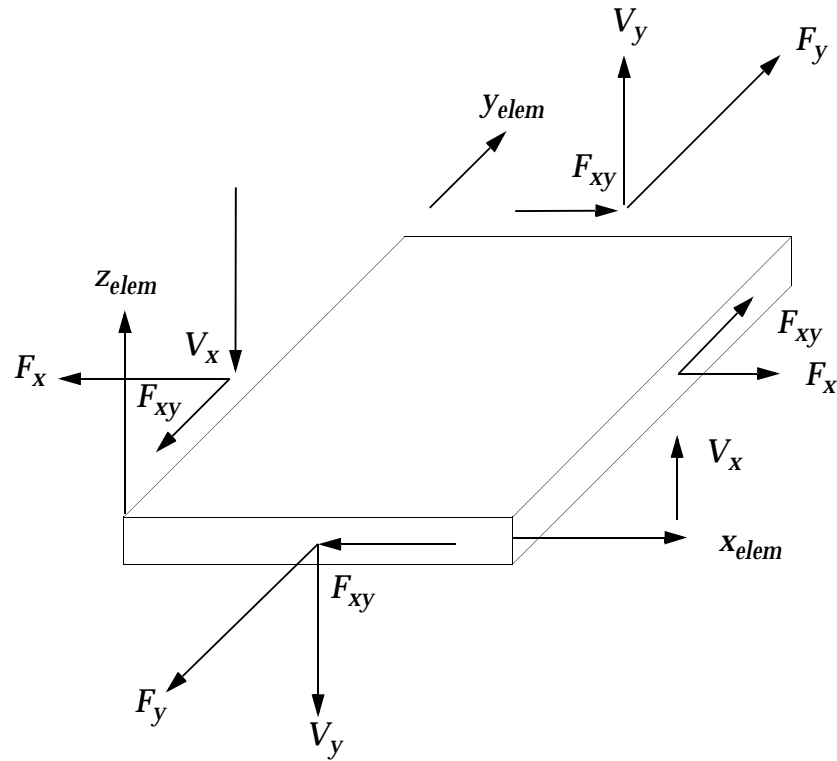


Figure 3-31 Forces in Shell Elements

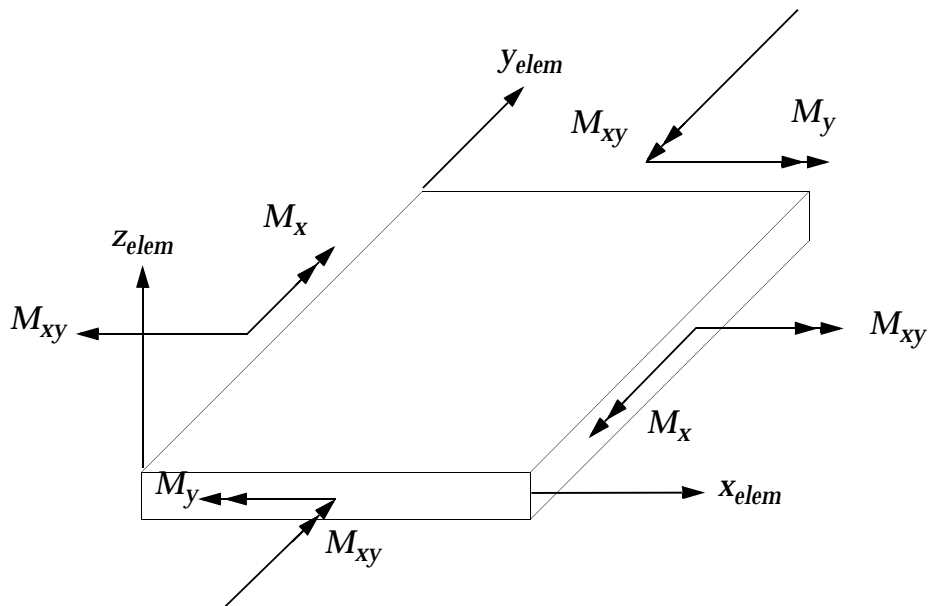


Figure 3-32 Moments in Shell Elements

For the CQUAD4 and CTRIA3 elements, the stresses are evaluated at the centroid of the element. For the CQUAD4, stress at the corner can be requested if STRESS(CORNER) is requested. For the CQUAD8, CTRIA6, CQUADR, and CTRIAR

elements, the stresses are evaluated at the centroid and at the vertices. The positive directions for the stresses are shown in **Figure 3-33**. The stresses are calculated in the element coordinate system. The following real stresses are output on request:

- Normal stresses in the x and y directions.
- Shear stresses on the x face in the y direction.
- Angle between the x axis and the major principal axis.
- Major and minor principal stresses.
- von Mises equivalent stress if STRESS(VONM) is requested or maximum shear stress if STRESS(SHEAR) is requested.

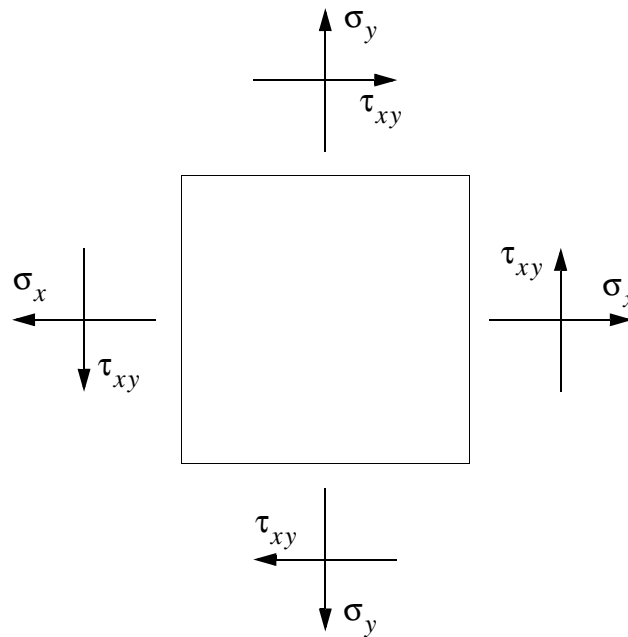


Figure 3-33 Stresses in Shell Elements

The von Mises equivalent stress for plane strain analysis is defined as follows:

$$\bar{\tau}_v = \left[1/2 \left\{ (\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 \right\} + 3\tau_{xy}^2 \right]^{1/2} \quad \text{Eq. 3-8}$$

For plane stress analysis, $\sigma_z = 0$. Only the normal stresses and shearing stresses are available in the complex form.

The von Mises equivalent strain is defined as

$$\bar{\epsilon}_v = \left[\frac{4}{9}(\epsilon_x^2 + \epsilon_y^2 - \epsilon_x \epsilon_y) + \frac{1}{3}\gamma_{xy}^2 \right]^{1/2} \quad \text{Eq. 3-9}$$

where the strain components are defined as

$$\varepsilon_x = \frac{\partial u}{\partial x}; \quad \varepsilon_y = \frac{\partial v}{\partial y}; \quad \gamma_{xy} = \frac{\partial v}{\partial y} + \frac{\partial u}{\partial x} \quad \text{Eq. 3-10}$$

and the curvatures are defined as

$$\chi_x = \frac{\partial^2 w}{\partial x^2}; \quad \chi_y = \frac{\partial^2 w}{\partial y^2}; \quad \chi_{xy} = 2 \frac{\partial^2 w}{\partial x \partial y} \quad \text{Eq. 3-11}$$

The maximum shear stress is

$$\hat{\tau} = \left[\left(\frac{\sigma_x - \sigma_y}{2} \right)^2 + \tau_{xy}^2 \right]^{1/2} \quad \text{Eq. 3-12}$$

The maximum shear strain is

$$\hat{\gamma} = [(\varepsilon_x - \varepsilon_y)^2 + \gamma_{xy}^2]^{1/2} \quad \text{Eq. 3-13}$$

The stresses are calculated at two specified points on the cross section. The distances to the specified points are given on the property entries. The default distance is one-half the thickness. The positive directions for these fiber distances are defined according to the right-hand sequence of the grid points specified on the connection entry.

In addition, interpolated grid point stresses and mesh stress discontinuities are calculated in user-specified coordinate systems for grid points which connect the shell elements. See “[Element Data Recovery Resolved at Grid Points](#)” on page 463 for a description of the Case Control Section to request grid point stresses. Only real stresses are available at the grid points. Mesh stress discontinuities are available in linear static analysis only.

Grid point stresses are computed by

$$\sigma_g = \sum_{e=1}^{N_e} W_{ge} \sigma_{ge} \quad \text{Eq. 3-14}$$

where σ_{ge} , a grid point stress component, multiplied by W_{ge} , the interpolation factor, and summed for all elements, N_e , connected to the grid point.

The stress discontinuity for one component and one element is

$$\delta_{ge} = \sigma_{ge} - \sigma_g \quad \text{Eq. 3-15}$$

The discontinuity from all elements for one component is then obtained by

$$\delta_g = \sqrt{\sum_{e=1}^{N_e} (W_{ge} \delta_{ge})^2} \quad \text{Eq. 3-16}$$

The total discontinuity at a grid point from all elements and all components N_c defines the error and is obtained by

$$\delta_g = \sqrt{\frac{1}{N_c} \sum_{c=1}^{N_c} \delta_{gc}^2} \quad \text{Eq. 3-17}$$

δ_{gc} and δ_g are requested and printed with the GPSDCON Case Control command.

The total discontinuity at an element for one component over all of its grid points N_g is

$$\delta_{ec} = \sqrt{\frac{1}{N_g} \sum_{g=1}^{N_g} \delta_g^2} \quad \text{Eq. 3-18}$$

and the error for an element is

$$\delta_e = \sqrt{\frac{1}{N_g} \sum_{g=1}^{N_g} \delta_g^2} \quad \text{Eq. 3-19}$$

δ_{ec} and δ_e are requested and printed with the ELSDCON Case Control command.

The CQUAD4 element is intended for use when the surfaces are reasonably flat and the geometry is nearly rectangular. For these conditions, the quadrilateral elements eliminate the modeling bias associated with the use of triangular elements and the quadrilaterals give more accurate results for the same mesh size. If the surfaces are highly warped, curved, or swept, triangular elements should be used. Under extreme conditions, quadrilateral elements will give results that are considerably less accurate than triangular elements for the same mesh size. Quadrilateral elements should be kept as nearly square as possible, because the accuracy tends to deteriorate as the

aspect ratio of the quadrilateral increases. Triangular elements should be kept as nearly equilateral as practicable because the accuracy tends to deteriorate as the triangle becomes obtuse and the ratio of the longest to the shortest side increases.

The CQUADR and CTRIAR elements are improved shell elements that, when compared to the CQUAD4 and CTRIA3, respectively, are much less sensitive to high aspect ratios and values of Poisson's ratio near 0.5. In these elements, a rotational stiffness is computed about the normal to the element at the vertices and used in the formulation of the element stiffness. It should be noted that this degree-of-freedom must not be constrained unless at a prescribed boundary. The CQUADR and CTRIAR element stiffness formulations do not include coupling of membrane and bending stiffness. These elements are not implemented for differential and nonlinear stiffness and heat transfer analysis.

The CQUAD8 and CTRIA6 elements should not be used with all midside nodes deleted since they are excessively stiff in this configuration. A user warning message is produced when all midside nodes are deleted. Use CQUAD4 and CTRIA3 for this condition.

3.3.3 New QUADR and TRIAR Elements

Introduction

Traditional Mindlin shell elements, such as QUAD4, have 5 degrees-of-freedom per grid point--three translations and two bending rotations. The stiffness for the rotational degree-of-freedom normal to the element (the drilling degree-of-freedom) is zero. This creates modeling difficulties which may eventually lead to poor solutions. Many methods--such as AUTOSPC and K6ROT--have been devised to remedy this deficiency. None of them, however, supply physical stiffness to the drilling degree-of-freedom to resist the drilling moments.

Shell elements (the old QUADR/TRIAR) with drilling degrees-of-freedom was introduced in MSC.Nastran Version 66. However, these elements have not been widely used in a production environment. The reason for this is that they lack many capabilities such as the differential stiffness, coupling in membrane and bending, and support of composite materials, etc. Also, the old QUADR/TRIAR formulations do not yield accurate solutions for curved shell models.

The new QUADR/TRIAR formulation removes the limitations of the old QUADR/TRIAR mentioned above.

Various examples are run comparing results using the QUADR, QUAD4, and QUAD8. Overall, the new QUADR provides the most accurate results and is the recommended element.

Benefits

The results for the new QUADR are compared with the QUAD4 and the old QUADR.

Comparing with the QUAD4, the new QUADR has the following benefits:

- It has six degrees-of-freedom at each grid point as opposed to the QUAD4, which has five degrees-of-freedom. This removes the necessity of using such methods as the AUTOSPC and K6ROT--which sometimes cast doubts on results--to account for the normal rotational DOF in the QUAD4.
- When using AUTOSPC or K6ROT for the QUAD4, the stiffness at the drilling degree-of-freedom is not an actual physical stiffness. Therefore, drilling loads cannot be correctly transferred between elements, which lead to modeling difficulties. For QUADR, the stiffness at the drilling degree-of-freedom is an actual physical stiffness and, therefore, the drilling loads can be transferred correctly between elements. The ability of the QUADR to resist the drilling moments correctly is demonstrated in Example 3.
- The membrane behaviors of the new QUADR is far superior than those of the QUAD4 and approaches that of the QUAD8, as demonstrated by Examples 2 and 6.

Many capabilities that are available to the QUAD4 are not included in the old QUADR. For the new QUADR, all the capabilities in QUAD4 have been incorporated into the QUADR. In comparison with the old QUADR, the current QUADR has the following benefits:

- An improved theoretical formulation to make the current QUADR a more robust element (see the **Theory** Section).
- Coupled with the shell normal, the current QUADR is well suited for modeling shell problems (see the Section on **Shell Models and Shell Normal**).
- The rotational mass is implemented for the drilling degree-of-freedom (see the Section **Mass Properties**.)
- The consistent surface loads and edge loads are implemented (see the Section **Consistent Surface and Edge Loads**).

- The differential stiffness matrix is implemented for the new QUADR so that buckling and other analyses requiring the differential stiffness can be performed.
- Coupling between bending and membrane is implemented.
- Composite material is implemented for layered stress/strain output.
- Offset is allowed for the new QUADR.
- The new QUADR is implemented into the solution sequence SOL 200, optimization.
- New QUADR is supported in heat transfer analysis.
- A System cell 370 is available for converting between QUAD4/TRIA3 and QUADR/TRIAR.
- Nonlinear material and geometric analysis will be implemented in a future version.

Theory

Allmann and Cook were the first to propose a successful membrane element with drilling DOFs. On the other hand, Hughes and Brezzi presented a variational formulation, which was first suggested by Reissner. This variational principle uses the drilling degrees-of-freedom as independent rotation field; i.e., it is based on the separate kinematic variables of displacements and rotations. Ibrahimbegovic, Taylor and Wilson combined the Allmann type of interpolation and the variational principle to construct a quadrilateral shell element. In MSC.Nastran, membrane elements with drilling degrees-of-freedom were first introduced in 1988. A summary of the theory for the membrane portion of the shell element is given below.

Allman type of interpolation fields assumes the normal displacements to be quadratic and the tangential displacements to be linear. For example, along the edge from grid 1 to 2 (see [Figure 3-34](#)), the displacement in the middle of the edge is

$$\mathbf{u}_m = \frac{1}{2}(\mathbf{u}_1 + \mathbf{u}_2) + \frac{1}{8}l_1(\theta_2 - \theta_1)\mathbf{n}_1 \quad \text{Eq. 3-20}$$

where all bold letters indicate vectors with x and y -components. \mathbf{u}_1 and \mathbf{u}_2 are the displacements at grids 1 and 2, respectively; l_1 is the length of the edge; θ_1 and θ_2 are the drilling rotations at grid 1 and 2, respectively; \mathbf{n}_1 is the vector normal to the edge in the x - y plane; and \mathbf{u}_m is the interpolated displacement vector in the middle of the edge.

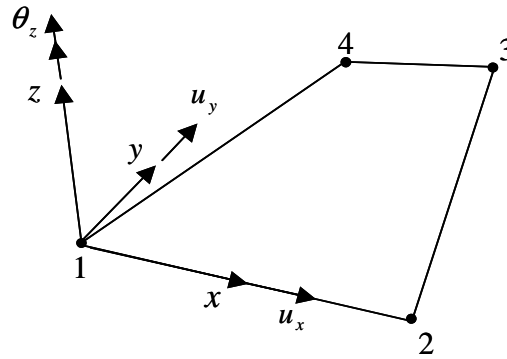


Figure 3-34 In-Plane displacements and drilling rotation

Hughes and Brezzi presented two variational formulations: one is the mixed type and the other is the displacement type, which is the one adopted here. The displacement type variational formulation states that the function to be minimized for elements with drilling degrees-of-freedom is

$$\Pi = \frac{1}{2} \int_A \boldsymbol{\varepsilon}^T \mathbf{G} \boldsymbol{\varepsilon} t dA + \frac{1}{2} \gamma G \int_A (\Omega - \theta)^2 t dA \quad \text{Eq. 3-21}$$

$$\Omega = \frac{1}{2} \left(\frac{\partial u_y}{\partial x} - \frac{\partial u_x}{\partial y} \right) \quad \text{Eq. 3-22}$$

where:

γ = non-dimensional constant, for most problems $\gamma = 1.0$

G = the shear modulus, A and t are the area and thickness

Ω = the rotational field given by [Eq. 3-22](#)

θ = the interpolated drilling field

$\boldsymbol{\varepsilon}$ = the strain vector

\mathbf{G} = the stress-strain relationship defined by $\boldsymbol{\sigma} = \mathbf{G} \boldsymbol{\varepsilon}$

The first term is the normal strain energy and the second term is the contribution due to the drilling degrees-of-freedom.

In the variational formulation given by [Eq. 3-21](#), the drilling field and the displacement field are interpolated independently. For the drilling field, the interpolation function used is the linear QUAD4 shape function. For displacement field, the Allman type interpolation functions given by [Eq. 3-20](#) are used. They are given by the following equations:

$$\theta(\xi, \eta) = \sum_{i=1}^4 N_i(\xi, \eta) \theta_i \quad \text{Eq. 3-23}$$

$$\mathbf{u}(\xi, \eta) = \sum_{i=1}^4 N_i(\xi, \eta) \mathbf{u}_i + \sum_{i=1}^4 N_{i+4}(\xi, \eta) \frac{l_i}{8} (\theta_{i+1} - \theta_i) \mathbf{n}_i c_r + N_0(\xi, \eta) \mathbf{u}_0 \quad \text{Eq. 3-24}$$

where:

$$i = 1, 2, 3, 4$$

$$\theta_5 = \theta_1$$

The N_i s in above equations are shape functions for QUAD4 and QUAD8 elements. They are defined by:

$$\begin{aligned} N_i &= \frac{1}{4}(1 + \xi_i \xi)(1 + \eta_i \eta) & \text{for } i = 1, 2, 3, 4 \\ N_i &= \frac{1}{2}(1 - \xi^2)(1 + \eta_i \eta) & \text{for } i = 5, 7 \\ N_i &= \frac{1}{2}(1 + \xi_i \xi)(1 - \eta^2) & \text{for } i = 6, 8 \end{aligned} \quad \text{Eq. 3-25}$$

The N_0 in **Eq. 3-24** is a bubble function or an auxiliary strain function. The \mathbf{u}_0 is the displacement corresponding to the bubble function or the auxiliary function. The \mathbf{n}_i is the outward normal vector to element boundary and l_i is its length.

With the shape functions defined by **Eq. 3-23** and **Eq. 3-24**, and the variational principle defined by **Eq. 3-21**, the standard finite element theory can be used to obtain the element forces and stiffness.

With the formulation given above, the singularity in the 6th degree-of-freedom of element such as QUAD4 is replaced with a true stiffness. In addition, the drilling rotation becomes a true load carrying degree-of-freedom. We will show that the membrane behavior is significantly improved. In fact, the accuracy of QUADR approaches that of the QUAD8.

Shell Models and Shell Normal

The shell element described in the previous **Theory** Section is for a flat element. In using flat shell elements for modeling a curved shell, local normals of adjacent shell elements have different directions. In MSC.Nastran, an algorithm has been developed to generate a unique normal at a grid point. The unique normal is the average normal

at each grid point so that the angles between the average normal and all the local normals of the adjacent shell elements are minimized. This unique normal is called the **Shell Normal**, which is requested by the PARAM, SNORM or the Bulk Data entry SNORM.

The old QUADR/TRIAR, formulation--which requires the shell normal in order to give a good solution for many shell problems--is not as robust as the new formulation.

The new QUADR/TRIAR, formulation is more robust, is less sensitive to the shell normal, and is well suited for solving shell problems. The following list illustrates the effect of the shell normal to the solution of shell problems. Some of the problems are the MacNeal/Hard test problems.

- Twist beam (MacNeal/Harder test, see example below) -- no effect, excellent solution.
- Scordelis-Lo roof (MacNeal/Harder test, see example below) -- no effect, excellent solution.
- Spherical shell (MacNeal/Harder test, see example below) -- the solution is slightly stiffer with the shell normal than without the shell normal. The solution for the coarse model with shell normal converges rapidly to that of the fine mesh without shell normal.
- Cylindrical shell with internal pressure (see example below) and without the shell normal, solution converges with the fine mesh. With the shell normal, the solution is excellent even with a coarse mesh.
- Pinched cylinder (not presented) -- almost no effect, excellent solution. The pinched cylinder is a cylinder with two rigid end diaphragms. This cylinder is subject to a pair of concentrated forces, which pinch the cylinder.
- Raasch hook (see example below) -- no effect, excellent solution.
- We can conclude from the results that the shell normal is beneficial for membrane-dominant shell problems, and unfavorable effect for bending-dominant shell problems. To provide good solutions for a wide class of shell problems, the default value for PARAM, SNORM is set to 20.0. Also from the above results, we can conclude that, with or without the shell normal, the new QUADR/TRIAR is well suited for modeling shell problems.

Shell normal yields wrong results for elements with offsets. Therefore, the shell normal should be turned off globally (PARAM,SNORM,0.0) or turned off locally with the Bulk Data entry, SNORM.

Mass Properties

In MSC.Nastran, two mass formulations are available--the lumped mass and the coupled mass. Lumped mass is the default formulation. Coupled mass formulation is requested by the parameter PARAM, COUPMASS, 1.

For the QUADR/TRIAR, the mass properties are as follows:

- Lumped mass -- only the translational masses are computed. No rotational mass is computed for the drilling degrees-of-freedom.
- Coupled mass -- in addition to the translational masses, the rotational masses are computed for the drilling degrees-of-freedom.

Consistent Surface and Edge Loads

PLOAD4, the most commonly used surface load, may have both normal and tangential components relative to the surface of an element. For the QUADR/TRIAR element, the consistent loads due to the surface load contain both concentrated forces at the connected grid points and the drilling moments at the drilling degrees-of-freedom. For the new QUADR/TRIAR, the consistent surface loads have been implemented.

For QUADR/TRIAR--because of the presence of drilling degrees-of-freedom--it is important to compute the consistent grid point forces for any distributed load along the edge of the element. This point is best illustrated by the following example. A single QUADR element is loaded with uniform tensile edge load along the opposite edge as shown in [Figure 3-35A](#). If we lump the uniform load into grid point forces, as shown in [Figure 3-35B](#), then we will get the non-uniform displacement solution, instead of the correct uniform displacement solution. This error can be corrected if we compute the consistent grid point forces for the distributed edge loads. The consistent grid point forces consist of the lumped grid forces and a pair of drilling moments, as shown in [Figure 3-35C](#).

For QUADR/TRIAR only, the consistent edge loads have been implemented. User can request the consistent edge loads using the PLOAD4 Bulk Data entry.

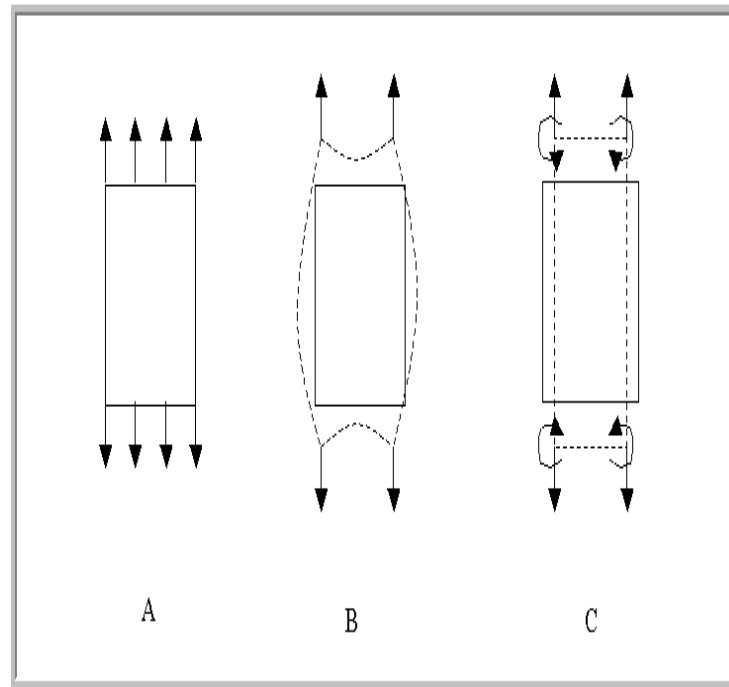


Figure 3-35 Consistent Edge Loads

User Interfaces

Parameters SNORM and Bulk Data Entry SNORM

The parameter SNORM is used for requesting unique shell normals for all grid points which are connected to the shell elements, such as QUADR, in a model. The default shell normal value is PARAM, SNORM, 20.0. For relationships between a shell model and the shell normal, please see the discussions in the **Shell Normal and Shell Models Section**.

The Bulk Data entry, SNORM requests a local shell normal at a grid point. It will override the shell normals requested by PARAM, SNORM, and can also be used to turn off the local default shell normal.

Bulk Data Entries CQUADR, CTRIAR, and PSHELL

The QUADR/TRIAR elements are requested by the existing CQUADR, CTRIAR, and PSHELL Bulk Data entries. For the new QUADR/TRIAR, there is no modification to the PSHELL entry. For the CQUADR and CTRIAR entries, two new fields are added:

- ZOFFS -- this entry is used to input the offsets from the surface of the grid points to the element plane.

- TFLAG -- this flag allows user to input the relative values for the variable grid point thickness, instead of absolute values. See Section 3.3.5 for further details.

Bulk Data Entry PLOAD4

The PLOAD4 entry is used to define a pressure load for a 3-D or 2-D element. For the QUADR/TRIAR only, this entry is extended to accommodate consistent edge load. See the **Consistent Surface and Edge Loads** Section for further details. Two new fields are added to PLOAD4 for this purpose:

- SORL -- selects whether the load is a surface or an edge load.
- SORL=SURF selects the surface load and SORL=LINE selects the consistent edge load. SORL=SURF is the default.
- LDIR -- defines the direction of the consistent edge load (SORL=LINE). It has the value of X, Y, Z, TANG, or NORM. See “**PLOAD4**” on page 1871 of the for their meanings. The default is NORM.

For example, the uniform extensile edge loads shown in **Figure 3-35** can be specified by the following PLOAD4 entries:

1	2	3	4	5	6	7	8	9	10
PLOAD 4	10	1	20.0						
					LINE				

The edge loads have a magnitude of 20.0 and the direction is NORM, i.e., it is in the mean plane of the element, normal to the edge, and pointing outward from the element.

NASTRAN System (370) -- QRMETH

The new default QUADR/TRIAR formulation is recommended. However, a new system cell 370 (QRMETH) is available for reverting back to the old formulation.

A companion version of the QUAD4/TRIA3 is also developed along with the new QUADR/TRIAR. We will call this version the new formulation for QUAD4/TRIA3, and the existing QUAD4 and TRIA3 the old formulation. The performances of the old and the new formulations are about the same.

The default for the QUADR/TRIAR uses the new formulation. The default for the QUAD4/TRIA3 uses the old formulation.

In addition to the selection of the new or old formulation for QUADR/TRIAR, QRMETH provides the following options:

- 0 -- selects the new QUADR/TRIAR formulation.
- 1 -- selects the old QUADR/TRIAR formulation.
- 2 -- converts QUADR/TRIAR into QUAD4/TRIA3 using the new QUAD4/TRIA3 formulation.
- 3 -- converts QUADR/TRIAR into QUAD4/TRIA3 using the old QUAD4/TRIA3 formulation.
- 4 -- selects the new QUAD4/TRIA3 formulation.
- 5 -- converts QUAD4/TRIA3 into QUADR/TRIAR using the new QUADR/TRIAR formulation.

The default is for QRMETH is 0.

Converting a QUAD4 model to QUADR

The following examples will illustrate that the QUADR is superior to the QUAD4. User may want to convert a QUAD4 model to QUADR by using system cell QRMETH. However, the following should be noted:

- Using QRMETH=5 will convert all QUAD4/TRIA3 in a model to QUADR/TRIAR.
- At the boundaries of the model, the drilling degrees-of-freedom must be constrained if the user wants fixed boundaries. For example, in the example **Static Analysis of Straight Beam with Coarse Mesh**, the drilling degrees-of-freedom must be constrained at the left end.
- At internal grid points of a model, user must remove any SPC or the PS field on the GRID Bulk Data entries which constrain the drilling degrees-of-freedom because the QUADR/TRIAR supply stiffness for these degrees-of-freedom. On the other hand, if user uses K6ROT or AUTOSPC to manage the drilling degrees-of-freedom, these will be automatically converted with QRMETHD=5. No user action is required.

Examples

Example 1 - Static Analysis of Straight Beam with Coarse Mesh

For membrane behavior, the performances of the QUADR element are substantially better than those of the QUAD4 element. This can best be illustrated by the static analysis of a straight beam with coarse mesh, shown in [Figure 3-36](#). The load is an in

plane shear load acting at the tip of the beam. This test is taken from the MacNeal-Harder tests and is designed to exhibit in plane behavior of shell elements for different element shapes. The results for this test are shown in [Table 3-6](#). The displacements shown are at the tip of the beam, normalized to the theoretical value.

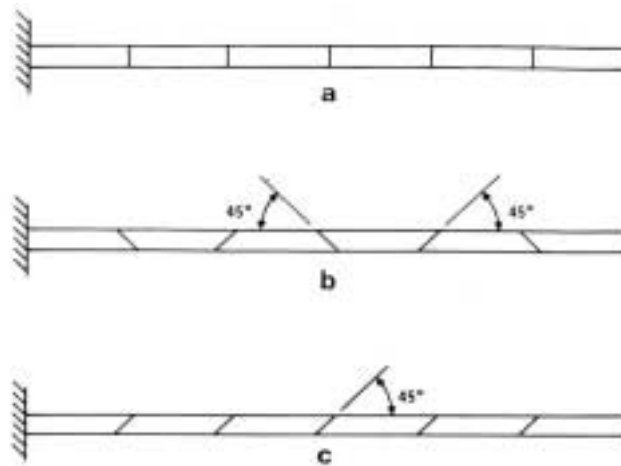


Table 3-6 Normalized displacements for in-plane shear load at tip

Element	Rectangle	Trapezoid	Parallelogram
QUAD4	0.9929	0.0515	0.6232
QUADR	0.9926	0.9613	0.9491

From the results, we see that the QUADR provides excellent results for both regular and irregular element shapes, while QUAD4 fails for the irregular shapes. We may argue that, in real life, we don't use element mesh as coarse as the one in this test. Next, we will look at the results of a fine element mesh, which is discussed in next sub-example.

Example 2 - Normal Modes Analysis of Straight Beam with Fine Mesh

For this example, a refined mesh of Example 1 is used in a normal mode analysis. [Table 3-7](#) summarizes the results of the first mode using various element shapes.

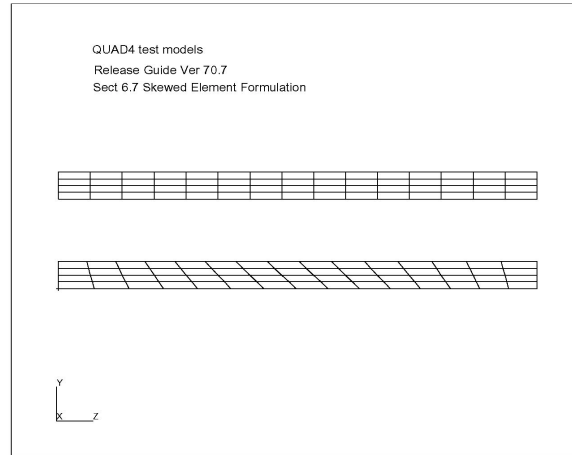


Figure 3-36 Straight beam with fine mesh.

Table 3-7 In-Plane Bending Frequency of Straight Beam with Fine Mesh

Element	Rectangle (hz)	Trapezoid (hz)	Parallelogram (hz)
QUAD4	9.39	15.89	9.41
QUADR	9.39	9.42	9.39

Once again, the QUAD4 element exhibits locking effects for trapezoidal shapes. Even for finer meshes, the results of the QUADR elements are still noticeable better than the results of the QUAD4 element in the case of distorted element shapes.

Example 3 - Static Analysis of Straight Beam with Drilling Moments

This example shows that the QUADR element provides the true physical stiffness for the drilling degrees-of-freedom and hence able to resist the actual drilling loads. A straight beam is modeled by six QUADR elements and is subject to a pair of end drilling moments as shown in [Table 3-7](#).

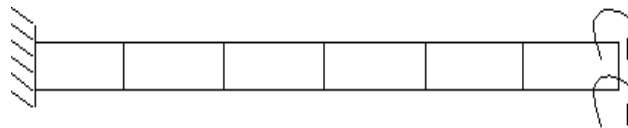


Figure 3-37 Straight Beam Subject to Tip Drilling Moments

The tip displacements of the straight beam for a static analysis are

- Theory = -5.40e-2
- Finite element solution = -5.46e-2

The above results show that the QUADR can predict correctly the response due to applied drilling moments.

Example 4 - Static Analysis of Cylindrical Shell with Internal Pressure

The purpose of this example is to illustrate the benefits of the shell normal in a shell structure modeled by QUADR elements. The shell normal is discussed in the **Shell Models and Shell Normal** Section. A cylinder of height 1000.0, radius 1000.0, and thickness 10.0, is subject to internal pressure of 1.0, as shown in **Figure 3-38**. The cylinder is free at both edges. Due to symmetry, only half of the height and a quarter of the cylinder is modeled. The radial displacement at the free edge is given in **Table 3-8**.

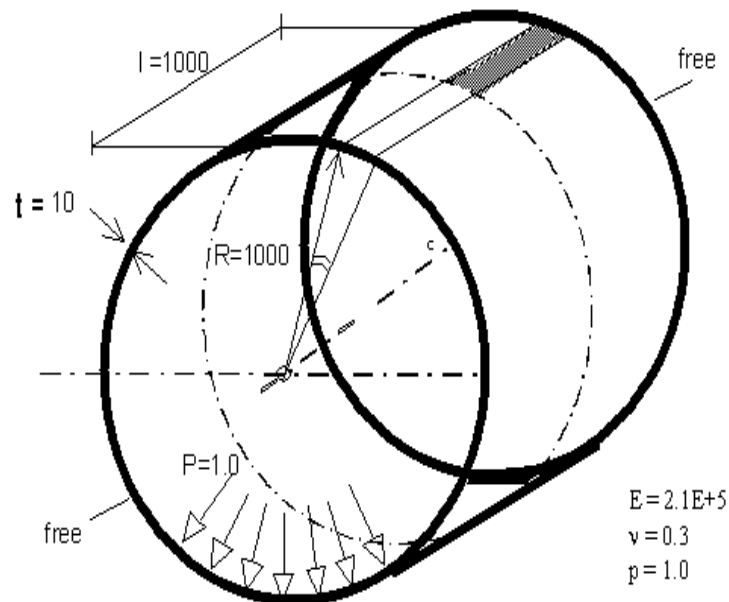


Figure 3-38 Cylindrical Shell with Internal Pressure

Table 3-8 Radial Displacement at Free Edge for Pressurize Cylinder

Mesh	Theory	With SNORM	No SNORM
4x4	0.4762	0.4670	0.6733
8x8	0.4762	0.4739	0.5251
16x16	0.4762	0.4756	0.4882

From **Table 3-8**, we can see that the results with the shell normal are substantially better than those without the shell normal.

Example 5 - Raasch Hook

The Raasch hook is a curved strip clamped at one end and loaded at the other end with an in-plane shear load as shown in **Figure 3-39**. A static analysis is performed on the Raasch hook for cases with and without the shell normal. The tip displacements in the direction of the shear force are summarized in **Table 3-9** using the new QUADR with various mesh size.

From **Table 3-9**, we can see that, with or without the shell normal, the results for the new QUADR are excellent. Using the old QUADR, the solution converges to the wrong results if shell normals are not used.

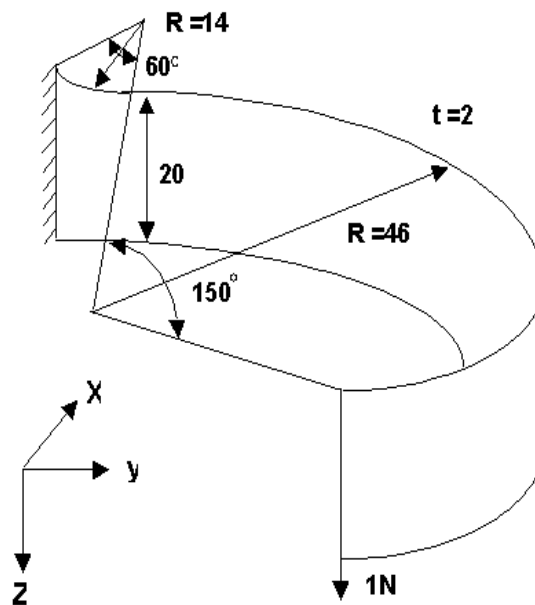


Figure 3-39 Raasch Hook

Table 3-9 Tip Displacement for Raasch Hook

Mesh	Theory	With SNORM	No SNORM
1x9	5.012	4.600	4.601
3x17	5.012	4.890	4.888
5x34	5.012	4.916	4.915
10x68	5.012	4.970	4.971
20x136	5.012	5.011	5.012

Example 6 - MacNeal and Harder Tests

MacNeal and Harder suggested a series of test problems for evaluating the accuracy of finite element. We have run these test problems for QUADR, QUAD4 and QUAD8 for comparisons. The results are summarized in [Table 3-10](#) and a score is given to each test problem based on how well it compares to the theoretical solution. The scoring criteria is shown:

- A -- error is less than and equal to 2%.
- B -- error is between 2% and 10%.
- C -- error is between 10% and 20%.
- D -- error is between 20% and 50%.
- F -- error is greater than 50%.

From [Table 3-10](#), we see that QUADR got the best scores among the three elements. It does not have a score below B. Between QUADR and QUAD4, the results of QUADR are better than those of the QUAD4. Between QUADR and QUAD8, the accuracy of the QUADR approaches that of QUAD8 for membrane problems and is better than that of QUAD8 for bending problems.

For QUADR, the tests are run with the default value of the shell normal equal to 20.0. Except for the spherical shell (14), the shell normal has no effect on the tests. For the spherical shell, it will score an A if shell normal is turned off (PARAM, SNORM, 0.0.)

Table 3-10 MacNeal and Harder Tests for QUADR, QUAD4, and QUAD8

MacNeal/Harder Test Description	Element Loading		Element Shapes	QUADR	QUAD4	QUAD8
	In-Plane	Out-of-Plane				
1. Patch Test	X		Irregular	A	A	C
2. Patch Test		X	Irregular	A	A	D
3. Straight Beam, Extension	X		All	A	A	A
4. Straight Beam, Bending	X		Regular	A	A	A
5. Straight Beam, Bending	X		Irregular	B	F	B

Table 3-10 MacNeal and Harder Tests for QUADR, QUAD4, and QUAD8

MacNeal/Harder Test Description	Element Loading		Element Shapes	QUADR	QUAD4	QUAD8
	In-Plane	Out-of-Plane				
6. Straight Beam, Bending		X	Regular	A	A	A
7. Straight Beam, Bending		X	Irregular	A	A	A
8. Straight Beam, Twist	X	X	All	B	B	B
9. Curved Beam	X		Regular	A	C	A
10. Curved Beam		X	Regular	B	B	B
11. Twisted Beam	X	X	Regular	A	A	A
12. Rectangular Plate (N=4)		X	Regular	A	B	B
13. Scordelis-Lo Roof (N=4)	X	X	Regular	B	B	A
14. Spherical Shell (N=8)	X	X	Regular	B	A	C
15. Thick Walled Cylinder	X		Regular	B	B	B

3.3.4 Conical Shell Element (RINGAX)

The properties of the conical shell element are assumed to be symmetrical with respect to the axis of the shell. However, the loads and deflections need not be axisymmetric because they are expanded in Fourier series with respect to the azimuthal coordinate. Due to symmetry, the resulting load and deformation systems for different harmonic orders are independent, a fact that results in a large time saving when the use of the conical shell element is compared with an equivalent model constructed from plate elements. Theoretical aspects of the conical shell element are provided in Section 5.9 of *The NASTRAN Theoretical Manual*.

The conical shell element cannot be combined with other types of elements. This is primarily because the Fourier coefficients are stored on internally generated pseudo grid points. The unconventional nature of this element results in its capabilities being

limited, as summarized in **Table 3-1**. The existence of a conical shell problem is defined by the **AXIC** entry. This entry also indicates the number of harmonics desired in the problem formulation. Only a limited number of Bulk Data entries are allowed when using conical shell elements. The list of allowable entries is given on “**AXIC**” on page 942 of the . This element is not allowed in the superelement solution sequences.

The geometry of a problem using the conical shell element is described with **RINGAX** entries instead of **GRID** entries. The **RINGAX** entries describe concentric circles about the basic z -axis, with their locations given by radii and z -coordinates as shown in **Figure 3-40**. The degrees-of-freedom defined by each **RINGAX** entry are the Fourier coefficients of the motion with respect to angular position around the circle. For example, the radial motion u_r at any angle ϕ is described by the equation

$$u_r(\phi) = \sum_{n=0}^N u_r^n \cos n\phi + \sum_{n=0}^N u_r^{n*} \sin n\phi \quad \text{Eq. 3-26}$$

where u_r^n and u_r^{n*} are the Fourier coefficients of radial motion for the n -th harmonic. For calculation purposes, the series is limited to N harmonics as defined by the **AXIC** entry. The first sum in the above equation describes symmetric motion with respect to the ϕ plane. The second sum with the “starred” (*) superscripts describes the antisymmetric motion. Thus each **RINGAX** entry will produce six times N degrees-of-freedom for each series.

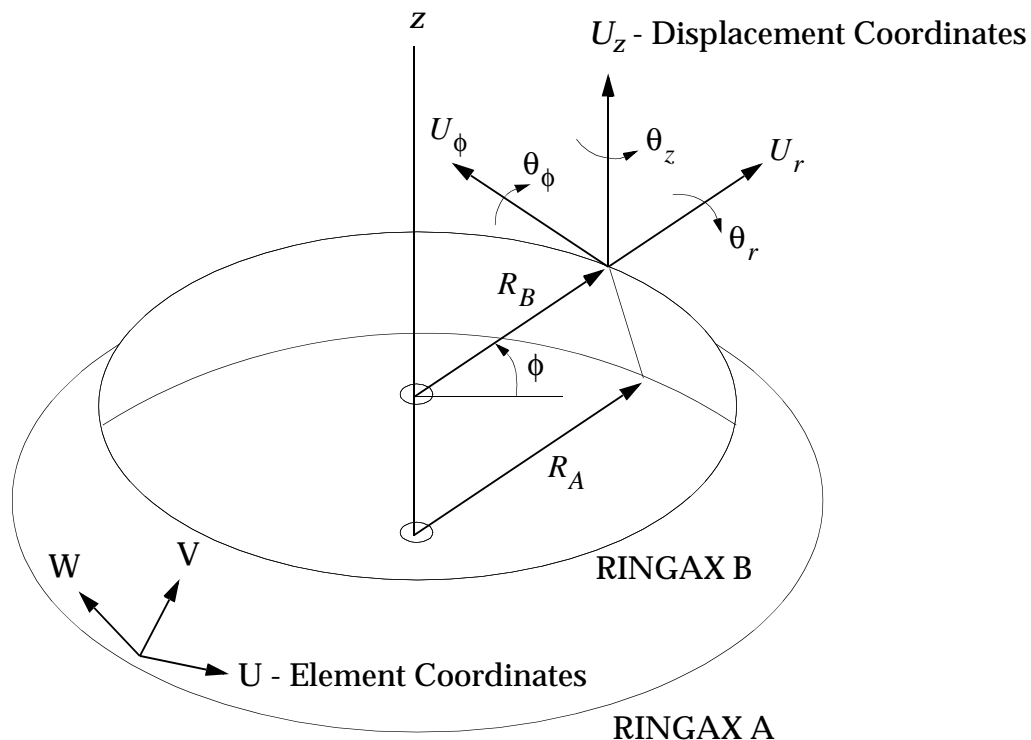


Figure 3-40 Geometry for Conical Shell Element

The selection of symmetric or antisymmetric solutions is controlled by the AXISYM command in the Case Control Section. For general loading conditions, a combination of the symmetric and antisymmetric solutions must be made, using the Case Control command, “**SYMCOM**” on page 450 of the *MSC.Nastran Quick Reference Guide*. If the AXISYM command is not present, stress and element force requests will be ignored.

Since the user is rarely interested in applying loads in terms of Fourier harmonics and interpreting data by manually performing the above summations, special data entries which automatically perform these operations are provided. The POINTAX entry is used like a GRID entry to define physical points on the structure for loading and output. Sections of the circle may be defined by a SECTAX entry which defines a sector with two angles and a referenced RINGAX entry. The POINTAX and SECTAX entries define six degrees-of-freedom each. The basic coordinate system for these points is a cylindrical system (r, ϕ, z) , and their applied loads must be described in this coordinate system. Since the displacements of these points are dependent on the harmonic motions, they may not be constrained in any manner.

The conical shell element is connected to two RINGAX rings with a CCONEAX entry. The properties of the conical shell element are described on the PCONEAX entry. The RINGAX points must be placed on the neutral surface of the element, and the points for stress calculation must be given on the PCONEAX entry relative to the neutral surface. Up to fourteen angular positions around the element may be specified for stress and force output. These values will be calculated midway between the two connected rings.

The structure defined with RINGAX and CCONEAX entries must be constrained in a special manner. All harmonics may be constrained for particular degree-of-freedom on a ring by using permanent single-point constraints on the RINGAX entries. Specified harmonics of each degree-of-freedom on a ring may be constrained with an SPCAX entry. The entry is the same as the SPC entry except that a harmonic must be specified. The MPCAX, OMITAX, and SUPAX data entries correspond to the MPC, OMIT, and SUPORT data entries except that harmonics must be specified. SPCADD and MPCADD entries may be used to combine constraint sets in the usual manner.

The stiffness matrix includes five degrees-of-freedom per grid circle per harmonic when transverse shear flexibility is included. Since the rotation about the normal to the surface is not included, the sixth degree-of-freedom must be constrained to zero when the angle between the meridional generators of two adjacent elements is zero. Since only four independent degrees-of-freedom are used when the transverse shear

flexibility is not included, the fifth and sixth degrees-of-freedom must be constrained to zero for all rings. These constraints can be conveniently specified on the RINGAX entry.

The conical shell structure may be loaded in various ways. Concentrated forces may be described by FORCE and MOMENT entries applied to POINTAX points. Pressure loads may be input by the PRESAX data entry which defines an area bounded by two rings and two angles. Temperature fields are described by a paired list of angles and temperatures around a ring as required by the TEMPAX entry. Direct loads on the harmonics of a RINGAX point are given by the FORCEAX and MOMAX entries. Since the basic coordinate system is cylindrical, the loads are given in the r , ϕ , and z directions. The value of a harmonic load F_n is the total load on the whole ring of radius r . If a sinusoidal load-per-unit length of maximum value is given, the value on the FORCEAX entry must be

$$F_n = 2\pi r a_n \quad n = 0 \quad \text{Eq. 3-27}$$

$$F_n = \pi r a_n \quad n = 0 \quad \text{Eq. 3-28}$$

Displacements of rings and forces in conical shell elements can be requested in two ways:

- The harmonic coefficients of displacements on a ring or forces in a conical element
- The displacements at specified points or the average value over a specified sector of a ring. The forces in the element at specified azimuths or average values over specified sectors or a conical element.

Harmonic output is requested by ring number for displacements and conical shell element number for element forces. The number of harmonics that will be output for any request is a constant for any single execution. This number is controlled by the Case Control command, “**HARMONICS**” on page 293 of the *MSC.Nastran Quick Reference Guide*.

The following element forces per unit of width are output either as harmonic coefficients or at specified locations on request

- Bending moments on the u and v faces
- Twisting moments
- Shearing forces on the u and v faces.

The following element stresses are calculated at two specified points on the cross section of the element and output as harmonic coefficients or at specified locations on request:

- Normal stresses in u and v directions
- Shearing stress on the u face in the v direction
- Angle between the u -axis and the major principal axis
- Major and minor principal stresses
- Maximum shear stress

3.3.5 Shell Elements Corner Thickness Specification

Introduction

The CQUAD4, CQUAD8, CQUADR, CTRIA3, CTRIA6, and CTRIAR elements allow the user to specify either corner element thickness or a single thickness on the PSHELL entry. A TFLAG field has been added to the connection entries of each of the above shell elements that allows the user to specify the corner thickness as a fraction relative to the T value of the PSHELL entry.

Benefits

In forming shell like structures, quite often the taper at the corners is really just a function of some average thickness. With this option, the average thickness can be specified as T on the PSHELL and the corner thickness will be computed as a fraction of the T on the PSHELL. In SOL 200, the T on the PSHELL can be specified on the DVPREL1 with average thickness specified using a DESVAR entry and the shell will be tapered accordingly.

Theory

If TFLAG=1 the following calculation is used for each corner:

$$T_i = T_i \times T$$

Inputs

If the fraction relative to the T value of the PSHELL is desired, TFLAG=1 should be specified on the element connection entry.

Outputs

Standard shell element output is obtained.

Guidelines and Limitations

This option should be useful in design optimization.

Example

The following example input demonstrates the form and use of the new entries:

Input File

```
diag 8,15,56
sol 200
cend
$
title = Cantilever Plate: Optimization
subtitle = Relative Thickness
$
analysis = statics
$
desobj = 1000
dessub = 2000
$
dsaprt
$
load = 10
spc = 20
$
disp = all
stress = all
$
begin bulk
$
param, grdpnt, 0
$
dresp1, 1000, weight, weight
$
dresp1, 2000, disp, disp,,, 2,, 5
dconstr, 2000, 2000, -0.5
$
doptprm, desmax, 15
$
$ Design Model: Relative Thickness
$
desvar, 1, t1, 1., .001
desvar, 2, t2, 1., .001
desvar, 3, t3, 1., .001
$
dvpres1, 100, pshell, 1, t
```

```
      , 1, 0.1
dvpshell, 200, pshell, 2, t
      , 2, 0.2
dvpshell, 300, pshell, 3, t
      , 3, 0.1

$
$ Test Plate: Sheet Metal Thickness Variation
$
$ Relative Thicknesses on CQUAD4/CTRIA3 Entries
$
$ -----
$
grid, 1,, 0., 0., 0.
= ,*(1),, *(1.), ==
=(3)
grid, 11,, 0., 1., 0.
= ,*(1),, *(1.), ==
=(2)
grid, 21,, 0., 2., 0.
= ,*(1),, *(1.), ==
=(3)
$
cquad4, 1, 1, 1, 2, 12, 11,
, , 1, 1., 1.1, 1.1, 1.
cquad4, 2, 1, 2, 3, 13, 12,
, , 1, 1.1, 1.2, 1.3, 1.1
cquad4, 3, 1, 3, 4, 14, 13,
, , 1, 1.2, 1.1, 1.6, 1.3
$
pshell, 1, 1, 0.1, 1
$
cquad4, 11, 2, 11, 12, 22, 21
, , 1, 1.0, 1.1, 1.1, 1.0
cquad4, 12, 2, 12, 13, 23, 22
, , 1, 1.1, 1.0, 1.2, 1.1
cquad4, 13, 2, 13, 14, 24, 23
, , 1, 1.0, 0.8, 1.1, 1.2
$
pshell, 2, 1, 0.2, 1
$
ctria3, 21, 3, 4, 5, 14
, , 1, 1.1, 1.0, 1.6
ctria3, 22, 3, 5, 25, 14
, , 1, 1.0, 2.0, 1.6
ctria3, 23, 3, 14, 25, 24
, , 1, 1.6, 2.0, 2.2
$
pshell, 3, 1, 0.1, 1
$
mat1, 1, 10000.,, 0.3, 1.
$
force, 10, 5,, 10., 0., -1.
force, 10, 25,, 10., 0., -1.
$
```

```

spc1, 20, 12, 1, 11, 21
spc1, 20, 3456, 1, thru, 25
$
enddata

```

Partial Output File

```

SUMMARY OF DESIGN CYCLE HISTORY
*****

(HARD CONVERGENCE ACHIEVED)

(SOFT CONVERGENCE ACHIEVED)

NUMBER OF FINITE ELEMENT ANALYSES COMPLETED      5
NUMBER OF OPTIMIZATIONS W.R.T. APPROXIMATE MODELS  4

OBJECTIVE AND MAXIMUM CONSTRAINT HISTORY
-----

```

CYCLE NUMBER	OBJECTIVE FROM APPROXIMATE OPTIMIZATION	OBJECTIVE FROM EXACT ANALYSIS	FRACTIONAL ERROR OF APPROXIMATION	MAXIMUM VALUE OF CONSTRAINT
INITIAL		1.299167E+00		-4.583377E-02
1	1.088284E+00	1.088204E+00	7.350594E-05	-1.224685E-02
2	1.067696E+00	1.067703E+00	-6.364063E-06	1.266837E-03
3	1.065613E+00	1.065610E+00	2.349259E-06	2.949357E-03
4	1.065610E+00	1.065610E+00	0.000000E+00	2.949357E-03

```

*
*****
1 CANTILEVER PLATE: OPTIMIZATION MARCH 18, 2003 MSC.NASTRAN 3/18/03
PAGE 40
RELATIVE THICKNESS
0
DESIGN VARIABLE HISTORY
-----
INTERNAL | EXTERNAL | LABEL | INITIAL : 1 : 2 : 3 : 4 :
DV. ID. | DV. ID. |      |          :   :   :   :   :
5 :
-----
1 | 1 | T1 | 1.0000E+00 : 1.2538E+00 : 1.3250E+00 : 1.3398E+00 : 1.3398E+00 :
2 | 2 | T2 | 1.0000E+00 : 8.2646E-01 : 7.5902E-01 : 7.5085E-01 : 7.5085E-01 :
3 | 3 | T3 | 1.0000E+00 : 3.8964E-01 : 3.8075E-01 : 3.7388E-01 : 3.7388E-01 :
*** USER INFORMATION MESSAGE 6464 (DOM12E)
RUN TERMINATED DUE TO HARD CONVERGENCE TO AN OPTIMUM AT CYCLE NUMBER = 4.

```

3.3.6 SHELL Structural Damping

If PARAM,SHLDAMP,DIFF is present in the run, a structural damping matrix κ^4 is formed for each shell element based on a relationship of the following form:

$$\begin{aligned} K^A &= g e_{mid1} \times \text{membrane} \cdot \text{stiffness} \\ &+ g e_{mid2} \times \text{bending} \cdot \text{stiffness} \\ &+ g e_{mid3} \times \text{transverse} \cdot \text{stiffness} \\ &+ g e_{mid4} \times \text{combined} \cdot \text{stiffness} \end{aligned}$$

3.4 Solid Elements (CTETRA, CPENTA, CHEXA)

MSC.Nastran includes three different solid polyhedron elements which are defined on the following Bulk Data entries:

1. CTETRA -- Four-sided solid element with 4 to 10 grid points. See [Figure 3-41](#).
2. CPENTA -- Five-sided solid element with 6 to 15 grid points. See [Figure 3-42](#).
3. CHEXA -- Six-sided solid element with 8 to 20 grid points. See [Figure 3-43](#).

These elements can be used with all other MSC.Nastran elements except the isoparametric elements. Constraints can apply only to displacement degrees of freedom at the grid points.

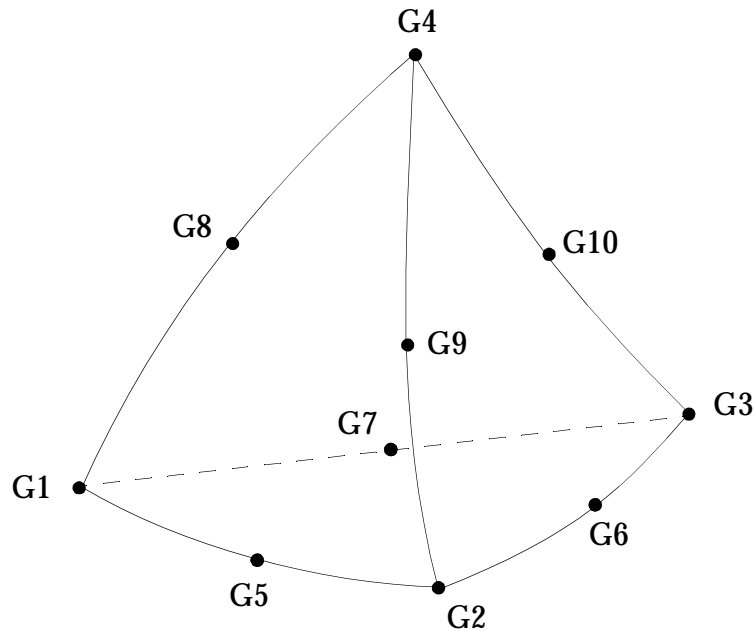


Figure 3-41 The CTETRA Element

Three types of integration are available for the CPENTA and CHEXA elements: reduced shear integration with and without the use of bubble functions and standard isoparametric integration. Only standard isoparametric integration is available for the CTETRA element. The type of integration is selected on the PSOLID entry.

Reduced shear integration minimizes shear locking and will not cause zero energy modes. If reduced shear integration is selected with bubble functions, it minimizes Poisson's ratio locking which occurs in nearly incompressible materials and in elements under bending. In plastic analysis, bubble functions are necessary to reduce locking caused by the plastic part in the material law. Therefore, reduced shear integration with bubble functions is the default option for the CPENTA and CHEXA elements. However, the use of bubble functions requires more computational effort.

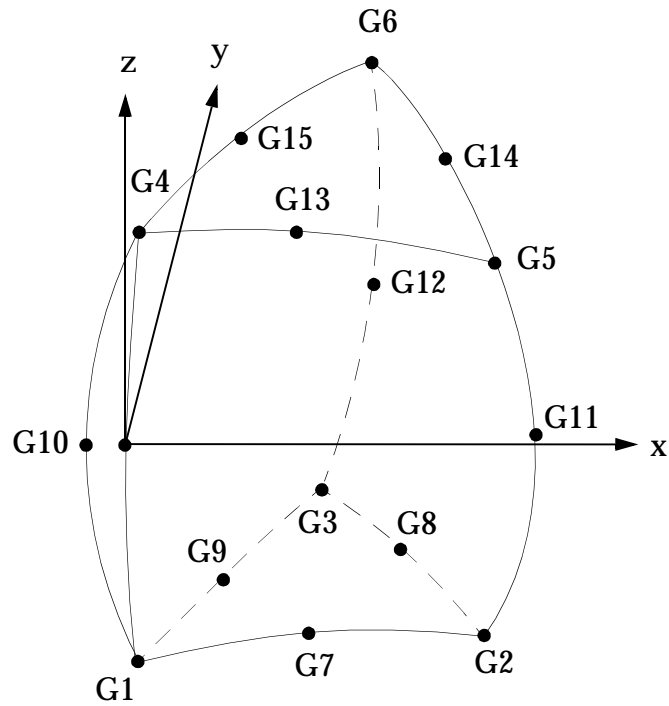


Figure 3-42 The CPENTA Element

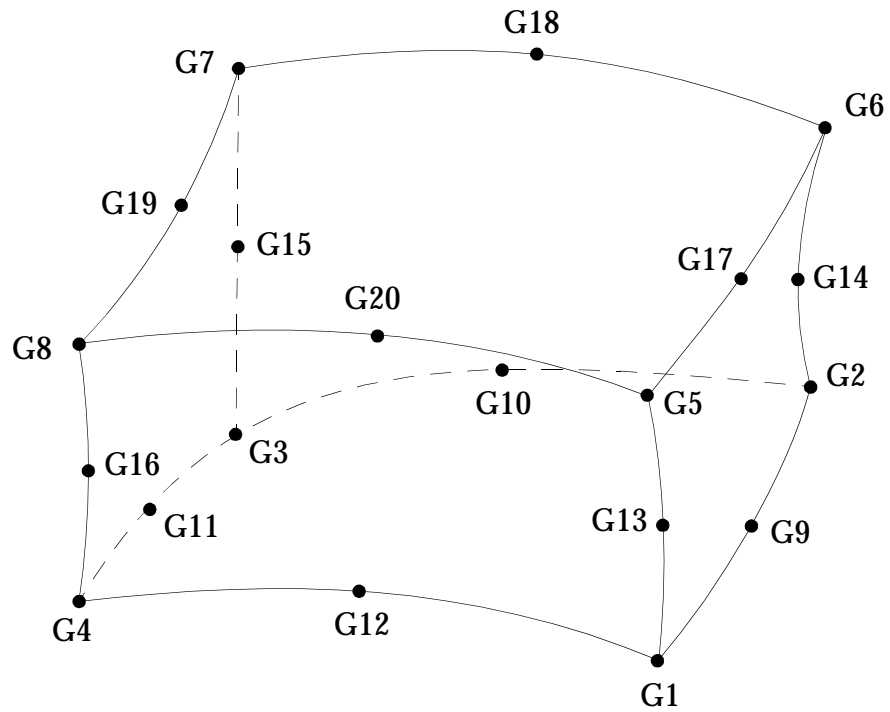


Figure 3-43 The CHEXA Element

In standard isoparametric integration, the number of Gauss points, or integration network, may be changed to underintegrate or overintegrate the solid elements. Underintegration may cause zero energy modes and overintegration results in an element which may be too stiff. Standard isoparametric integration is more suited to nonstructural problems.

The element coordinate systems are described under the description of the CHEXA, CPENTA, and CTETRA entries, respectively. These elements may use anisotropic materials as defined on the MAT9 Bulk Data entry. The material reference and integration network is specified on the PSOLID entry.

Structural mass is calculated for all solid elements. The default mass procedure is lumped mass. The coupled mass procedure may be requested with PARAM, “**COUPMASS**” on page 614 of the *MSC.Nastran Quick Reference Guide*.

The solid elements can be used to define fluid elements for coupled fluid-structural analysis. See “**Coupled Fluid-Structure Interaction**” on page 753.

The solid elements can be used in differential stiffness analysis and buckling analysis. A geometric and material nonlinear stiffness formulation is available for these elements only if there are no midside nodes.

The following stresses and strains are output on request:

- Normal: $\sigma_x, \sigma_y, \sigma_z$, and $\varepsilon_x, \varepsilon_y, \varepsilon_z$
- Shear: $\tau_{xy}, \tau_{yz}, \tau_{zx}$, and $\gamma_{xy}, \gamma_{yz}, \gamma_{zx}$
- Principal with magnitude and direction
- Mean pressure $p_o = \frac{1}{3}(\sigma_x + \sigma_y + \sigma_z)$
- Octahedral shear stress or

$$\bar{\tau} = \frac{1}{3}[(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 + 6\tau_{yz}^2 + 6\tau_{zx}^2 + 6\tau_{xy}^2]^{1/2} \quad \text{Eq. 3-29}$$

the von Mises equivalent stress

$$\bar{\tau}_v = \left(\frac{3}{\sqrt{2}} \right) \bar{\tau}_o \quad \text{Eq. 3-30}$$

- Octahedral shear strain or

$$\bar{\varepsilon}_o = \left[\frac{1}{9} \left\{ (\varepsilon_x - \varepsilon_y)^2 + (\varepsilon_y - \varepsilon_z)^2 + (\varepsilon_z - \varepsilon_x)^2 \right\} + \frac{1}{6} (\gamma_{xy}^2 + \gamma_{yz}^2 + \gamma_{zx}^2) \right]^{1/2} \quad \text{Eq. 3-31}$$

the von Mises equivalent strain

$$\bar{\varepsilon}_v = \left[\frac{2}{9} \left\{ (\varepsilon_x - \varepsilon_y)^2 + (\varepsilon_y - \varepsilon_z)^2 + (\varepsilon_z - \varepsilon_x)^2 \right\} + \frac{1}{3} (\gamma_{xy}^2 + \gamma_{yz}^2 + \gamma_{zx}^2) \right]^{1/2} \quad \text{Eq. 3-32}$$

where the strain components are defined as

$$\varepsilon_x = \frac{\partial u}{\partial x}; \quad \varepsilon_y = \frac{\partial v}{\partial y}; \quad \varepsilon_z = \frac{\partial w}{\partial z} \quad \text{Eq. 3-33}$$

$$\gamma_{xy} = \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right); \quad \gamma_{yz} = \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right); \quad \gamma_{zx} = \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \quad \text{Eq. 3-34}$$

In nonlinear nonhyperelastic (small strain) analysis, the mechanical strain σ / E is output rather than the total strain defined above. The hyperelastic elements output total strains. See “**Fully Nonlinear Hyperelastic Elements**” on page 614.

By default, the stresses and strains are evaluated in the basic coordinate system at each of the corner points and the centroid of the element. The stresses and strains may also be computed in the material coordinate system as defined in the CORDM field on the PSOLID entry.

In addition, interpolated grid point stresses and mesh stress discontinuities are calculated in user-specified coordinate systems for grid points which connect solid elements. See “**Element Data Recovery Resolved at Grid Points**” on page 463 for a description of the Case Control Section to request grid point stresses and mesh stress discontinuities. Only real stresses are available at the grid points. Mesh stress discontinuities are available in linear static analysis only. For further discussion see the end of “**Shell Elements (CTRIA3, CTRIA6, CTRIAR, CQUAD4, CQUAD8, CQUADR)**” on page 131.

3.5 Rigid Elements and Multipoint Constraints (R-type, MPC)

The multipoint constraint, or MPC entry, provides the capability to model rigid bodies and to represent other relationships which can be treated as rigid constraints. The MPC entry provides considerable generality but lacks user convenience. Specifically, the user must supply all of the coefficients in the equations of constraint defined through the MPC entry.

To enhance user convenience, nine rigid body elements are available in MSC.Nastran. See [Table 3-11](#). These elements require only the specification of the degrees-of-freedom that are involved in the equations of constraint. All coefficients in these equations of constraint are calculated internally in MSC.Nastran.

Table 3-11 Rigid Element and MPC Entries

Name	Description	Finite Rotation	$m =$ Dependent Degrees-of-freedom
RROD	A open-ended rod which is rigid in extension	Y	$m = 1$
RBAR	Rigid bar with six degrees-of-freedom at each end.	Y	$1 \leq m \leq 6$
RJOINT	Rigid joint with six degrees-of-freedom at each end.	Y	$1 \leq m \leq 6$
RTRPLT	Rigid triangular plate with six degrees-of-freedom at each vertex.	Y	$1 \leq m \leq 12$
RBE2	A rigid body connected to an arbitrary number of grid points. The independent degrees-of-freedom are the six components of motion at a single grid point. The dependent degrees-of-freedom at the other grid points all have the same user-selected component numbers.	Y	$m \geq 1$

Table 3-11 Rigid Element and MPC Entries (continued)

Name	Description	Finite Rotation	$m =$ Dependent Degrees-of-freedom
RBE1	A rigid body connected to an arbitrary number of grid points. The independent and dependent degrees-of-freedom can be arbitrarily selected by the user.	Y	$m \geq 1$
RBE3	Defines a constraint relation in which the motion at a “reference” grid point is the least square weighted average of the motions at other grid points. The element is useful for “beaming” loads and masses from a “reference” grid point to a set of grid points.	Y	$1 \leq m \leq 6$
RSPLINE	Defines a constraint relation whose coefficients are derived from the deflections and slopes of a flexible tubular beam connected to the referenced grid points. This element is useful in changing mesh size in finite element models.	N	$m \geq 1$
RSSCON	Define a multipoint constraint relation which models a clamped connection between shell and solids.	N	$m \geq 5$
MPC	Rigid constraint that involves user-selected degrees-of-freedom at both grid points and at scalar points. The coefficients in the equation of constraint are computed and input by the user.	N	$m = 1$

Any combination of the above elements may be used in an MSC.Nastran analysis in any of the structural solution sequences, however, these elements should be used with care in geometric nonlinear analysis (see the *MSC.Nastran Handbook for Nonlinear*

Analysis). The nine rigid elements are ignored in the heat transfer solution sequences. See “**Rigid Element Enhancements**” on page 173 for nonlinear enhancements to the rigid element.

User Input

A Bulk Data entry is associated with each of the nine rigid elements (see the “**Bulk Data Entries**” on page 849 of the *MSC.Nastran Quick Reference Guide*). Before proceeding with a discussion of input requirements, it is important to understand how MPC and rigid element entries are treated.

Each MPC entry generates a single equation of the form

$$A_1 u_1 + A_2 u_2 + \dots A_n u_n = 0 \quad \text{Eq. 3-35}$$

where u_1, u_2, \dots, u_n are user-designated degrees-of-freedom (grid point plus component) and A_1, A_2, \dots, A_n are coefficients which are user-supplied. The first named degree-of-freedom is placed in the u_m set (degrees-of-freedom eliminated by multipoint constraints). A rigid element (RRD, RBAR, RTRPLT, RBE1, RBE3, RSPLINE, RBE2, RJOINT, or RSSCON) generates a number of equations of the form of **Eq. 3-35**. The user designates the degrees-of-freedom to be placed in the u_m set (one for each equation of constraint). In some cases (RBAR and RTRPLT), the designation can be accomplished by a default option. The coefficients in **Eq. 3-35** are computed automatically from geometric data and for the RBE3 element. For the RSSCON, the M-set degrees-of-freedom are automatically assigned. The complete u_m set consists of the first named terms on the MPC entries plus the designated degrees-of-freedom on the rigid element entries. The user has complete control over the membership of this set.

Input Requirements

The following table lists the procedural requirements and rules that the user must follow when using MPCs and the (RBAR, RTRPAT, RBE1, RBE2, RBE3, RSSCON, and RSPLINE) rigid elements in an analysis.

1. A member of the u_m set cannot also be a member of any other user defined set. User-defined sets include:
 - u_s -- degrees-of-freedom eliminated by SPCi entries, AUTOSPC, and PS field on GRID entries
 - u_o -- degrees-of-freedom specified on OMITi Bulk Data entries
 - u_r -- degrees-of-freedom specified on SUPORT and SUPORT1 Bulk Data entries

u_a -- members of the analysis set specified on ASETi Bulk Data entries or exterior degrees-of-freedom in superelement analysis

2. A degree-of-freedom cannot be designated as a member of the u_m set more than once.

A fatal error will result if, for example, the same degree-of-freedom is designated as dependent by two rigid elements or it the first-named degree-of-freedom on any MPC entry is also a designated member of u_m on a rigid element entry.

3. The user-selected independent degrees-of-freedom u_n , for the RBAR, RTRPLT, RBE1, RBE2, RBE3 and RSPLINE rigid elements must be sufficient to define any general rigid body motion of the element.

These degrees-of-freedom are independent only for the particular element, and they may be declared dependent by other rigid element, MPCs, SPCs, OMITs, or SUPORTs. As far as a particular rigid element is concerned, it is always acceptable to select all six independent degrees-of-freedom at one grid point. This may not, however, be a good choice when the total problem requirements are considered.

For six of the elements under discussion (RBAR, RTRPLT, RBE1, RBE3, RSPLINE, and RBE2), the user lists the degrees-of-freedom in u_m and u_n . The remaining degrees-of-freedom at the grid points to which the rigid element is jointed are not involved with the rigid element. This lack of connection represents either a sliding or rotating joint release, or both. The rigid rod element (RROD) is an exception because once a component of translation is placed in u_m , all of the five remaining components of translation will automatically be placed in u_n . The rotational degrees-of-freedom are not involved in the RROD element.

4. The user must avoid over-constraining the structure when two or more rigid elements are used.

A structure is over-constrained when the degrees-of-freedom, which remain after the members of u_m have been selected, are insufficient to represent a general rigid body motion of the structure as a whole.

Consider, for example, a number of RBAR elements connected together to form a rigid ring. Let the grid points be numbered from 1 to N and assume that the u_m degrees-of-freedom for each rigid element are placed at the higher numbered grid point so that the only degrees-of-freedom which

remain independent as each element is added to the ring are those at grid point 1. The addition of the last rigid element between grid points N and 1 will remove even those independent degrees-of-freedom and thereby over-constrain the structure.

5. For the RSSCON, the shell degrees-of-freedom are placed in u_m . The translational degrees-of-freedom of the shell edge are connected to the translational degrees-of-freedom of the upper and lower edge of the solid. The shell's two rotational degrees-of-freedom are also connected to the translational degrees-of-freedom of the upper and lower edge of the solid. The RSSCON only impresses a rigid constraint on the shell's two rotational degrees-of-freedom.
6. Nonlinear forces in dynamic analysis cannot be applied to members of the u_m set.

The RBE3 entry (see “**RBE3**” on page 1949 of the) defines a reference grid point (REFGRID, field 4) and connected grid points and components (Gij, Ci) in subsequent fields. If only the translational degrees-of-freedom are listed for the connected grid points, the behavior of the element in MSC.Nastran 2001 (and Version 70.7) is unchanged from that of Version 70.5. The element theory has been modified with result changes when rotations of the connected grid points (Ci=4, 5, 6) are included. The reference grid point theory, on the other hand, remains unchanged; there are no new (or even old) restrictions on reference grid point rotations. Changes for results are avoided in Version 70.7 and MSC.Nastran 2001 if translations rather than rotations are used on the connected grid points.

However, several third-party graphical user interfaces (GUIs) apparently do connect these rotational degrees-of-freedom by default, which leads to some cases of substantial differences in answers and resulting in subsequent client support calls. We recommend that the default degrees-of-freedom for the connected grid points be set to the translational degrees-of-freedom only (i = 1, 2, 3). Again, no such restriction exists for the reference grid.

In a limited number of contexts it makes modeling sense to connect these rotational degrees-of-freedom. For example, when all of the connected points are colinear, the element will be unstable for rotations about this axis. A rotation component on one of the connected grid points that has a component parallel to the line of connected points will solve this problem. The results will then be the same on all systems new and old, because this is a statically determinate load path, and there is only one (automatically adjusted) coefficient that will provide equilibrium.

If there are many connected grid points spread over a surface or volume and one compares the results first with no connected rotations, and then with all rotations connected one often finds that the more highly connected model is actually softer. (See TAN 4800, where adding rotations to connectivity lowered the natural mode frequencies of a component of a large model.) While this is counter-intuitive for flexible finite elements, it is to be expected for spline constraint elements such as the RBE3. If more points are connected and they have lower stiffness than the originally connected points, the overall stiffness of the reference point is dominated by the softer connected rotational stiffness effects. The net effect is that connecting more degrees-of-freedom often reduces the reference point stiffness, rather than increasing it as one would expect.

The following is a summary of Technical Application Notes on this subject:

TAN Number	Title
2402	RBE3 - THE INTERPOLATION ELEMENT
3280	RBE3 ELEMENT CHANGES IN VERSION 70.5, IMPROVED DIAGNOSTICS
4155	RBE3 ELEMENT CHANGES IN VERSION 70.7, NONDIMENSIONALITY
4494	MATHEMATICAL SPECIFICATION OF THE MODERN RBE3 ELEMENT
4497	AN ECONOMICAL METHOD TO EVALUATE RBE3 ELEMENTS IN LARGE-SIZE MODELS
4800	THE EFFECTS OF CONNECTING ROTATIONS TO RBE3 ELEMENTS [In preparation]
4801	RELUMPING NON-STRUCTURAL MASSES WITH CONM2, RBE3 ENTRIES [In preparation]

3.5.1 Rigid Element Enhancements

Introduction

Rigid elements, such as RBAR, RBE1, ... etc., have been available in MSC.Nastran since the inception of MSC.Nastran. However, they contain many limitations, which are discussed in the section “Types of Rigid Element” below.

In MSC.Nastran 2004, a new type of rigid elements is introduced, which is called the Lagrange rigid element. For the Lagrange rigid elements, the limitations for the existing rigid elements are removed. These are also discussed in the section “Type of Rigid Element” below.

In the “Theory” section, first, an introduction is given to the formulation of large rotation theory used by the Lagrange rigid element, followed by a summary of theoretical formulation for each type of the rigid element is given.

Limitations for the initial MSC.Nastran 2004 SOL 400 are discussed in the “Limitations on SOL 400” section.

Finally, the “User Interface” and “Examples” sections demonstrate the usage of the new Lagrange rigid elements.

Types of Rigid Element

In MSC.Nastran 2004, two types of rigid elements are available -- the linear rigid element and the Lagrange rigid element.

Linear Rigid Elements

Linear rigid elements have been available since the inception of MSC.Nastran. The linear rigid elements use the linear elimination technique and are not “real” finite elements. Instead, they are internally represented by a set of MPC equations. By using these MPC equations, the dependent degrees-of-freedom (the m-set) are eliminated from the solution set (the l-set). These types of rigid elements have the following limitations. They:

- Do not compute thermal load.
- Do not have differential stiffness matrix so that the solutions are incorrect for the buckling analysis or other solution sequences where the differential stiffness matrix is required.
- Use the small rotation theory in the geometrical nonlinear analysis so that the solutions are incorrect in this type of analysis.
- Use the elimination method for solution resulting in very dense stiffness matrices. These dense matrices cannot take advantages of the sparse matrix algorithm.

The Lagrange rigid element, which is discussed in next sub-section, does not have the above limitations.

Lagrange Rigid Elements

In MSC.Nastran 2004, a new type of rigid element, called the Lagrange rigid element, is introduced. These Lagrange rigid elements are “real” finite elements, similar to, for example, QUAD4 element. Instead of using MPC equations, the element stiffness matrix is computed for each Lagrange rigid element. All the limitations for the linear rigid element are removed, i.e., they

- Include thermal load effect.
- Include the differential stiffness.
- Support for both the small rotation and the large rotation. Use the large rotation theory in the geometrical nonlinear analysis (PARAM,LDISP,1).
- Take advantages of the sparse matrix algorithm if the augmented Lagrange multiplier method, defined below, is used for solution.

For each Lagrange rigid element, a number of Lagrange multiplier degrees-of-freedom are created by MSC.Nastran internally. For example, one to six Lagrange multiplier degrees-of-freedom are created for RBAR and one Lagrange multiplier degree-of-freedom is created for RROD. For each Lagrange rigid element, the independent degrees-of-freedom, the dependent degrees-of-freedom, and the Lagrange multiplier degrees-of-freedom are first placed in the solution set (the l-set). Two methods are available for solution:

Augmented Lagrange Multiplier Method

For this method, the solution is obtained with the independent degrees-of-freedom, the dependent degrees-of-freedom, and the Lagrange multipliers degrees-of-freedom in the solution set. Thus, the sparse characteristic of the stiffness matrix is maintained and the sparse matrix algorithm can be utilized.

Lagrange Elimination Method

For this method, additional operations of eliminating both the dependent and Lagrange multiplier degrees-of-freedom from the solution set are performed after the global stiffness matrix is assembled. The solution is performed on the independent set (the n-set), creating dense matrices after elimination. Once again, these dense matrices cannot take advantage of the efficiency features of the sparse method.

Method Selection

For most problems, the augmented Lagrange multiplier method is the preferred method. The tangent stiffness matrix using this method, however, is not positive definite, and can potentially pose numerical difficulties. It is for this reason that the Lagrange eliminator method is introduced as a backup method.

For linear rigid elements, the dependent degrees-of-freedom are eliminated and placed in the mp-set. For the augmented Lagrange multiplier rigid elements, both the dependent degrees-of-freedom and the internally created Lagrange multipliers are placed in the l-set. For the Lagrange elimination rigid elements, both the dependent degrees-of-freedom and the Lagrange multipliers are eliminated and placed in the mr-set. The mp-set and the mr-set are subsets of the m-set. Please see “[Degree-of-Freedom Set Definitions](#)” on page 844 of the *MSC.Nastran Quick Reference Guide*.

The Lagrange elements are only available for linear static analysis (SOL 101), normal modes analysis (SOL 103), buckling analysis (SOL 105), and general nonlinear analysis (SOL 400). Furthermore, the Lagrange elimination method is not implemented for the following reasons:

- For nonlinear analysis, the tangent stiffness matrix must be updated at each nonlinear iteration for the Lagrange elimination method. (See “[The Lagrange Elimination Method](#)” on page 182.) This implies that only the full Newton-Raphson method can be used, which conflicts with the general iteration solution scheme employed in MSC.Nastran.
- The denseness of the tangent stiffness matrix after elimination poses an inefficiency problem for the sparse solver.
- For nonlinear analysis, only stiffness matrix decomposition is required and the eigenvalue analysis is not required. For stiffness matrix decomposition, the method for solving a non-positive definite matrix is well established, and hence removes the need for a backup method for the Lagrange multiplier method.

For MSC.Nastran 2004, there are some additional limitations for SOL 400. Please see “[Limitations on SOL 400](#)” on page 188.

Theory

In this section, we give introduction to the theoretical formulation of the rigid elements. Rigid elements use large rotation theory and hence warrant a discussion of large rotation. In the following sections, the method for representing large rotation for rigid elements is presented first, followed by the solution methods for the linear

elimination method, the Lagrange multiplier method, and the Lagrange multiplier method with elimination. Also, only the solution methods for the nonlinear analysis are discussed.

Rotation Vector

Mathematically, there are many ways of representing large rotation; the most commonly used methods are Euler angles, Gimbal angles, Rodrigues parameters, quaternion (Euler parameters), and rotation vector methods. MSC.Nastran implemented the Gimbal angles and the rotation vector method into the geometric nonlinear analysis. The Gimbal angles have inherent limitations. They are non-invariant parameters, which means that their values are intimately associated with the coordinate system used. On the other hand, the rotation vector gives a simple geometric meaning of rotations and it is an invariant parameter. Thus, the Lagrange elements use the rotation vector for representing large rotations, which is described in the next paragraph. Please note that the Lagrange elements do not have an option to use the Gimbal angles.

The most general motions of a rigid body can be represented by a translation of a base point and rotation about the base point. Therefore, the motion of a GRID point rigid body can be decomposed into a translation of the GRID point and a rotation defined by:

$$\Psi = \psi \mathbf{p}$$

where:

$$\psi = \|\Psi\| = (\Psi^T \Psi)^{\frac{1}{2}} \quad \text{Eq. 3-36}$$

$$\mathbf{p} = \frac{\Psi}{\psi} = \begin{Bmatrix} p_x \\ p_y \\ p_z \end{Bmatrix}$$

Thus ψ is the magnitude or the angle of the rotation and \mathbf{p} is the direction of the rotation.

It is well known that large rotations cannot be treated as vectors. Consequently, the assumption that rotations are linear, which is one of the basic assumptions of linear analysis, does not provide a unique solution when a series of large rotations is applied in different sequences. However, the effect of large rotation can be expressed in terms of the rotation matrix $\mathbf{R}(\Psi)$. Any vector \mathbf{v} can be rotated into new position \mathbf{v}_r given by the equation:

$$\mathbf{v}_r = \mathbf{R}(\Psi)\mathbf{v}$$

$$\mathbf{R}(\Psi) = \begin{bmatrix} \cos\Psi + p_x^2(1 - \cos\Psi) & -p_z \sin\Psi + p_x p_y(1 - \cos\Psi) & p_y \sin\Psi + p_x p_z(1 - \cos\Psi) \\ p_z \sin\Psi + p_x p_y(1 - \cos\Psi) & \cos\Psi + p_y^2(1 - \cos\Psi) & -p_x \sin\Psi + p_y p_z(1 - \cos\Psi) \\ -p_y \sin\Psi + p_x p_z(1 - \cos\Psi) & p_x \sin\Psi + p_y p_z(1 - \cos\Psi) & \cos\Psi + p_z^2(1 - \cos\Psi) \end{bmatrix} \quad \text{Eq. 3-37}$$

The rotational matrix is an orthogonal matrix, which has the property:

$$\mathbf{R}\mathbf{R}^T = \mathbf{I} \quad \text{or} \quad \mathbf{R}^{-1} = \mathbf{R}^T$$

A very important matrix associated with the rotation vector Ψ is the skew-symmetric matrix $\hat{\Psi}$, which is defined by:

$$\hat{\Psi} = \begin{bmatrix} 0 & -\Psi_z & \Psi_y \\ \Psi_z & 0 & -\Psi_x \\ -\Psi_y & \Psi_x & 0 \end{bmatrix} \quad \text{Eq. 3-38}$$

The rotation vector Ψ and its skew-symmetric matrix have the relationship:

$$\hat{\Psi}\Psi = \mathbf{0} \quad \text{and} \quad \hat{\Psi}\mathbf{v} = \Psi \times \mathbf{v} \quad \text{for any vector } \mathbf{v} \quad \text{Eq. 3-39}$$

Ψ is the axial vector of the skew-symmetric matrix $\hat{\Psi}$. In term of the skew-symmetric matrix and the unit vector \mathbf{p} of rotation, the rotation matrix can be rewritten as:

$$\mathbf{R}(\Psi) = \mathbf{p}\mathbf{p}^T + \cos\Psi(\mathbf{I} - \mathbf{p}\mathbf{p}^T) + \sin\Psi\hat{\mathbf{p}} \quad \text{Eq. 3-40}$$

where $\hat{\mathbf{p}}$ is the skew-symmetric matrix of the unit vector \mathbf{p} .

Composite Rotations -- Left Rotation or Right Rotation

A composite rotation is successive application of two or more rotations. For large rotation, this series of rotations is no longer additive. In addition, they are not commutative, i.e., the order of application of the rotation is important. For example, in a geometric nonlinear analysis, an incremental rotation $\Delta\theta_g$ is computed for a load increment or a nonlinear iteration. The composite rotation of $\Delta\theta_g$ and the previous rotation Ψ_i can be expressed by the equation:

$$\mathbf{R}_{i+1}(\Psi_{i+1}) = \Delta\mathbf{R}_l(\Delta\theta_g)\mathbf{R}_i(\Psi_i) \quad \text{Eq. 3-41}$$

Let us examine the coordinate system used for Eq. 3-41. Both Ψ_{i+1} and Ψ_i are computed in terms of the global coordinate system at the GRID point. Note that the global coordinate system at any GRID point is fixed in space. Here we must be very careful about the coordinate system used for the incremental rotation $\Delta\theta_g$. In order for Eq. 3-41 to be true, we must compute the incremental rotation in terms of the same coordinate system used by Ψ_{i+1} and Ψ_i , i.e., the global coordinate system at the GRID point. The corresponding incremental rotation matrix $\Delta\mathbf{R}_l$ computed is called **the left rotation**.

There is another way to compute the incremental rotation matrix. Let us rotate the global coordinate system at the GRID point by Ψ_i , i.e.

$$\mathbf{m}_x = \mathbf{R}_i(\Psi_i)\mathbf{g}_x \quad \text{Eq. 3-42}$$

where \mathbf{g}_x , $x = 1, 2, 3$, are the unit vectors of the global coordinate system \mathbf{g} at the GRID point and \mathbf{m}_x are the unit vectors of the rotated coordinate system \mathbf{m} . The coordinate system \mathbf{m} is called the material coordinate system at the GRID point. Please note that the material coordinate system is **not** fixed in the space and it is rotated with the GRID point. Now, let us compute the incremental rotation $\Delta\theta_m$ in terms of the material coordinate system. Then, the composite rotation of the incremental rotation and the previous rotation can be expressed by:

$$\mathbf{R}_{i+1}(\Psi_{i+1}) = \mathbf{R}_i(\Psi_i)\Delta\mathbf{R}_r(\Delta\theta_m) \quad \text{Eq. 3-43}$$

The rotation matrix $\Delta\mathbf{R}_r$ so computed is called **the right rotation**. It can be shown that the right rotation and the left rotation are related by the equations:

$$\Delta\theta_m = \mathbf{R}_i^T \Delta\theta_g \quad \text{Eq. 3-44}$$

$$\Delta\mathbf{R}_r = \mathbf{R}_i^T \Delta\mathbf{R}_l \mathbf{R}_i \quad \text{Eq. 3-45}$$

The right rotation has the following advantages:

- The representation of inertia forces is greatly simplified.
- The stiffness properties of each elastic member may be described in a rigorous manner, including the geometric stiffening effects.

To improve the geometric nonlinear analysis, the rigid elements are formulated in terms of the right rotation approach. However, all the existing elements, CQUAR4, CTRIA3, and CBEAM, are formulated in terms of the left rotation approach. In order to make the existing elements compatible with the rigid elements, the existing elements are modified in terms of the right rotation approach.

The Linear Elimination Method

For the linear elimination method, all rigid elements are reduced using MPC constraint equations given by:

$$\mathbf{u}_m = \mathbf{G}_m \mathbf{u}_n \quad \text{Eq. 3-46}$$

where \mathbf{u}_m are the dependent degrees-of-freedom, \mathbf{u}_n are the independent degrees-of-freedom, and \mathbf{G}_m is the constraint matrix. During the solution process, the dependent degrees-of-freedom are eliminated from the solution set (the I-set). This method is valid only for small rotation because the relationship given by Eq. 3-46 is linear. Also, the thermal loads and the differential stiffness matrix are not computed. For large rotation, the relationships between dependent degrees of freedom and independent degrees of freedom are nonlinear. This elimination method can be modified to solve the nonlinear rigid element problem. However, we use the augmented Lagrange multiplier method instead, which is discussed in next sub-section.

The Augmented Lagrange Multiplier Method

The equilibrium equation for a static nonlinear solution of a constrained structural model using the Lagrange multiplier method is given by:

$$\begin{aligned} \mathbf{F} + \mathbf{H}^T \lambda &= \mathbf{p} \\ \Phi(\mathbf{u}) &= \mathbf{q} \end{aligned} \quad \text{Eq. 3-47}$$

where \mathbf{F} is the internal force vector, λ is a vector of Lagrange multipliers, \mathbf{p} is the external force vector, \mathbf{u} is the displacement vector, Φ is a vector of nonlinear constraints on the displacements generated by the rigid elements, \mathbf{q} is the external enforced displacements or effect due to thermal loads, and \mathbf{H} is the displacement gradient given by:

$$\mathbf{H} = \frac{\partial}{\partial \mathbf{u}} (\Phi(\mathbf{u}) - \mathbf{q}) = \frac{\partial \Phi}{\partial \mathbf{u}} \quad \text{Eq. 3-48}$$

The explicit form for Φ and \mathbf{H} are dependent on the individual rigid element type. For example, the forms are different for the RBAR and the RBE3.

The equilibrium for Eq. 3-47 is obtained by minimizing the functional:

$$\Pi = \Pi_k + k(\Phi - \mathbf{q})^T \lambda + \frac{1}{2}(\Phi - \mathbf{q})^T p(\Phi - \mathbf{q}) \quad \text{Eq. 3-49}$$

The first term Π_k of Eq. 3-49 contains the regular strain energy and external work done to the system. For rigid elements, this term is zero. The second term is contribution due to the Lagrange multipliers, where k is a scale factor introduced to balance the magnitude of the Lagrange multipliers with other terms in the stiffness matrix. The last term is the added penalty function term and p is the penalty function. The reason for adding the penalty function term is to reduce numerical difficulty during matrix decomposition.

The constrained equilibrium equation can be also solved by the **penalty function method**. For this method, the magnitude of the penalty function must be very large in comparison to the tangential stiffness matrix in order to obtain an approximation solution of the original constrained equation. When the penalty function approaches infinity, the solution becomes the exact solution of the original constrained equation. The shortcoming of this method is that, when the penalty function approaches infinity, the tangential stiffness matrix becomes ill conditioned. On the other hand, in the augmented Lagrangian method as given by Eq. 3-49, the penalty function p needs not to be large. It only needs to be a similar magnitude as the other terms in the tangential stiffness matrix for enforcing the positive definiteness of the stiffness matrix.

For the nonlinear analysis, the next step is to derive the Newton-Raphson iteration equation from Eq. 3-49. Let $(\mathbf{u}^*, \lambda^*)$ be an approximation solution to Eq. 3-49. By using the Newton-Raphson method, a corrected solution can be obtained in the form:

$$(\mathbf{u}, \lambda) = (\mathbf{u}^* + \Delta\mathbf{u}, \lambda^* + \Delta\lambda) \quad \text{Eq. 3-50}$$

$$\mathbf{K}_{ee} \begin{Bmatrix} \Delta\mathbf{u} \\ \Delta\lambda \end{Bmatrix} = \begin{Bmatrix} \mathbf{e}_u \\ \mathbf{e}_\lambda \end{Bmatrix} \quad \text{Eq. 3-51}$$

where \mathbf{K}_{ee} is the tangential stiffness matrix and the right hand side is the error vector. The first and second variations of Eq. 3-49 are

$$\delta\Pi = \delta\mathbf{u}^T \mathbf{H}^T (p(\Phi - \mathbf{q}) + k\lambda) + \delta\lambda^T k(\Phi - \mathbf{q}) \quad \text{Eq. 3-52}$$

$$d(\delta\Pi) = \delta\mathbf{u}^T \mathbf{H}^T (p\mathbf{H}d\mathbf{u} + kd\lambda) + \delta\lambda^T k\mathbf{H}d\mathbf{u} + \delta\mathbf{u}^T \frac{\partial}{\partial\mathbf{u}} (\mathbf{H}^T (p(\Phi - \mathbf{q}) + k\lambda)) d\mathbf{u} \quad \text{Eq. 3-53}$$

From Eq. 3-53, we obtain the corresponding element tangential stiffness matrix as:

$$\mathbf{K}_{ee} = \mathbf{K}_{ee}^e + \mathbf{K}_{ee}^d = \begin{bmatrix} \mathbf{H}^T p\mathbf{H} & k\mathbf{H}^T \\ k\mathbf{H} & 0 \end{bmatrix} + \begin{bmatrix} \frac{\partial}{\partial\mathbf{u}} (\mathbf{H}^T (p(\Phi - \mathbf{q}) + k\lambda)) & 0 \\ 0 & 0 \end{bmatrix} \quad \text{Eq. 3-54}$$

In the above equation, the first term is the element stiffness and the second term is the differential stiffness for the rigid element. Next, we need to compute the error vector for the rigid element in a nonlinear iteration. For the approximate solution $(\mathbf{u}^*, \lambda^*)$, the error vector can be obtained from [Eq. 3-52](#) as

$$\begin{Bmatrix} \mathbf{e}_u \\ \mathbf{e}_\lambda \end{Bmatrix} = \begin{Bmatrix} -\mathbf{H}^T(\mathbf{u}^*)(k\lambda^* + p(\Phi(\mathbf{u}^*) - \mathbf{q})) \\ k(\mathbf{q} - \Phi(\mathbf{u}^*)) \end{Bmatrix} \quad \text{Eq. 3-55}$$

With the tangential stiffness matrix and the error vector defined, [Eq. 3-50](#) and [Eq. 3-51](#) are used to obtain a better approximation. This process continues until the solution converges.

The Lagrange Elimination Method

The stiffness matrix for the Lagrange multiplier method given by [Eq. 3-54](#) is not positive definite. For most problems, this method is appropriate. However, for certain type of problems, this method may create numerical difficulties. Therefore, the Lagrange multiplier method with elimination, called the Lagrange elimination method, is implemented as a backup method for the Lagrange multiplier method. For reasons given in “[Types of Rigid Element](#)” on page 174, the Lagrange elimination method is implemented only for the linear static analysis, the normal modes analysis, and the buckling analysis. However, for completeness, the theory given below is still based on the nonlinear analysis.

The Lagrange elimination method is summarized as follows:

- Compute the rigid element tangential stiffness matrix and error vector by using the Lagrange multiplier method as given by [Eq. 3-54](#) and [Eq. 3-55](#) with the penalty function $p=0$.
- Assemble the element stiffness and error vector into global stiffness and error vector, which will create a system equation similar to [Eq. 3-51](#).
- Eliminate the Lagrange multipliers and the dependent degrees-of-freedom from the tangential stiffness matrix and the error vector.
- Construct and solve the Newton-Raphson equation in terms of the independent degrees-of-freedom.

The method of elimination is derived as follows. Let g denote the total displacement degrees of freedom for the structural model, m the dependent degrees-of-freedom, and n the independent degrees-of-freedom. Then the displacement gradient defined by [Eq. 3-48](#) can be partitioned into m -set and n -set as

$$\mathbf{H}_{mg} \Delta \mathbf{u}_g = \begin{bmatrix} \mathbf{H}_{mm} & \mathbf{H}_{mn} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{u}_m \\ \Delta \mathbf{u}_n \end{Bmatrix} \quad \text{Eq. 3-56}$$

And the constraint equation for the dependent degrees-of-freedom can be computed as

$$\begin{aligned} \Delta \mathbf{u}_m &= \mathbf{G}_{mn} \Delta \mathbf{u}_n \\ \mathbf{G}_{mm} &= -\mathbf{H}_{mm}^{-1} \mathbf{H}_{mn} \end{aligned} \quad \text{Eq. 3-57}$$

Now, the global tangential stiffness matrix in g-set, similar to [Eq. 3-51](#), [Eq. 3-54](#), and [Eq. 3-55](#) can be expanded into m-set and n-set as

$$\begin{bmatrix} \bar{\mathbf{K}}_{nn} & \mathbf{K}_{nm} & \mathbf{G}_{mn}^T \\ \mathbf{K}_{nm}^T & \mathbf{K}_{mm} & -\mathbf{I} \\ \mathbf{G}_{mn} & -\mathbf{I} & 0 \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{u}_n \\ \Delta \mathbf{u}_m \\ \Delta \lambda_m \end{Bmatrix} = \begin{Bmatrix} \mathbf{p}_n - \mathbf{F}_n(\mathbf{u}_g^*) - \mathbf{G}_{mn}^T(\mathbf{u}_g^*) \lambda_m^* \\ \mathbf{p}_m - \mathbf{F}_m(\mathbf{u}_g^*) + \lambda_m^* \\ -\mathbf{H}_{mm}^{-1}(\mathbf{q}_m - \Phi(\mathbf{u}_g^*)) \end{Bmatrix} = \begin{Bmatrix} \bar{\mathbf{e}}_n \\ \mathbf{e}_m \\ \bar{\mathbf{q}}_m \end{Bmatrix} \quad \text{Eq. 3-58}$$

The right hand side of above equation is the error vector. Elimination of m-set displacements and the Lagrange multipliers from above equation yields the tangent stiffness matrix in terms of independent degrees-of-freedom, the n-set, as:

$$\mathbf{K}_{nn} \Delta \mathbf{u}_n = \mathbf{e}_n \quad \text{Eq. 3-59}$$

$$\mathbf{K}_{nn} = \bar{\mathbf{K}}_{nn} + \mathbf{K}_{nm} \mathbf{G}_{mn} + \mathbf{G}_{mn}^T \mathbf{K}_{nm}^T + \mathbf{G}_{mn}^T \mathbf{K}_{mm} \mathbf{G}_{mn} \quad \text{Eq. 3-60}$$

$$\mathbf{e}_n = \bar{\mathbf{e}}_n + \mathbf{G}_{mn}^T \mathbf{e}_m + (\mathbf{K}_{nm} + \mathbf{G}_{mn}^T \mathbf{K}_{mm}) \bar{\mathbf{q}}_m \quad \text{Eq. 3-61}$$

$$\Delta \mathbf{u}_m = \mathbf{G}_{mn} \Delta \mathbf{u}_n - \bar{\mathbf{q}}_m \quad \text{Eq. 3-62}$$

$$\Delta \lambda_m = \mathbf{K}_{nm}^T \Delta \mathbf{u}_n + \mathbf{K}_{mm} \Delta \mathbf{u}_m - \mathbf{e}_m \quad \text{Eq. 3-63}$$

[Eq. 3-59](#), [Eq. 3-60](#), and [Eq. 3-61](#) can be used in the Newton-Raphson iterations, and [Eq. 3-62](#) and [Eq. 3-63](#) can be used to recover the displacements in m-set and the Lagrange multipliers. In general, the error vector given by [Eq. 3-59](#) is dependent on the tangential stiffness matrix. In order to compute the correct error vector, the stiffness matrix must be updated at each nonlinear iteration. This means only the full Newton-Raphson method can be used to obtain the solution for the Lagrange elimination method.

User Interfaces

Case Control Command RIGID

Control Case command RIGID selects the type of rigid element. It has the following format:

$$\text{RIGID} = \left\{ \begin{array}{l} \text{LINEAR} \\ \text{LAGR} \\ \text{LGELIM} \end{array} \right\}$$

LINEAR will select the linear rigid elements, LAGR will select the Lagrange rigid element with the Lagrange multiplier method, and LGELIM will select the Lagrange element with the Lagrange elimination method.

If RIGID command does not exist in the user Case Control file, the linear rigid element will be used for all solution sequences except SOL 400. For SOL 400, the default rigid element type is the Lagrange rigid element with the Lagrange multiplier method.

Parameters LANGLE, LMFACT, PENFN, and ORIGID

The parameter LANGLE selects the method to represent the larger rotations, which has the following values:

LANGLE,1 -- use the Gimbal angle method.

LANGLE,2 -- use the left rotation method.

LANGLE,3 -- use the right rotation method.

See sub-section, “[Composite Rotations -- Left Rotation or Right Rotation](#)” on page 178 for a description of the rotation methods. LANGLE=1 or 2 is the existing method. LANGLE=3 is the new method implemented for the Lagrange rigid elements and is the only method available for the Lagrange rigid elements. Therefore, if there are Lagrange rigid elements in a structural model, MSC.Nastran will automatically use LANGLE=3, and ignore user input on the LANGLE parameter.

The parameters LMFACT and PENFN are the scale factor k and penalty function p (see [Eq. 3-49](#)) for the Lagrange rigid element with the Lagrange multiplier method. The purpose of LMFACT and PENFN is to make the values of the stiffness matrix for the Lagrange rigid elements about the same relative magnitude as those of the other elements in a structural model. Too small a value produces inaccurate results and too large a value poses numerical difficulties. Usually, the same value is assign to both LMFACT and PENFN. Under special requirement, user may assign different values to LMFACT and PENFN. For example, if LMFACT \neq 0.0 and PENFN = 0.0, then the

solution method becomes the pure Lagrange multiplier method instead of the augmented Lagrange multiplier method. However, user must exercise caution when assigning different values to LMFACT and PENFN. The default value is 1.0e+5 for all solution sequences except SOL 400. For the nonlinear solution sequence SOL 400, MSC.Nastran will compute the appropriate default value in nonlinear iterations.

The parameter ORIGID is the offset for the identification number of the Lagrange multiplier. MSC.Nastran internally generates the Lagrange multiplier degrees-of-freedom for the Lagrange elements. The identification number for the Lagrange multiplier is ORIGID+MGID, where MGID is the grid point identification number of the dependent grid point of the rigid element. The default value for ORIGID is 101000000 if there are no other internally generated grid points. This value is modifiable using system cell 178

Bulk Data Entries

Existing rigid elements -- RBAR, RBE1, RBE2, RBE3, RROD, and RTRPLT -- can be used as either the linear rigid element or the Lagrange element, which is selected by the Case Control command, RIGID. In addition, there are three new Bulk Data entries, RBAR1, RTRPLT1, and RJOINT, which are discussed in next two sub-sections.

For each Lagrange rigid element, a number of Lagrange multiplier degrees-of-freedom are created internally by MSC.Nastran. The number of the Lagrange multiplier degrees-of-freedom is equal to the dependent degrees-of-freedom defined by the rigid element. The identification numbers of Lagrange multiplier are determined by the parameter ORIGID.

All of the rigid elements described above can either be the linear rigid element or the Lagrange rigid element. However, the input rules are not the same for these two types of rigid element. The following paragraphs discuss the differences.

Except for the RBE3 and RROD, the major difference in the input format between the linear rigid element and the Lagrange rigid element is the selection of the independent degrees-of-freedom. This can be illustrated by the Bulk Data entry for RBAR:

Table 3-12

RBAR	EID	GA	GB	CNA	CNB	CMA	CMB	ALPHA	
------	-----	----	----	-----	-----	-----	-----	-------	--

The independent degrees-of-freedom are selected by CNA and CNB. For the linear rigid element, the independent degrees-of-freedom can be assign to both CNA and CNB: for example CNA=1236, CNB=34, as long as the total number equals to six and they can jointly represent any general rigid body motion. However, for the Lagrange

rigid element, all six independent degrees-of-freedom must be assigned to a single grid point, i.e., “CNA=123456, CNB=blank”, or “CNA=blank, CNB=123456”. The same rule applies to the RBE1, RBE2, and RTRPLT elements.

For the RBE3, the REFC degrees-of-freedom can be any combination of the integer 1 through 6 for the linear rigid element. For the Lagrange rigid element, RFEC must be 123, 456, or 123456.

For the RROD element, the user must select one dependent degree-of-freedom by inputting either CMA or CMB for the linear rigid element. However, for Lagrange rigid element, user can leave both fields blank and let MSC.Nastran to select the best component as the dependent degree-of-freedom. In fact, this is the recommended method.

For all rigid elements, there is a new field ALPHA. This is the thermal coefficient of expansion. For the Lagrange rigid elements, if ALPHA is given and the thermal loads are requested by the Case Control command TEMPERATURE(INITIAL) and TEMPERATURE(LOAD), the thermal load effect will be computed for the rigid elements. The temperature loads are taken as the average temperature given by the independent grid point and the dependent grid point. For example, the temperature load for the BAR element is taken as the average temperature of grid points GA and GB. For the linear rigid element, no temperature effect is computed and the ALPHA field is ignored.

RBAR1 and RTRPLT1

RBAR1 and RTRPLT1 are two new Bulk Data entries and are alternative input formats for RBAR and RTRPLT. The Bulk Data entry for RBAR1 is shown below:

Table 3-13

RBAR1	EID	GA	GB	CMB	ALPHA				
-------	-----	----	----	-----	-------	--	--	--	--

GA is the independent grid point and all six degrees-of-freedom are independent. The dependent degrees-of-freedom are selected by CMB of the dependent grid point GB. This is a much simpler input format than RBAR and is preferred input format for the Lagrange rigid element.

The format for RTRPLT1 is shown below:

Table 3-14

RTRPLT1	EID	GA	GB	GC	CMB	CMC	ALPHA		
---------	-----	----	----	----	-----	-----	-------	--	--

RTRPLT1 defines one independent grid point GA and two dependent grid points GB and GC with dependent degrees-of-freedom CMB and CMC, respectively. Between RTRPLT1 and RTRPLT, RTRPLT1 is the preferred input format for the Lagrange rigid element.

RJOINT

RJOINT is a new rigid element. The format for RJOINT is as follows:

Table 3-15

RJOINT	EID	GA	GB	CB					
--------	-----	----	----	----	--	--	--	--	--

GA is the independent grid point and all six degrees-of-freedom are independent. GB is the dependent grid point. The length between points GA and GB must be zero. Since its length is zero, the thermal load effect is not applicable.

If CB=123456 or blank, then the grid point GB is constrained to move with the grid point GA and two the grid points move as a single point. If any degree-of-freedom is released on CB, then RJOINT becomes a mechanical joint. A mechanical joint is a mechanical system that has two bodies jointed at a point. The two bodies can rotate relatively about one, two or three axis of a local coordinate system at the point. For example:

- Hinge - A hinge is a mechanical joint that rotates freely about one axis about the local coordinate system. It can be simulated by RJOINT with one rotational degree of freedom released, i.e., CB = 12356, 12346, or 12345.
- Universal joint - A universal joint is a mechanical joint that rotates freely in two axes. It can be simulated by RJOINT with two rotational degrees-of-freedom released, i.e., CB = 1234, 1235, or 1236.
- Spherical joint - A spherical joint is a mechanical joint that rotates freely about all three axes. It can be simulated by a zero length RJOINT with all rotational degrees of freedom released, i.e. CB = 123.
- Prismatic joint - A prismatic joint is a mechanical system with two blocks that are constrained to have the same rotations, but translate relative with each other along a local axis. It can be simulated by RJOINT with one translational degree-of-freedom released, i.e., CB = 23456, 13456, 12456.
- Cylindrical joint - A cylindrical joint is a mechanical system that allows two grid points have relative translation along a moving axis and, at the same, have relative rotation about the same axis. It can be simulated by RJOINT with one translational degree of freedom and one rotational degree of freedom released, i.e. CB = 2356, 1346, 1245.

Guidelines for the RJOINT element:

- The theory for the RJOINT is formulated such that a consistent mechanical joint is created even if user requests different global coordinate systems at grid points GA and GB. If different global coordinate systems are used, the degrees-of-freedom to be released is determined by the coordinate system at GB.
- For linear or nonlinear static analysis, the joints must be constrained by other elements in the structural model. Otherwise a singular stiffness matrix may be produced, leading to failure in decomposition.

Limitations on SOL 400

As described in sub-section, “**Composite Rotations -- Left Rotation or Right Rotation**” on page 178, the Lagrange rigid elements are formulated in terms of the right rotation method and the existing elements, CQUAD4, CTRIA3, and CBEAM, are formulated in the left rotation method. In order to make these existing elements compatible with the Lagrange rigid elements, the existing elements must be modified to formulate in the right rotation method. This work is under development for the initial release of MSC.Nastran 2004. Therefore, for the initial release, SOL 400 has the following limitations:

- If Lagrange rigid elements are used in a model that contains CQUAD4, CTRIA3, or CBEAM elements, then the problem will converge only for truly 2D motion. If there are follower force effects, the out-of-plane motion may need to be constrained in this case. Full 3D motion is supported with the Lagrange rigid elements if the model does not contain the above elements.

Examples

In this section, four simple examples are included to illustrate the usage of the Lagrange rigid elements. These four examples are

- Linear static analysis with thermal loads
- Nonlinear geometric analysis with thermal loads
- Buckling analysis
- Normal modes analysis with differential stiffness

Example 1: Linear Static Analysis with Thermal Loads (nlrgd11a.dat)

To illustrate the effect of thermal loads on the Lagrange rigid elements in a static analysis, a simple beam is modeled by a RBAR1 element, which has a length of 1.0. At left end of the beam, a very soft rotational spring is connected to the normal rotation and other five degrees-of-freedom are fixed. At right end, all six degrees-of-freedom are free and a vertical force of 100.0 is acting on it. This beam is subjected to a thermal load, which will double its length to 2.0. This problem is solved by the Lagrange multiplier method (RIGID=LAGR) with the linear static analysis (SOL 101).

The input file for this problem is shown below.

```
ID RIGID,NLRGD10A
SOL 101
CEND
TITLE = LINEAR STATIC ANALYSIS - THERMAL LOAD ON RBAR
SUBTIT= LAGRANGE MULTIPLIER METHOD
RIGID = LAGR
TEMP(INIT) = 10
SUBCASE 1
  TEMP(LOAD) = 20
  LOAD = 100
  SPC = 10
  DISP = ALL
  SPCF = ALL
  FORC = ALL
BEGIN BULK
FORCE,100,2,0,500.0,0.0,1.0,0.0
CELAS2,101,1000.0,1,6
GRID,1,,1.0,0.0,0.0
GRID,2,,2.0,0.0,0.0
RBAR1,3,1,2,123456,2.0-2
TEMP,10,1,0.0
TEMP,10,2,0.0
TEMP,20,1,50.0
TEMP,20,2,50.0
SPC1,10,12345,1
ENDDATA
```

Results for nlrgd11a.dat:

Description	Value
Vertical displacement	0.5
Horizontal displacement	1.0
Angle of rotation	28.6°
Initial beam length	1.0
Final beam length	2.06

These results show that horizontal thermal expansion is correctly computed as 1.0. However, because of the soft rotational spring on the left end of beam, this problem is a large displacement problem; therefore, the vertical and horizontal displacements, angle of rotation, and the final beam length are incorrect.

Example 2: Geometric Nonlinear Analysis with Thermal Loads (nlrgd11b.dat)

This problem is to illustrate the use of the Lagrange rigid elements in geometric nonlinear analysis. The model and loading for this problem is identical to nlrgd11a, except that the geometric nonlinear analysis (SOL 400 and LGDISP,1) is used. Note that the RIGID Case Control command in the is not needed for SOL 400 as the default for SOL 400 is RIGID=LAGR.

The input file for this problem follows.

```
ID RIGID,NLRGD11B
SOL 400
CEND
TITLE = GEOMETRIC NONLINEAR ANALYSIS - THERMAL LOADS ON RBAR
SUBTIT= LAGRANGE MULTIPLIER METHOD
TEMP(INIT) = 10
NLPARM = 10
SUBCASE 1
  TEMP(LOAD) = 20
  LOAD = 100
  SPC = 10
  DISP = ALL
  SPCF = ALL
  FORC = ALL
BEGIN BULK
NLPARM,10,10
PARAM,LGDISP,+1
.
.
.
ENDDATA
```

Results for nlrgd11b.dat:

Description	Value
Vertical displacement	1.344
Horizontal displacement	0.481
Angle of rotation	42.22°
Initial beam length	1.0
Final beam length	2.0

Now the final beam length is 2.0, which is expected, and all above results are correct when large displacement theory is used.

Example 3: Buckling Analysis (nlrgd10d.dat)

To illustrate the use of the Lagrange rigid elements in a buckling analysis, two simple columns are modeled by RBAR elements and subject to compressive loads of 100.0 at top of the column. At bottom of the columns, they are connected to a single grid point and are constrained by very soft rotational spring in both horizontal directions. Also, at bottom, the three translations and axial rotation are fixed. The buckling analysis (SOL 105) and the Lagrange elimination method (RIGID=LGELIM) are used to obtain the solution.

The input file for this problem is shown below.

```
ID RIGID,NLRGD10D
SOL 105
CEND
TITLE = BUCKLING ANALYSIS - RBAR
SUBTI = LAGRANGE ELIMINATION METHOD
DISP = ALL
SPC = 10
RIGID = LGELIM
SUBCASE 1
$ STATIC PRELOAD CASE
  LOAD = 100
SUBCASE 2
$ BUCKLING CASE
  METHOD = 10
BEGIN BULK
EIGRL,10,, ,10
FORCE,100,4 ,0,-100.0,1.0,0.0,0.0
FORCE,100,41,0,-100.0,1.0,0.0,0.0
CELAS2,101,100.0,3,5
CELAS2,102,120.0,3,6
GRID, 3 , ,2.0,0.0,0.0
GRID, 4 , ,4.0,0.0,0.0
GRID, 41, ,4.0,0.0,0.0
RBAR, 4,3,4
RBAR, 3,3,41,123456, , ,123456
SPC1,10,1234,3
ENDDATA
```

The buckling load factors calculated are 0.25 and 0.30. For SOL 105, currently only the Lagrange elimination method (RIGID=LGELIM) is supported (see error 1-11842201).

Example 4: Normal Modes Analysis with Differential Stiffness (nlrgd14c.dat)

This problem illustrates the effect of differential stiffness matrix with the Lagrange rigid elements for normal modes analysis. The model of this problem is similar to that of nlrgd10a, except that concentrated masses are added to the top of the columns and tensile forces of 100.0 units are applied. The normal modes analysis (SOL 103) using the Lagrange multiplier method (RIGID=LAGR) are used to obtain the solution.

Two problems are investigated--one without and one with the effects of the differential stiffness matrix included.

The input file for the problem including the differential stiffness effect is shown below. For the problem without the differential stiffness, just remove subcase 1.

```
ID RIGID,NLRGD14C
SOL 103
CEND
TITLE = NORMAL MODES ANALYSIS WITH DIFFERENTIAL STIFFNESS - RBAR
SUBTI = LAGRANGE MULTIPLIER METHOD
DISP = ALL
SPC = 10
RIGID = LAGR
SUBCASE 1
$ STATIC PRELOAD CASE
  LOAD = 100
SUBCASE 2
$ NORMAL MODES CASE
  STATSUB = 1
  METHOD = 10
BEGIN BULK
EIGRL,10,,10
CONM2,201,4,,10.
CONM2,202,41,,10.
FORCE,100,4,0,+100.0,1.0,0.0,0.0
FORCE,100,41,0,+100.0,1.0,0.0,0.0
CELAS2,101,100.0,3,5
CELAS2,102,120.0,3,6
GRID,3,,2.0,0.0,0.0
GRID,4,,4.0,0.0,0.0
GRID,41,,4.0,0.0,0.0
RBAR,4,3,4
RBAR,3,3,41,123456,,123456
SPC1,10,1234,3
ENDDATA
```

Results for nlrqd11b.dat:

Description	Value
Eigenvalues without preload	1.25, 1.50
Eigenvalues with preload	6.25, 6.50

From the above results, we can see clearly that the differential stiffness can affect the results substantially.

3.6 Scalar Elements (CELASi, CMASSi, CDAMPi)

Scalar elements are connected between pairs of degrees-of-freedom (at either scalar or geometric grid points) or between one degree-of-freedom and ground. Scalar elements are available as springs, masses, and viscous dampers. Scalar spring elements are useful for representing elastic properties that cannot be conveniently modeled with the usual structural elements (elements whose stiffnesses are derived from geometric properties). Scalar masses are useful for the selective representation of inertia properties, such as occurs when a concentrated mass is effectively isolated for motion in one direction only. The scalar damper is used to provide viscous damping between two selected degrees-of-freedom or between one degree-of-freedom and ground. It is possible, using only scalar elements and constraints, to construct a model for the linear behavior of any structure. However, use of scalar elements with offsets will cause incorrect results in buckling analysis and differential stiffness because the large displacement effects are not calculated. Offsets will also cause internal constraints in linear analysis, i.e., hidden constraints to ground. Therefore, it is expected that these elements will be used only when the usual structural elements are not satisfactory. Scalar elements are useful for modeling part of a structure with its vibration modes or when trying to consider electrical or heat transfer properties as part of an overall structural analysis. The reader is referred to Sections 5.5 and 5.6 of *The NASTRAN Theoretical Manual* for further discussions on the use of scalar elements.

The most general definition of a scalar spring is given with a CELAS1 entry. The associated properties are given on the PELAS entry. The properties include the magnitude of the elastic spring, a damping coefficient, and a stress coefficient to be used in stress recovery. The CELAS2 defines a scalar spring without reference to a property entry. The CELAS3 entry defines a scalar spring that is connected only to scalar points and the properties are given on a PELAS entry. The CELAS4 entry defines a scalar spring that is connected only to scalar points and without reference to a property entry. No damping coefficient or stress coefficient is available with the CELAS4 entry. Element force F is calculated from the equation.

Static:

$$F = k(u_1 - u_2) \quad \text{Eq. 3-64}$$

Frequency:

$$F = (1 + i(g + g_e))k\bar{U} ; \bar{U} = (u_1 - u_2)_{\text{real}} + i(u_1 - u_2)_{\text{img}} \quad \text{Eq. 3-65}$$

Transient:

$$F = (u_1 - u_2)k + \left(\frac{g}{W3} + \frac{g_e}{W4} \right) k (\dot{u}_1 - \dot{u}_2) \quad \text{Eq. 3-66}$$

where k is the stiffness coefficient for the scalar element and u_1 is the displacement of the first degree-of-freedom listed on its connection entry. Element stresses are calculated from the equation

$$\sigma = S \cdot F \quad \text{Eq. 3-67}$$

where S is the stress coefficient on the connection or property entry and F is as defined above.

Scalar elements may be connected to ground without the use of constraint entries. Grounded connections are indicated on the connection entry by leaving the appropriate scalar identification number blank. Since the values for scalar elements are not functions of material properties, no references to such entries are needed.

The CMASS1, CMASS2, CMASS3, and CMASS4 entries define scalar masses in a manner similar to the scalar spring definitions. The associated PMASS entry contains only the magnitude of the scalar mass.

The CDAMP1, CDAMP2, CDAMP3, CDAMP4, and CDAMP5 entries define scalar dampers in a manner similar to the scalar spring definitions. The associated PDAMP entry contains only a value for the scalar damper. The mode displacement method (PARAM,DDRMM,-1) must be selected for element force output.

For the CDAMP, the force equations are:

Frequency:

$$F = i\omega B(u_1 - u_2)$$

Transient:

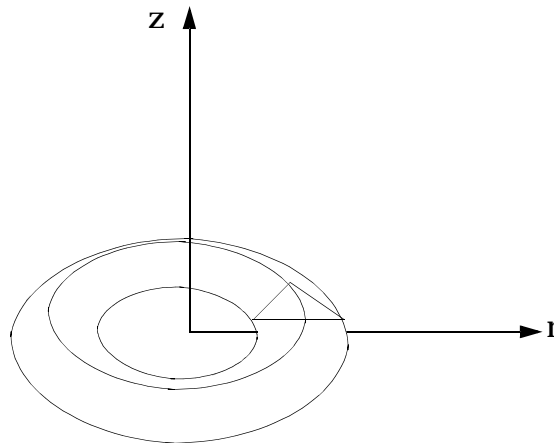
$$F = B(\dot{u}_1 - \dot{u}_2)$$

3.7 Axisymmetric Solid Elements (CTRIAX6, CTRIAX, CQUADX)

The axisymmetric elements CTRIAX6, CTRIAX, and CQUADX define a solid ring by sweeping a surface defined on a plane through a circular arc. The CTRIAX and CQUADX elements are described in “[Hyperelastic Elements](#)” on page 221. The CTRIAX6 element is described as follows.

The triangular ring element (CTRIAX6) is a linear isoparametric element with an axisymmetric configuration that is restricted to axisymmetric applied loading. It is used for the modeling of axisymmetric, thick-walled structures of arbitrary profile. This element is not designed to be used with any other elements. Otherwise, CTRIAX6 is used in a conventional manner, and except for its own connection entry (and a pressure load entry, PLOADX1) it does not require special bulk data entries.

The coordinate system for the CTRIAX6 element is shown in [Figure 3-44](#). Cylindrical anisotropy is optional. Orientation of the orthotropic axes in the (r, z) plane is specified by the angle θ . Deformation behavior of the element is described in terms of translations in the r and z directions at each of the six grid points. All other degrees-of-freedom should be constrained.



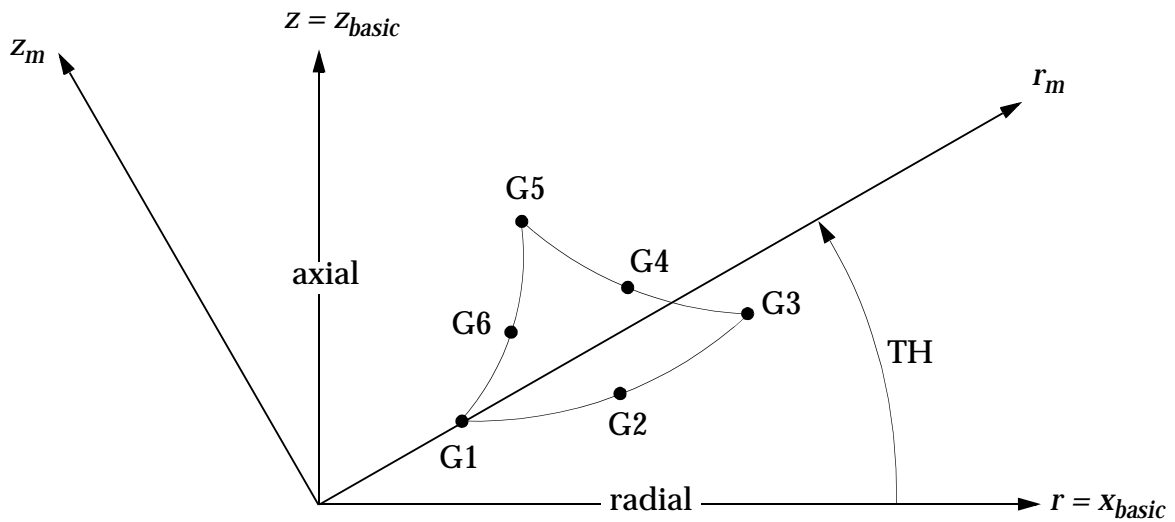


Figure 3-44 CTRIAX6 Element Coordinate System

The isoparametric triangular ring element is defined with a CTRIAX6 entry. No property entry is used for this element. The material property reference is given on the connection entry.

The following stresses, evaluated at the three vertex grid points and at the centroid of the element, are output on request:

- σ_r -- stress in r_m direction of material coordinate system.
- σ_θ -- stress in azimuthal direction.
- σ_z -- stress in z_m direction of material coordinate system.
- τ_{rz} -- shear stress in material coordinate system.
- Maximum principal stress.
- Maximum shear stress.
- von Mises equivalent or octahedral shear stress.

3.8 Crack Tip Elements (CRAC2D, CRAC3D)

MSC.Nastran crack tip elements include both two-dimensional (CRAC2D) and three-dimensional (CRAC3D) types. ADUM8 and ADUM9 Bulk Data entries are needed with these elements. The formats of these entries are as follows:

	1	2	3	4	5	6	7	8	9	10
ADUM8	18	0	5	0	CRAC2D					
ADUM9	64	0	6	0	CRAC3D					

The two geometries of the CRAC2D element are shown in [Figure 3-45](#) and [Figure 3-46](#). For the quadrilateral crack element, corner and crack grid points 1 through 10 are required while midside grid points 11 through 18 are optional. The CRAC3D element geometry is depicted in [Figure 3-47](#). For the brick crack element, grid points 1 through 10 and 19 through 28 are required; whereas grid points 11 through 18 and 29 through 64 are optional.

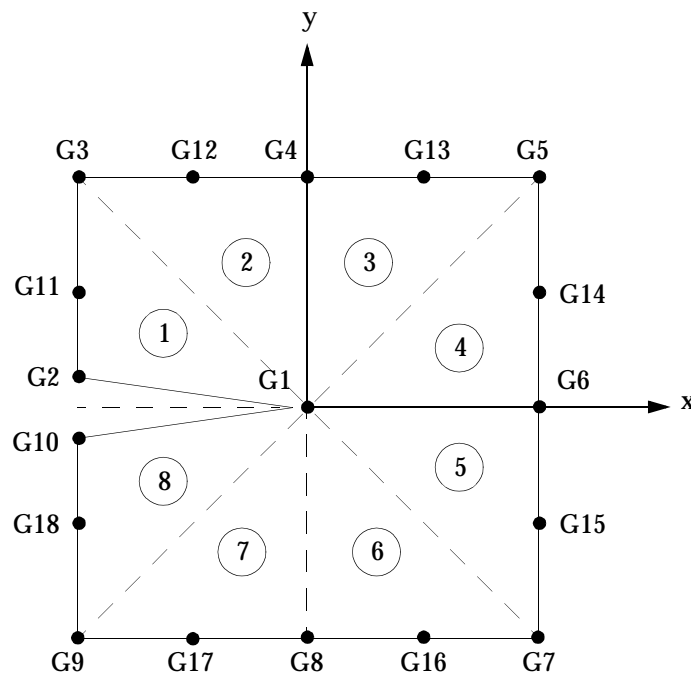


Figure 3-45 Quadrilateral Crack Element

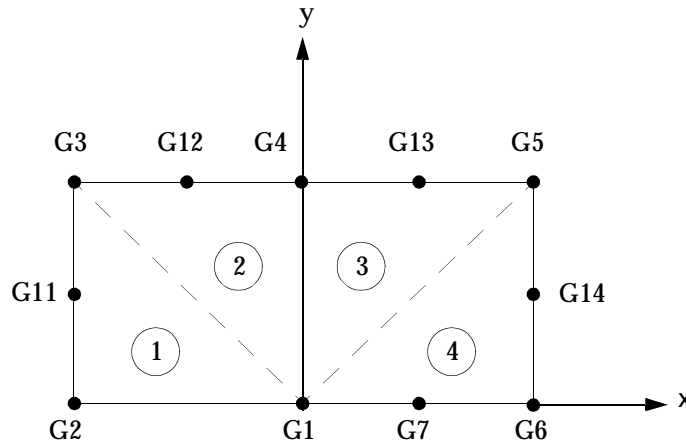


Figure 3-46 Symmetric Half-Crack Option

The CRAC2D and CRAC3D entries are used to define the element geometry, while properties are defined on PRAC2D and PRAC3D entries. Both elements can be modeled with temperature-independent anisotropic materials. The 2-D element may be either plane stress or plane strain. The element will generate either coupled or lumped mass matrices.

Figure 3-45 depicts the quadrilateral and symmetric half options for the CRAC2D element. The element is divided, by the program, into basic triangular elements 1 through 8, whereas the symmetric half-crack option is divided into triangles 1 through 4. Grid points 1 through 7 are required for the symmetric half-crack option, while grid points 11 through 14 are optional. Stresses and the local coordinates of these stresses for the quadrilateral option are computed at the origin of the natural coordinates of triangles 4 and 5. The stresses and coordinates are then averaged and reported. Stress intensity factors, K_I and K_{II} , are computed for triangles 1 and 8, averaged and reported. For the symmetric half-crack option, coordinates and stresses are reported at the origin of the natural coordinates of triangle 4 while the stress intensity factor K_I only is reported for triangle 1.

Interpretation of CRAC2D Element Stress Output (Dummy Element Format)

S1	S2	S3	S4	S5	S6	S7	S8	S9
x	y	σ_x	σ_y	τ_{xy}	K_I	K_{II}	0	0

where x and y are the element coordinates where stresses are reported. K_I and K_{II} are stress intensity factors.

Figure 3-47 depicts the 3-D brick and symmetric half-crack options for the CRAC3D element. The brick element is divided, by the program, into basic wedge elements labeled 1 through 8, whereas the symmetric half-crack option is divided into wedges

1 through 4. Grid points 1 through 7 and 19 through 25 are required for symmetric crack option. Stresses and the local coordinates of these stresses for the brick element option are computed at the origin of the natural coordinates of wedges 4 and 5. The stresses and coordinates are then averaged and reported. Stress intensity factors, K_I and K_{II} , are computed for wedges 1 and 8, averaged, and reported. For the symmetric half-crack option, coordinates and stresses are reported at the origin of the natural coordinates of wedge 4, whereas the stress intensity factor K_I only is reported for wedge 1.

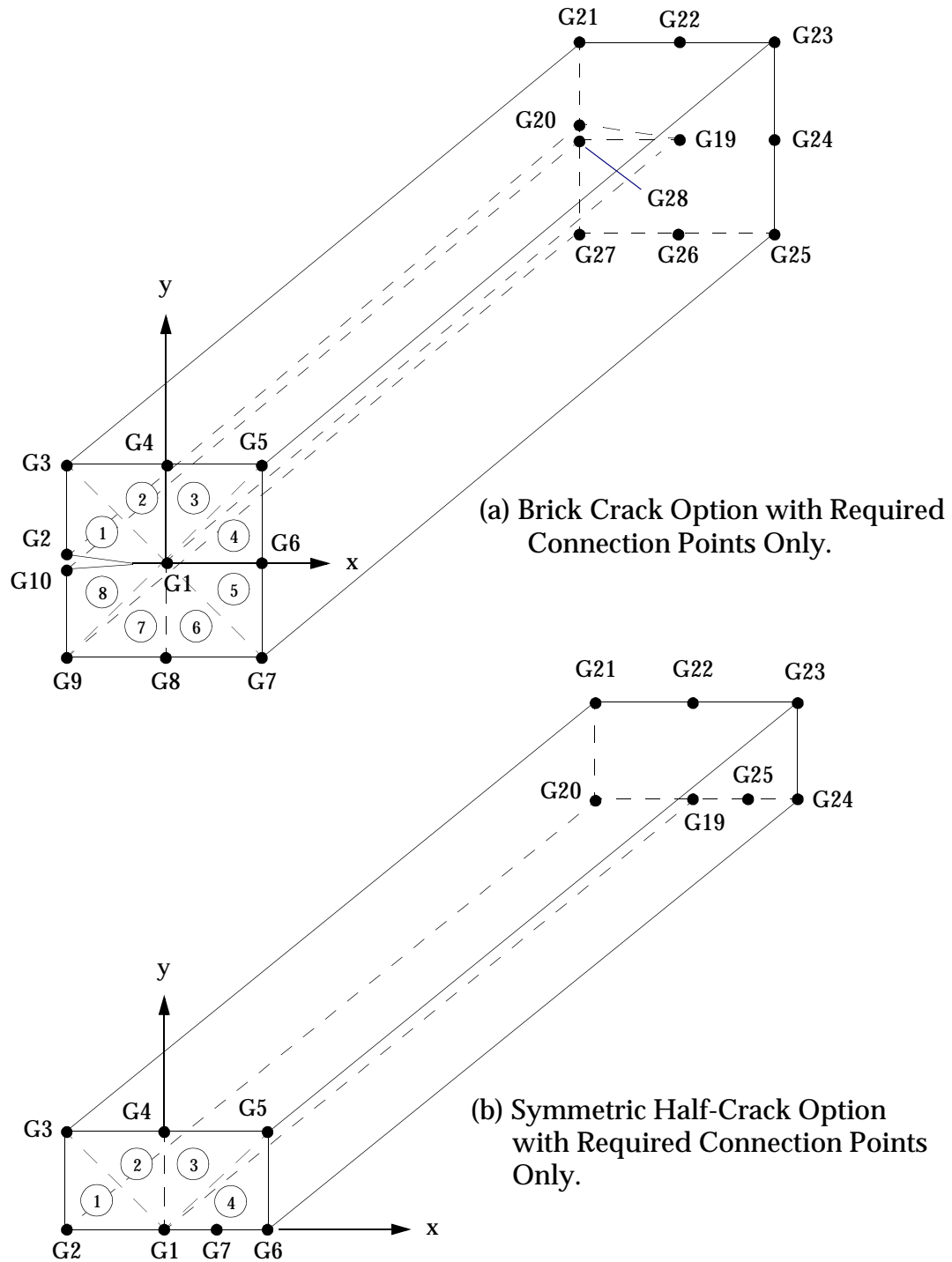


Figure 3-47 The CRAC3D Solid Crack Tip Element

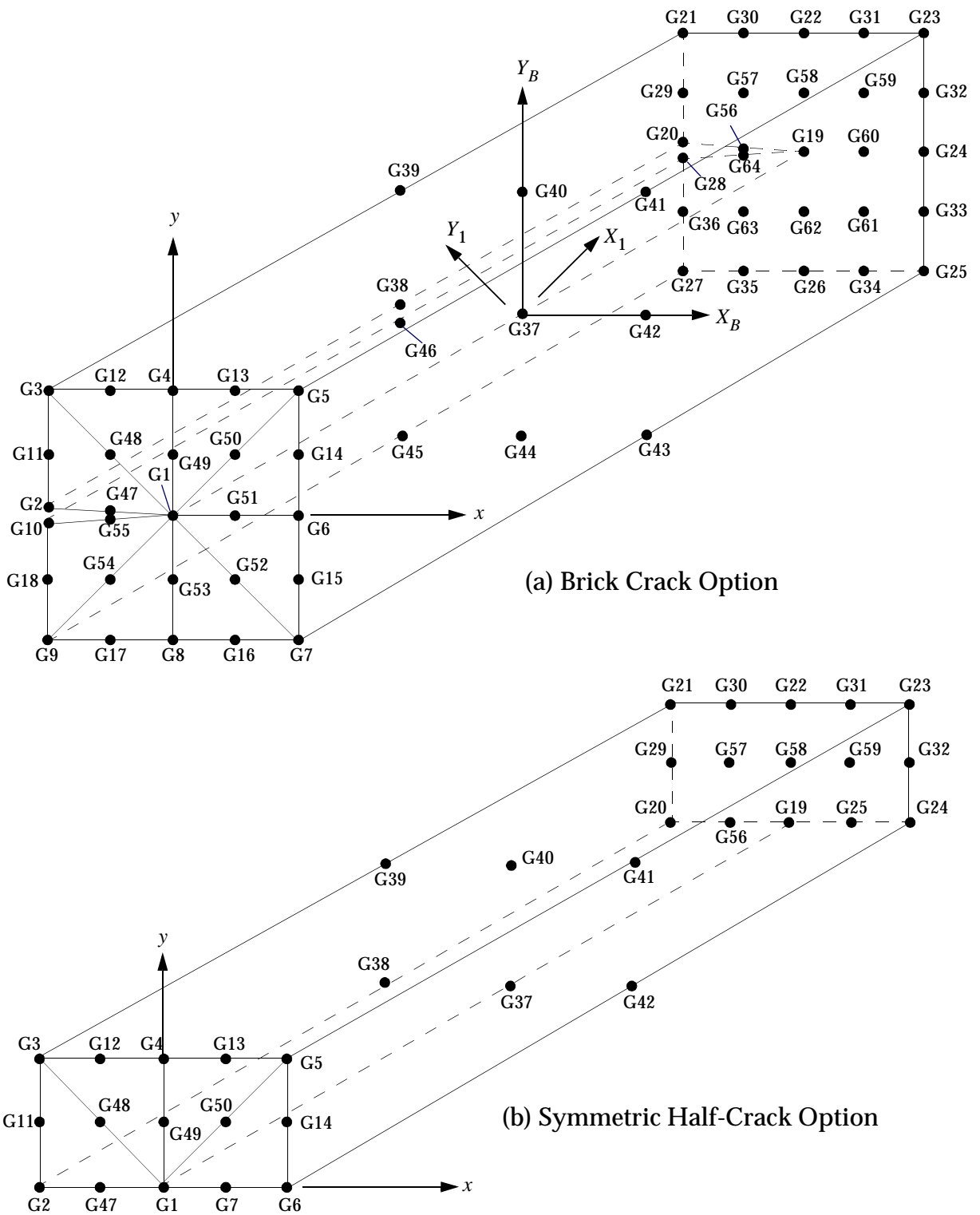


Figure 3-47 The CRAC3D Solid Crack Tip Element (continued)

Interpretation of CRAC3D Element Stress Output (Dummy Element Format)

S1	S2	S3	S4	S5	S6	S7	S8	S9
σ_x	σ_y	σ_z	τ_{xy}	τ_{yz}	τ_{zx}	K_I	K_{II}	K_{III}

Stresses are reported at the average of the origin of the natural coordinate of wedges 4 and 5, whereas for the symmetric crack option they are reported at the origin of the natural coordinate of wedge 4.

The CRAC2D element is based upon a 2-D formulation, but may be used in three-dimensional structures. However, the element should be planar. Any deviation from a planar element is checked, and if significant deviations arise, error messages will be issued.

The CRAC3D element is based upon a 3-D formulation. Both the faces (formed by grid points 1 through 18 and grid points 19 through 36) and the midplane (grid points 37 through 46) should be planar. Any significant deviation will be checked and error messages will be issued.

3.9 Gap and Line Contact Elements (CGAP, BCONP, BLSEG)

Gap Element

The gap and friction elements are specified on a CGAP entry. The element coordinate system and nomenclature are shown in **Figure 3-48**. CID is required, if it is used to define the element coordinate system. Otherwise, the X -axis of the element coordinate system, x_{elem} , is defined by a line connecting GA and GB of the gap element. The orientation of the gap element is determined by vector \vec{v} similar to the definition of the beam element, which is in the direction from grid points GA to GO or defined by (X1, X2, X3).

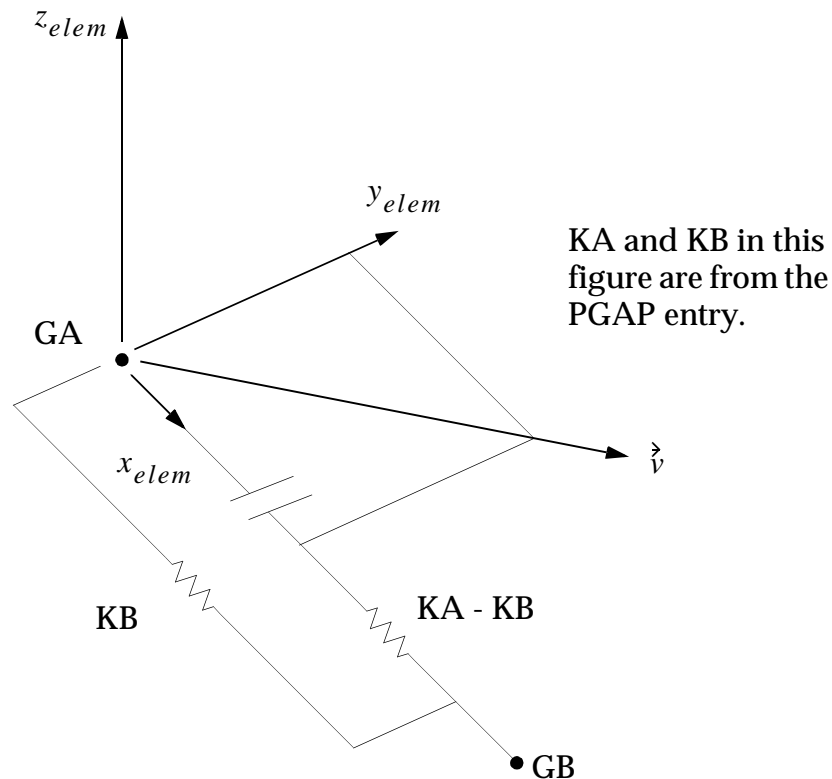


Figure 3-48 Gap Element Coordinate Systems

The properties for the gap elements are defined on the PGAP entry. The initial gap opening is defined by U_o . If the gap is closed ($U_A - U_B \geq U_o$), the axial stiffness (KA) has a very large value (relative to the adjacent structure). When the gap is open, there is a small stiffness KB in the axial direction.

MSC.Nastran includes two types of gap elements: nonadaptive and adaptive. When the nonadaptive GAP element is used, the anisotropic coefficients of friction (μ_1 and μ_2) are specified for the frictional displacements. Also, the anisotropic coefficients of friction are replaced by the coefficients of static and dynamic friction μ_s and μ_k . On the PGAP continuation entry, the allowable penetration limit T_{max} should be specified because there is no default. In general, the recommended allowable penetration T_{max} is about 10% of the element thickness for plates or the equivalent thickness for other elements that are connected by GA and GB. When T_{max} is set to zero, the penalty values will not be adjusted adaptively.

Gap element forces (or stresses) and relative displacements are requested by the STRESS or FORCE Case Control command and computed in the element coordinate system. A positive axial force F_x indicates compression. For the element with friction, the magnitude of the slip displacement is always less than the shear displacement after the slip starts. For the element without friction, the shear displacements and slip displacements have the same value.

3-D Slideline Contact

Definition and Acronyms

The terminology associated with the slideline capability is as follows:

- Two deformable bodies that come into contact are called the master and the slave. The modeling of interactions between the master and slave bodies requires you to define the contact regions in terms of slidelines. A master line is a list of grid points in the topological order on the master body and a slave line is a list of grid points in the topological order on the slave body. A slideline contact region consists of a master line and a slave line as shown in **Figure 3-49**. The grid points on the master line are called the master nodes and those on the slave line are called the slave nodes.

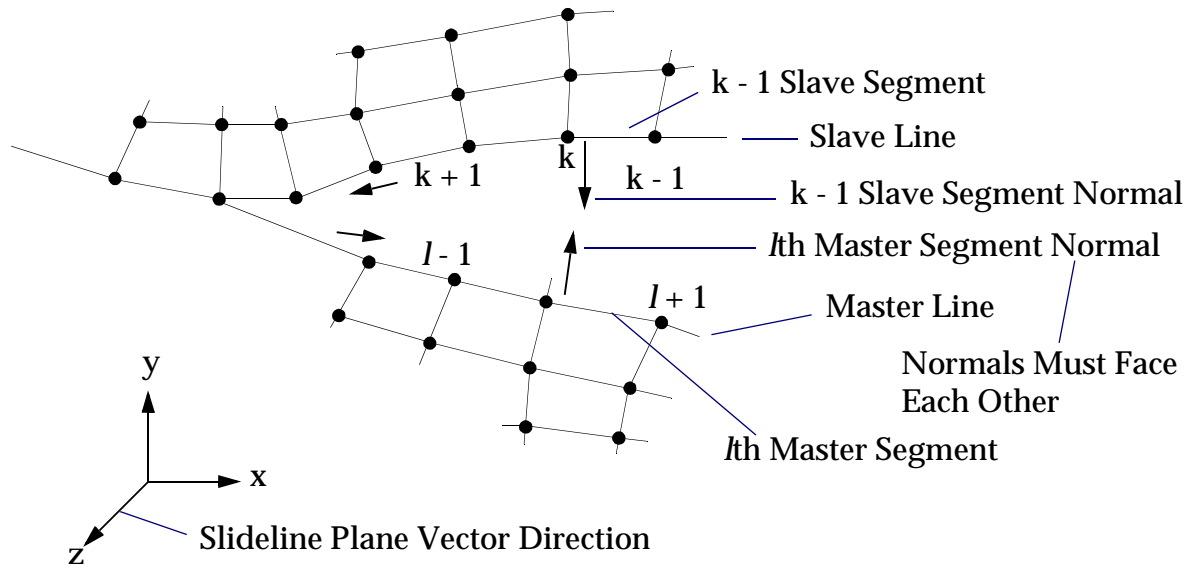


Figure 3-49 Typical Slideline Contact Region

- Arrows show positive direction for ordering nodes. Counterclockwise from master line to slave line.
- X-Y plane is the slideline plane. Unit normal in the Z-direction is the slideline plane vector.
- Slave and master segment normals must face each other; i.e., the normals must be in the opposite directions.
- A line segment joining two consecutive master nodes is called a master segment. The master and the slave nodes in a contact region must lie in a plane called the slideline plane (see [Figure 3-49](#)). The slideline plane is defined by a vector normal to the plane called the slideline plane vector. The master and the slave can have large relative motions within the slideline plane. However, relative motion outside the slideline plane is ignored, and therefore, motion must be small compared to a typical master segment. Thus, slideline contact is ideal for modeling interactions between two bodies that may come in contact in specified planes. MSC.Nastran checks to make sure that in the initial geometry all the nodes of a contact region lie in the slideline plane and issues the appropriate messages if they are not. However, no such check is made during the analysis.

In general, contact is determined only for slave nodes and the master line. This may result in master nodes penetrating the slave line. However, the error involved depends only on the mesh discretization. At the expense of increase in computer runtime, you may wish to determine contact between

master nodes and the slave line in addition to determining contact between slave nodes and the master line. This is called symmetric penetration. Thus for symmetric penetration, no distinction is made between slave and master.

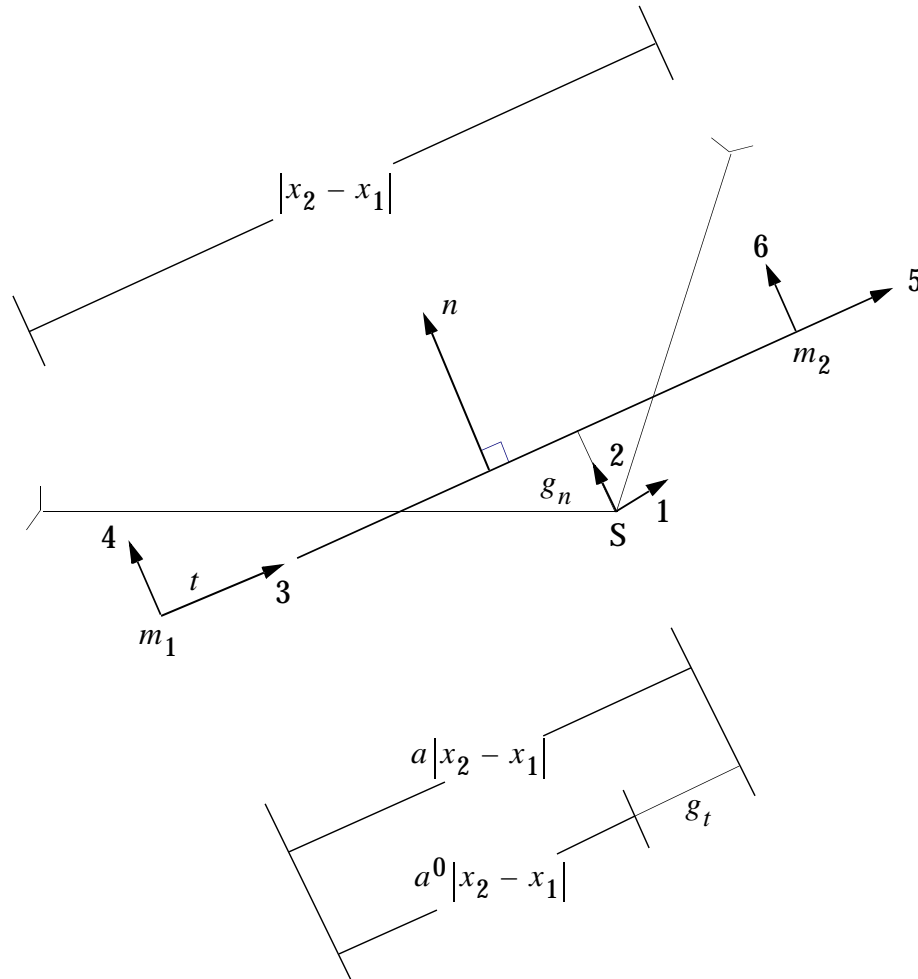
Initial penetration between two bodies is not allowed. The initial gaps between the slave nodes and the master line are calculated based on the coordinates specified for the slave and the master nodes. If the grid coordinates are not specified accurately enough (i.e., MSC.Nastran will determine whether initial penetration exists), then MSC.Nastran will adjust the coordinates automatically. In the case where the calculated initial penetration is less than ten percent of the length of the master segment to which it projects, MSC.Nastran issues a warning message and automatically adjusts the coordinates. However, in the case where the initial penetration is larger than ten percent, the analysis terminates and a fatal message is issued. If the element normal of the master and slave segments are not pointing towards each other, MSC.Nastran will stop and issue a fatal message similar to one issued in the case of initial penetration, even though the initial gaps may be acceptable.

Guidelines

MSC.Nastran creates a local coordinate system for each master segment to determine contact between the slave nodes and the master line. The tangent direction for the master segment is given by the vector from its first node to the second node. It is very important that you specify the master nodes in the topological order. MSC.Nastran then determines the normal direction to the master segment by the cross product between the slideline plane vector and the tangent vector. The normal direction must point from the master line to the slave line because MSC.Nastran determines the contact between the slave nodes and the master line by measuring how close the slave nodes are to the master segment in that normal direction (see [Figure 3-49](#)).

For symmetric penetration, tangent and normal directions are also defined for the slave segment similar to the master segment. The normals for the master segment must point toward the slave and the normals for the slave must point toward the master (see [Figure 3-49](#)). This is generally accomplished by traversing from master line to slave line in a counter-clockwise fashion or clockwise fashion depending on whether the slideline plane vector forms a right-hand or a left-hand coordinate system with the slideline plane. It is your responsibility to make sure that the normals face each other. MSC.Nastran checks indirectly to make sure that the normals face each other by calculating the initial gap between the two bodies. This is not a foolproof check, especially when there is no initial clearance.

Once MSC.Nastran detects a contact between a node and a body, a three node slideline element as shown in **Figure 3-50** is automatically created. The first node of this element is the slave node and the last two nodes are the master nodes of the master segment with which the slave node is in contact. The slave node is forced to slide on the master line after contact and must remain on the master line until a tensile force develops.



- S, m_1 , , m_2 = slave, master node 1 and master node 2, respectively
- a, a_0 = current and previous surface coordinate
- g_n = penetration of slave node into the master segment
- g_t = sliding of the slave node on the master segment
- n = normal direction for the master segment

Figure 3-50 Geometry and Definition of a Typical Three-Node Slideline Element

In MSC.Nastran the contact and traction forces/stresses are always associated with slave nodes. In order to compute the contact and traction stresses, an area is associated with each slave node, based on the contributory length and the width/thickness from the adjacent slave segments. Therefore, you need to specify widths associated with slave lines in a three-dimensional model or thickness associated with slave lines in a two dimensional model. You can also request output results for any number of slideline contact regions. The output for a slave node consists of (a) the slideline contact region identification number, (b) the master segment to which it projects, (c) the parametric surface coordinate to identify the exact projection of slave nodes relative to the master nodes of the master segment, and (d) a slip ratio to indicate whether the slave node is sticking, slipping or sliding.

Text Input

Table 3-16 shows how the slideline input data is given to MSC.Nastran through executive and case control as well as the bulk data entries. For a complete list of these input entries, please refer to the *MSC.Nastran Quick Reference Guide*.

Table 3-16 MSC.Nastran Input for Nonlinear Slideline Contact

Slideline Input	MSC.Nastran Input
Analysis Type and Diagnostics Request	Executive Control Parameters:
Nonlinear Statics	SOL 106
Nonlinear Dynamics	SOL 129
Slideline Diagnostics	DIAG 35 -- Provides extended slideline diagnostics output.
Slideline Output Request	Case Control Parameters: BOUTPUT -- Selects the contact regions for history output.
Contact Parameters	Bulk Data Entries: BCONP -- Defines the parameters for contact between two bodies.
Boundary Line Segments	BLSEG -- Defines the curve, which consists of a number of line segments via grid point numbers that may come in contact with other body.
Boundary Line Segments Width/Thickness	BWIDTH -- Defines width/thickness for line segments in 3-D/2-D slideline contact defined in the corresponding BLSEG Bulk Data entry.

Table 3-16 MSC.Nastran Input for Nonlinear Slideline Contact (continued)

Slideline Input	MSC.Nastran Input
Contact Friction	BFRIC -- Defines frictional properties between two bodies in contact.
Contact Output	BOUTPUT -- Defines the slave nodes at which the output is requested.
User Control and Results Processing; Parameters	ADPCON -- (Default = 1.0); Automatically calculated penalty values can be scaled by specifying a real value for the ADPCON. This is useful for changing the penalty values on restart. This will change the penalty value to be equal to a number calculated by MSC.Nastran times SFAC value specified in BCONP Bulk Data entry times ADPCON . ADPCON is applied to all the contact regions.

The slideline contact formulation is based on the penalty method, which uses an artificial stiffness value. MSC.Nastran will automatically calculate the appropriate numbers based on local stiffness. You have the option to scale the calculated penalty values. The automatically calculated penalty values may sometimes cause convergence difficulties. However, you can overcome the convergence problems by restarting the analysis from any previous converged solution and scaling down the penalty value. Generally, we recommend decreasing the penalty values by one order of magnitude at a time.

3.10 Concentrated Mass Elements (CONM1, CONM2)

The concentrated mass elements are used to define a concentrated mass at a grid point. Two forms of input of concentrated mass are supported, the CONM1 and CONM2 elements.

The CONM1 allows a general 6×6 symmetric mass matrix in a specified coordinate system to be assigned to a geometric grid point. In addition, a less general form, the CONM2 element allows a concentrated mass about its center of gravity to be specified. CONM2 allows for the specification of the offset of the center of gravity of the concentrated mass relative to grid point location, a reference coordinate system, the mass and a 3×3 symmetric matrix of mass moments of inertia measured from its center of gravity. For additional information see the “[Bulk Data Entries](#)” on page 849 of the *MSC.Nastran Quick Reference Guide*.

3.11 General Element Capability (GENEL)

The input required for the general element is the GENEL Bulk Data entry. The description of “**GENEL**” on page 1393 of the contains a complete description of the input options and an example.

The principal application of the GENEL element is to represent the stiffness of a substructure which has an arbitrary number of connected grid point components and/or scalar points. The input data may be obtained either from another computer run or from test data.

The general element is a structural stiffness element connected to any number of degrees-of-freedom, as specified by the user. In defining the form of the externally generated data on the stiffness of the element, two major options are provided to the user.

1. Instead of supplying the stiffness matrix for the element directly, the user provides the deflection influence coefficients for the structure supported in a nonredundant manner. The associated matrix of the restrained rigid body motions may be input or may be generated internally by the program.
2. The stiffness matrix of the element may be input directly. This stiffness matrix may be for the unsupported body, containing all the rigid body modes, or it may be for a subset of the body’s degrees-of-freedom from which some or all of the rigid body motions are deleted. In the latter case, the option is given for automatic inflation of the stiffness matrix to reintroduce the restrained rigid body terms, provided that the original support conditions did not constitute a redundant set of reactions. An important advantage of this option is that, if the original support conditions restrain all rigid body motions, the reduced stiffness matrix need not be specified by the user to high precision in order to preserve the rigid body properties of the element.

The defining equation for the general element when written in the flexibility form is

$$\begin{Bmatrix} u_i \\ f_d \end{Bmatrix} = \begin{bmatrix} Z & | & S \\ -S^T & | & \mathbf{0} \end{bmatrix} \begin{Bmatrix} f_i \\ u_d \end{Bmatrix} \quad \text{Eq. 3-68}$$

where:

$[Z]$ = is the matrix of deflection influence coefficients for coordinates $\{u_i\}$ when coordinates $\{u_d\}$ are rigidly restrained.

$[S]$ = is a rigid body matrix whose terms are the displacements $\{u_i\}$ due to unit motions of the coordinates $\{u_d\}$, when all $f_i = 0$.

$\{f_i\}$ = are the forces applied to the element at the $\{u_i\}$ full coordinates.

$\{f_d\}$ = are the forces applied to the element at the $\{u_d\}$ coordinates. They are assumed to be statically related to the $\{f_i\}$ forces, i.e., they constitute a nonredundant set of reactions for the element.

The defining equation for the general element when written in the stiffness form is

$$\begin{Bmatrix} f_i \\ f_d \end{Bmatrix} = \begin{bmatrix} k & -kS \\ -S^T k & S^T k S \end{bmatrix} \begin{Bmatrix} u_i \\ u_d \end{Bmatrix} \quad \text{Eq. 3-69}$$

where all symbols have the same meaning as in **Eq. 3-68** and $[k] = [Z]^{-1}$, when $[k]$ is nonsingular. Note, however, that it is permissible for $[k]$ to be singular. **Eq. 3-69** derivable from **Eq. 3-68** when $[k]$ is nonsingular.

Input data for the element consists of lists of the u_i and u_d coordinates, which may occur at either geometric or scalar grid points; the values of the elements of the $[Z]$ matrix, or the values of the elements of the $[k]$ matrix; and (optionally) the values of the elements of the $[S]$ matrix.

The user may request that the program internally generate the $[S]$ matrix. If so, the $\{u_i\}$ and coordinates occur only at geometric grid points, and there must be six or less $\{u_d\}$ coordinates that provide a nonredundant set of reactions for the element as a three-dimensional body.

The $[S]$ matrix is internally generated as follows. Let $\{u_b\}$ be a set of six independent motions (three translations and three rotations) along coordinate axes at the origin of the basic coordinate system. Let the relationship between $\{u_d\}$ and $\{u_b\}$.

$$\{u_d\} = [D_d]\{u_b\} \quad \text{Eq. 3-70}$$

The elements of $[D_d]$ are easily calculated from the basic (x,y,z) geometric coordinates of the grid points at which the elements of $\{u_d\}$ occur, and the transformations between basic and global (local) coordinate systems. Let the relationship between $\{u_i\}$ and $\{u_b\}$ be

$$\{u_i\} = [D_i]\{u_b\} \quad \text{Eq. 3-71}$$

where $[D_i]$ is calculated in the same manner as $[D_d]$. Then, if $[D_d]$ is nonsingular,

$$[S] = [D_i][D_d]^{-1} \quad \text{Eq. 3-72}$$

Note that, if the set $\{u_d\}$ is not a sufficient set of reactions, $[D_d]$ is singular and $[S]$ cannot be computed in the manner shown. When $\{u_d\}$ contains fewer than six elements, the matrix $[D_d]$ is not directly invertible but a submatrix $[a]$ of rank r , where r is the number of elements of $\{u_d\}$, can be extracted and inverted.

A method which is available only for the stiffness formulation and not for the flexibility formulation will be described. The flexibility formulation requires that $\{u_d\}$ have six components. The method is as follows. Let $\{u_d\}$ be augmented by 6- r displacement components $\{u_d'\}$ which are restrained to zero value. We may then write

$$\begin{Bmatrix} u_d \\ d_d' \end{Bmatrix} = \begin{bmatrix} D_d \\ D_d' \end{bmatrix} \{u_b\} = [\bar{D}]\{u_b\} \quad \text{Eq. 3-73}$$

The matrix $[D_d]$ is examined and a nonsingular subset $[a]$ with r rows and columns is found. $\{u_b\}$ is then reordered to identify its first r elements with $\{u_d\}$. The remaining elements of $\{u_b\}$ are equated to the elements of $\{u_d\}$. The complete matrix $[\bar{D}]$ then has the form

$$[\bar{D}] = \begin{bmatrix} a & | & b \\ \hline & | & \\ \mathbf{0} & | & I \end{bmatrix} \quad \text{Eq. 3-74}$$

with an inverse

$$[\bar{D}]^{-1} = \begin{bmatrix} a^{-1} & | & -a^{-1}b \\ \hline & | & \\ \mathbf{0} & | & I \end{bmatrix} \quad \text{Eq. 3-75}$$

Since the members of $\{u_d'\}$ are restrained to zero value,

$$\{u_b\} = [D_r]\{u_d\} \quad \text{Eq. 3-76}$$

where $[D_r]$ is the $(6 \times r)$ partitioned matrix given by

$$[D_r] = \begin{bmatrix} a^{-1} \\ - \\ 0 \end{bmatrix} \quad \text{Eq. 3-77}$$

The $[D_i]$ matrix is formed as before and the $[S]$ matrix is then

$$[S] = [D_i][D_r] \quad \text{Eq. 3-78}$$

Although this procedure will replace all deleted rigid body motions, it is not necessary to do this if a stiffness matrix rather than a flexibility matrix is input. It is, however, a highly recommended procedure because it will eliminate errors due to nonsatisfaction of rigid body properties by imprecise input data.

The stiffness matrix of the element written in partitioned form is

$$[K_{ee}] = \begin{bmatrix} K_{ii} & | & K_{id} \\ - & + & - \\ K_{id}^T & | & K_{dd} \end{bmatrix} \quad \text{Eq. 3-79}$$

When the flexibility formulation is used, the program evaluates the partitions of $[K_{ee}]$ from $[Z]$ and $[S]$ as follows:

$$[K_{ii}] = [Z]^{-1} \quad \text{Eq. 3-80}$$

$$[K_{id}] = [Z]^{-1}[S] \quad \text{Eq. 3-81}$$

$$[K_{dd}] = [S]^T[Z]^{-1}[S] \quad \text{Eq. 3-82}$$

If a stiffness matrix, $[k]$, rather than a flexibility matrix is input, the partitions of $[K_{ee}]$ are

$$[K_{ii}] = [k] \quad \text{Eq. 3-83}$$

$$[K_{id}] = -[k][S] \quad \text{Eq. 3-84}$$

$$[K_{dd}] = -[S]^T[k][S] \quad \text{Eq. 3-85}$$

No internal forces or other output data are produced for the general element.

3.12 Direct Matrix Input on Grid Points (DMIG)

The Bulk Data entry DMIG can be used to input a stiffness (or mass) matrix which connects specified degrees-of-freedom. The matrix so defined will be added to the stiffness (or mass) matrix computed from finite element properties.

The DMIG entry includes provisions for unsymmetric terms and complex values, both of which are useful in dynamic analysis. These provisions should not be used in static or normal modes. Note that an entry in the Case Control Section is required (K2GG = NAME for a stiffness matrix or M2GG = NAME for a mass matrix). See “[Case Control Commands](#)” on page 175 of the *MSC.Nastran Quick Reference Guide* for general instructions regarding the Case Control Section.

The primary application of the DMIG Bulk Data entry is to enter stiffness and mass data for parts of the structure which are obtained from another computer run. The format is cumbersome (two matrix terms per continuation entry) and the matrix should be input to high precision (see “[Use of Parameters](#)” on page 35 for a discussion of double-field entries). For stiffness matrices only, the GENEL Bulk Data entry is an alternative for manually inputting data.

3.13 p-Elements

p-elements are elements that have variable degrees-of-freedom. The user can specify the polynomial order for each element (Px, Py, and Pz) and the program will generate the degrees-of-freedom required.

There are a total of six different forms of p-elements that are defined by connection entries:

CTETRA	Four-sided solid element specified by 4 corner grid points
CPENTA	Five-sided solid element specified by 6 corner grid points
CHEXA	Six-sided solid element specified by 8 corner grid points
CTRIA	Three-sided curved shell element specified by 3 corner grid points
CQUAD	Four-sided curved shell element specified by 4 corner grid points
CBEAM	A curved beam element specified by its 2 end grid points

The properties of the p-elements will be specified using the PSOLID, PSHELL, PBEAM, or PBEAML entry. These elements may use either isotropic materials as defined on the MAT1 entry or anisotropic materials as defined on the MAT9 entry. The material coordinate system can use the basic system (0), any defined system (integer > 0), or the element coordinate system (-1 or blank).

Geometry

The model geometry of the p-elements can be done by

1. Supplying additional POINT entries to define the geometry of the edges, i.e., in the FEEDGE entry, the user needs to specify GEOMIN = POINT. In this case, when the FEEDGE entry contains the ID of one point, then the FEEDGE is considered to have a quadratic geometry. When the FEEDGE entry contains the ID of two points, then the FEEDGE is considered to have a cubic geometry. If there are no additional points specified on the FEEDGE entry, then the FEEDGE is considered to have a linear geometry (see [Figure 3-51](#)).

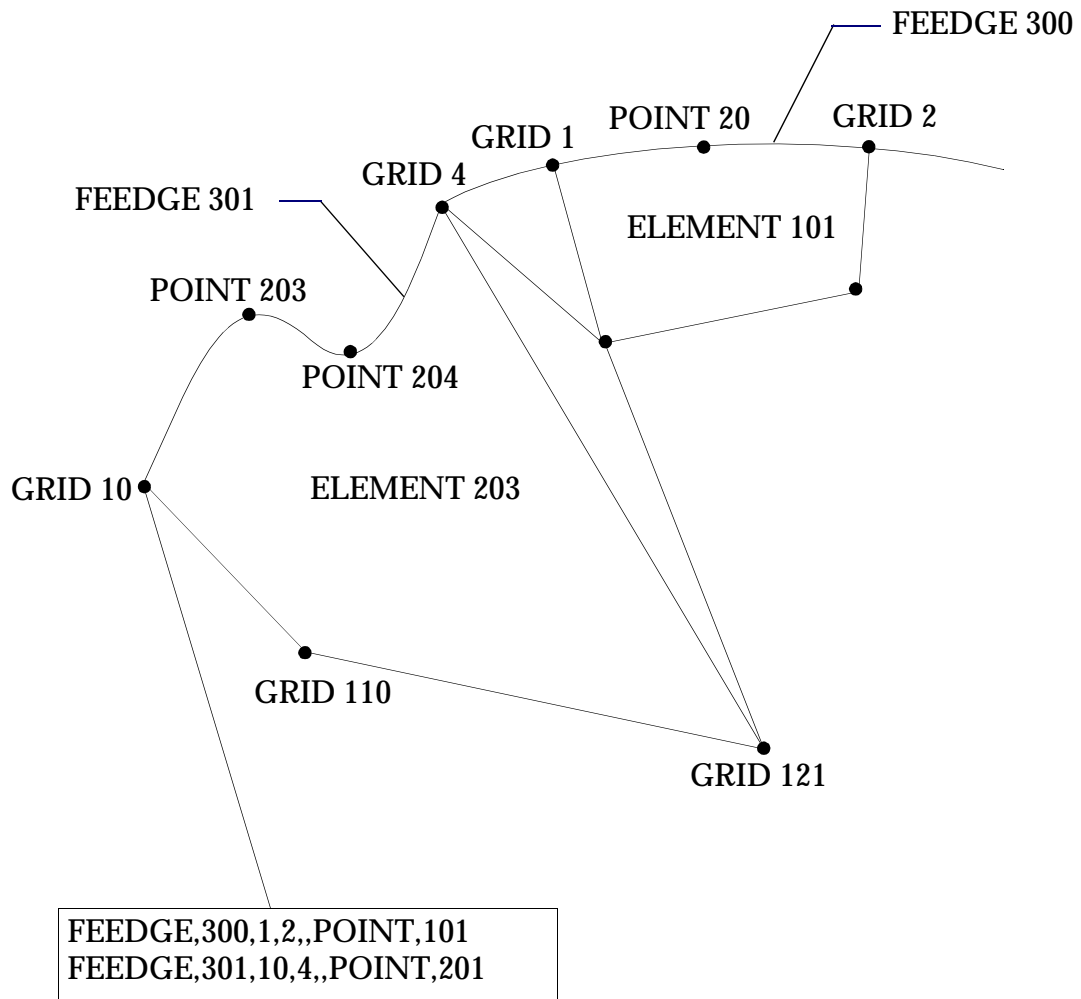


Figure 3-51 Specifying Geometry Using GEOMIN = POINT Method

2. Supplying the actual geometry, using the GMCURV and GMSURF entries, to define the geometry of the edges. The user needs to specify GEOMIN = GMCURV in the FEEDGE entry and providing a SURFID on the FEFACE entry (see [Figure 3-52](#)).

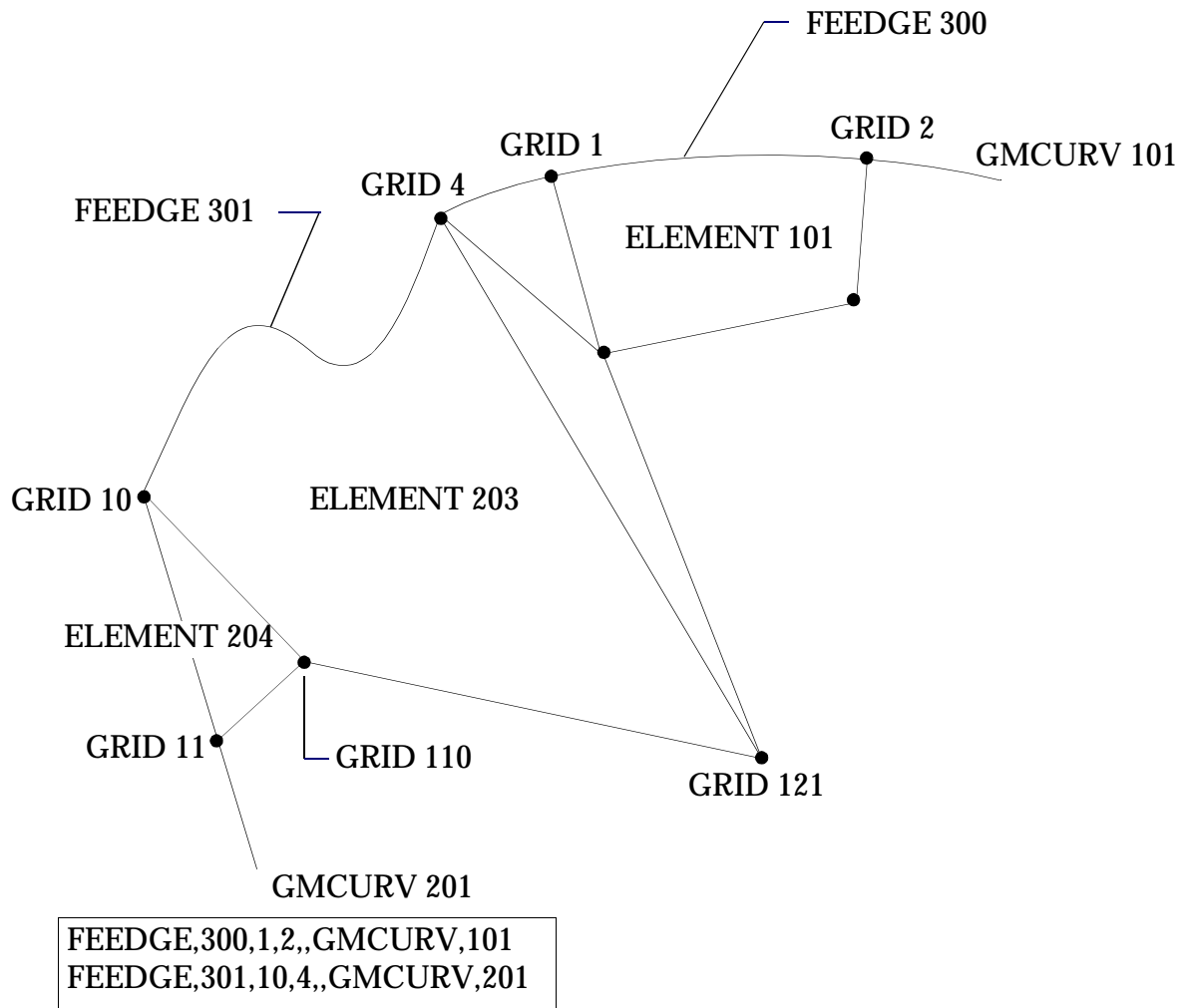


Figure 3-52 Specifying Geometry Using GEOMIN = GMCURV Method

In general, the geometry specification for any edge using a FEEDGE entry overrides the geometry specification for that edge via the FEFACE entry. In other words, the geometry of the edges belonging to both a FEEDGE and a FEFACE will be calculated using the data supplied on the FEEDGE entry. In addition, whenever two or more GMSURF are intersecting and FEFACE entries referring to the individual GMSURF entries are supplied, an additional FEEDGE entry must be supplied for the edges that are common to the multiple surfaces.

Bulk entries defining p-elements with p-value greater than 1 are:

ADAPT	GMSPC
DEQATN	GMSURF
FEEDGE	OUTPUT

FEFACE	OUTRCV
GMBC	PSET
GMCORD	PVAL
GMCURV	TABLE3D
GMLOAD	TEMPF

3.14 Hyperelastic Elements

The hyperelastic elements are intended for fully nonlinear (finite deformation) analysis including the effect of large strain and large rotation. Geometric nonlinearity is a subset of this type of analysis. In addition, the elements are especially designed to handle nonlinear elastic materials at the nearly incompressible limit. Volumetric locking avoidance is provided through a mixed formulation, based on a three field variational principle, with isoparametric displacement and discontinuous pressure and volumetric strain interpolations. Shear locking avoidance is provided through the use of second order elements. See “[Small Strain Elements](#)” on page 610.

The hyperelastic elements are defined on the same connection entries as the other shell and solid elements and are distinguished by their property entries. A PLPLANE or PLSOLID entry defines a hyperelastic element. The hyperelastic material, which is characterized by a generalized Rivlin polynomial form of order 5, applicable to compressible elastomers, is defined on the MATHP entry. See “[Hyperelastic Material](#)” on page 585 for a more detailed discussion.

Hyperelastic Solid Elements

The following elements are available:

1. CTETRA -- Four-sided solid element with 4 to 10 nodes.
2. CPENTA -- Five-sided solid element with 6 to 15 nodes.
3. CHEXA -- Six-sided solid element with 8 to 20 nodes.

See [Figure 3-41](#), [Figure 3-42](#), and [Figure 3-43](#) on solid elements.

There is no element coordinate system associated with the hyperelastic solid elements. All output is in the basic coordinate system. The following quantities are output at the Gauss points:

- Cauchy stresses

$$\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx}$$

- Pressure

$$p = \frac{1}{3}(\sigma_x + \sigma_y + \sigma_z)$$

- Logarithmic strains

$$\varepsilon = \sum_{l=1}^3 \ln \gamma_l N_l N_l^T$$

- Volumetric strain

$$J - 1 = \frac{dV - dV_0}{dV_0}$$

See “**Small Strain Elements**” on page 610 for definitions of these quantities.

Hyperelastic Plane Elements

These are plane strain elements defined on the following connectivities:

- CQUAD -- Quadrilateral element with 4 to 9 nodes. When the center node is missing, this element may also be specified on a CQUAD8 connectivity entry. When all edge nodes are missing, the CQUAD4 connectivity may be used.
- CTRIA3 -- Triangular element with 3 nodes.
- CTRIA6 -- Triangular element with 3 to 6 nodes.

Figure 3-53 shows the element connectivity for the CQUAD element.

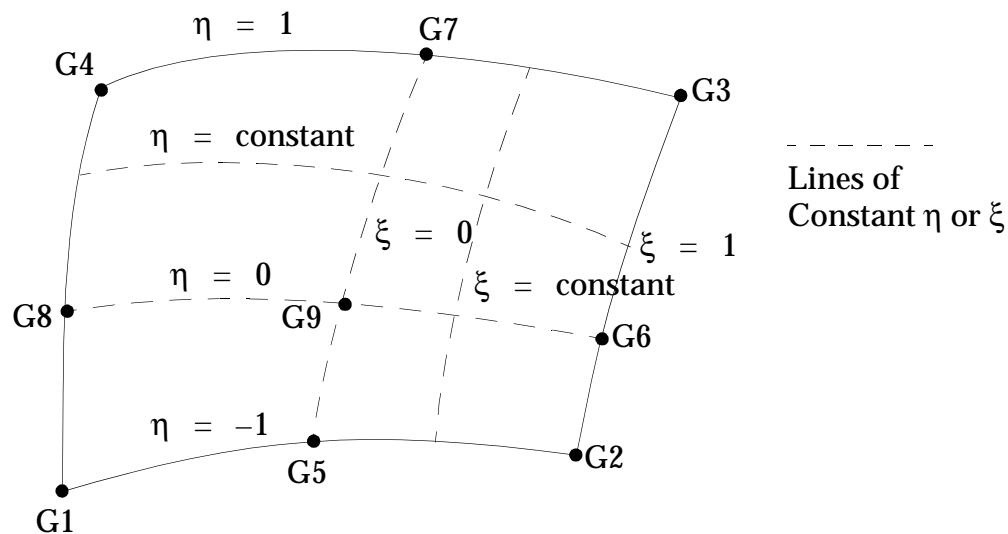


Figure 3-53 CQUAD Element

See **Figure 3-27** through **Figure 3-30** for the other elements. Note, however, that there is no element coordinate system associated with the hyperelastic plane elements. All output is in the CID coordinate system. Cauchy stresses σ_x , σ_y , σ_z , τ_{xy} , pressure $p = 1/3(\sigma_x + \sigma_y + \sigma_z)$, logarithmic strains and volumetric strain are output at the Gauss points.

The plane of deformation is the XY plane of the CID coordinate system, defined on the PLPLANE property entry. The model and all loading must lie on this plane, which, by default, is the XY plane of the basic coordinate system. The displacement along the Z axis of the CID coordinate system is zero or constant.

Hyperelastic Axisymmetric Elements

The axisymmetric hyperelastic elements are defined on the following connectivity entries:

- CQUADX -- Quadrilateral axisymmetric element with 4 to 9 nodes.
- CTRIAX -- Triangular axisymmetric element with 3 to 6 nodes.

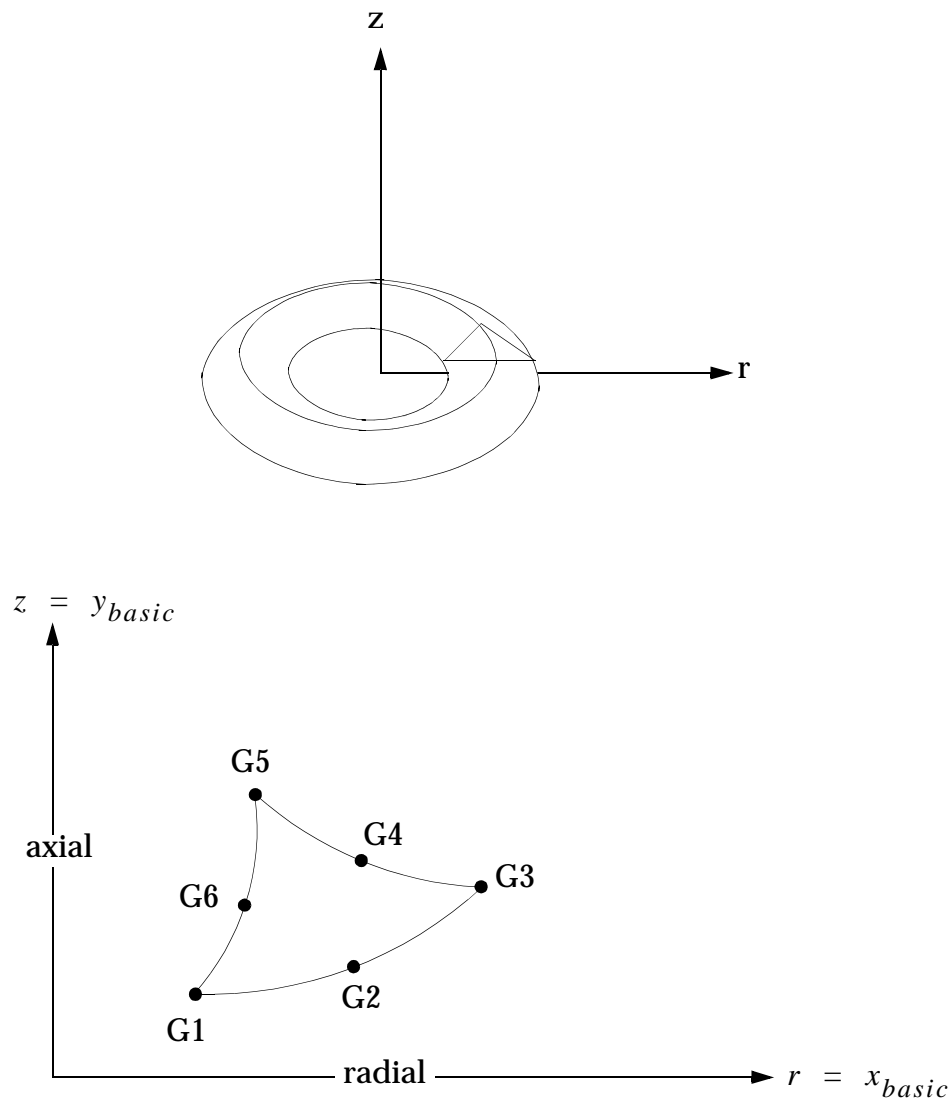


Figure 3-54 The CTRIAX Hyperelastic Coordinate System Element

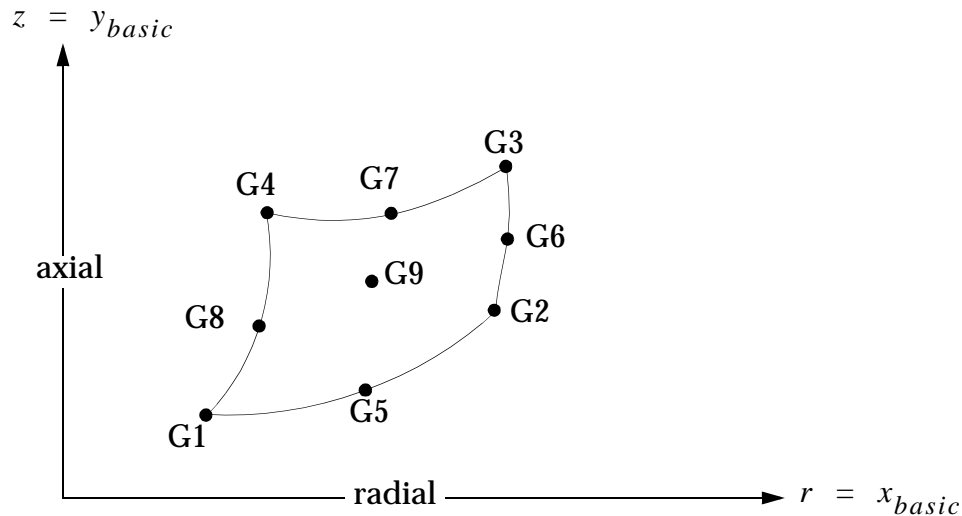


Figure 3-55 The CQUADX Hyperelastic Coordinate System Element

The plane of deformation is the XY plane of the basic coordinate system, and Y is the axisymmetry axis. The model and all loading must lie on this plane.

All output is in the basic coordinate system. Cauchy stresses σ_x (radial), pressure σ_y (axial), σ_z (circumferential), τ_{xy} , pressure $p = 1/3(\sigma_x + \sigma_y + \sigma_z)$, logarithmic strains and volumetric strain are output at the Gauss points.

See [Table 13-3](#) of “ ” on page 568 for a list of Hyperelastic elements and their characteristics.

Pressure loads, specified on the PLOAD4 and PLOADX1 Bulk Data entries respectively, with follower force characteristics, are available for the solid and axisymmetric elements. A pressure load may not be specified on the plane strain elements.

Temperature loads may be specified for all hyperelastic elements on the TEMP and TEMD entries. The hyperelastic material, however, may not be temperature dependent. Temperature affects the stress-strain relation as indicated in “[Hyperelastic Material](#)” on page 585.

GPSTRESS and FORCE (or ELFORCE) output is not available for hyperelastic elements.

3.15 Bushing Elements

The bushing (generalized spring and damper) elements consist of the following:

- CBUSH
- CBUSH1D

3.15.1 Bush Element (CBUSH)

The generalized spring-damper element CBUSH is a structural scalar element connecting two noncoincident grid points, or two coincident grid points, or one grid point with an associated PBUSH entry. This combination is valid for any structural solution sequence. To make frequency dependent the PBUSH need only have an associated PBUSHT Bulk Data entry. The PBUSHT entry, for frequency dependency, is only used in SOL 108 and SOL 111. The PBUSHT entry can be used to define load-displacement dependency in SOL 106.

In modal frequency response, the basis vectors (system modes) $[\phi]$ will be computed only once in the analysis and will be based on nominal values of the scalar frequency dependent springs. In general, any change in their stiffness due to frequency will have little impact on the overall contribution to the structural modes.

The stiffness matrix for the CBUSH element takes the diagonal form in the element system:

$$K_o = \begin{bmatrix} k_u & & & & & & \\ & k_v & & & & & \\ & & k_w & & & & \\ & & & k_{\theta_x} & & & \\ & & & & k_{\theta_y} & & \\ & & & & & k_{\theta_z} & \\ & & & & & & \end{bmatrix}$$

For the B matrix replace the k terms with b.

When transformed into the basic system, there is coupling between translations and rotations, thus ensuring rigid body requirements.

The element axes are defined by one of the following procedures:

1. If a CID is specified then the element x-axis is along T1, the element y-axis is along T2, and the element z-axis is along T3 of the CID coordinate system. The options GO or (X1,X2,X3) have no meaning and will be ignored. Then $[T_{ab}]$ is computed directly from CID.
2. For noncoincident grids ($GA \neq GB$), if neither options GO or (X1,X2,X3) is specified and no CID is specified, then the line AB is the element x-axis. No y-axis or z-axis need be specified. This option is only valid when K1 (or B1) or K4 (or B4) or both on the PBUSH entry are specified but K2, K3, K5, K6 (or B2, B3, B5, B6) are not specified. If K2, K3, K5, K6 (or B2, B3, B5, B6) are specified, a fatal message will be issued. Then $[T_{ab}]$ is computed from the given vectors like the beam element.

DMAP Operations

Direct Frequency Response

Nominal Values. The following matrices are formed only once in the analysis and are based on the parameter input to EMG of /' / implying that the nominal values only are to be used for frequency dependent springs and dampers.

$$K_{dd} = (1 + ig)K_{dd}^1 + K_{dd}^{2x} + iK_{dd}^4$$

$$M_{dd} = M_{dd}^1 + M_{dd}^2$$

$$B_{dd} = B_{dd}^1 + B_{dd}^{2x}$$

ESTF. The following matrices are formed at each frequency in the analysis and are based on the parameter input to EMG of /'ESTF' /

$$\Delta K_{dd}^f = (1 + ig)\Delta K_{dd}^{1f}$$

$$\Delta K_{dd}^{4f} = g_e^f[\Delta k_j^f] \quad j = 1 \rightarrow 6$$

$$\Delta B_{dd}^f = \Delta B_{dd}^{1f}$$

ESTNF. The following matrices are formed at each frequency in the analysis and are based on the parameter input to EMG of /'ESTNF' /

$$\Delta K_{dd}^{4f \text{ nominal}} = [\Delta g_e^f k_j^{f \text{ nominal}}] \quad j = 1 \rightarrow 6$$

and j runs through the stiffness values specified for the CBUSH element or $j = 1$ for the CELAS1 and CELAS3 elements.

Then at each frequency form:

$$K_{dd} \leftarrow K_{dd} + \Delta K_{dd}^f + i\Delta K_{dd}^{4f} + i\Delta K_{dd}^{4f \text{ nominal}}$$

$$B_{dd} \leftarrow B_{dd} + \Delta B_{dd}^f$$

Then the equation to be solved is:

$$[-\omega^2 M_{dd} + i\omega B_{dd} + K_{dd}][u_d] = [P_d]$$

Modal Frequency Response

Basis Vector and Nominal Values. The basis vector matrix $[\phi]$ (system modes) is formed only once in the analysis using nominal values for frequency dependent elements.

The following matrices are formed only once in the analysis and are based on the parameter input to EMG of ' ' / implying that the nominal values only are to be used for frequency dependent springs and dampers.

$$K_{dd}^2 = K_{dd}^{2x} + igK_{dd}^1 + iK_{dd}^4$$

$$B_{dd}^2 = B_{dd}^1 + B_{dd}^{2x}$$

ESTF. The following matrices are formed at each frequency in the analysis and are based on the parameter input to EMG of 'ESTF' /

$$\Delta K_{dd}^{2f} = ig\Delta K_{dd}^{1f}$$

$$\Delta K_{dd}^{4f} = g_e^f[\Delta k_j^f] \quad j = 1 \rightarrow 6$$

$$\Delta B_{dd}^{2f} = \Delta B_{dd}^{1f}$$

ESTNF. The following matrices are formed at each frequency in the analysis and are based on the parameter input to EMG of /'ESTNF'/

$$\Delta K_{dd}^{4f\ nominal} = [\Delta g_e^f k_j^{f\ nominal}] \quad j = 1 \rightarrow 6$$

and j runs through the stiffness values specified for the CBUSH element or j = 1 for the CELAS1 and CELAS3 elements.

Then at each frequency form:

$$K_{dd}^2 \leftarrow K_{dd}^2 + \Delta K_{dd}^f + i\Delta K_{dd}^{4f} + i\Delta K_{dd}^{4f\ nominal}$$

$$B_{dd}^2 \leftarrow B_{dd}^2 + \Delta B_{dd}^f$$

The GKAM module will then produce:

$$K_{hh} = [k] + [\Phi]^T [K_{dd}^2] [\Phi]$$

$$B_{hh} = [b] + [\Phi]^T [B_{dd}^2] [\Phi]$$

$$M_{hh} = [m] + [\Phi]^T [M_{dd}^2] [\Phi]$$

Then the equation to be solved is:

$$[-\omega^2 M_{hh} + i\omega B_{hh} + K_{hh}] [u_h] = [P_h]$$

Element force calculation:

Frequency:

$$F_e = [(1 + ig_t)K_e + i\omega B_e] U_e$$

where

$$g_t = g + g_e$$

Static:

$$F_e = K_e U_e$$

Transient:

$$F_e = K_e U_e(t) + \left[B_e + \left(\frac{g}{w3} + \frac{g_e}{w4} \right) K_e \right] \dot{U}_e(t)$$

CBUSH Stiffness and Structural Damping Matrices. If both stiffness K and structural damping g_e is specified each with their own frequency dependent tables, matrix terms of the following form are created:

$$K_j^f + i \cdot g_e^f \cdot K_j^f$$

If stiffness K with frequency dependent tables and non frequency dependent structural damping g_e are specified, matrix terms of the following form are created:

$$K_j^f + i \cdot g_e^0 \cdot K_j^f$$

If non frequency dependent stiffness K with frequency dependent structural damping g_e are specified, matrix terms of the following form are created:

$$K_j^0 + i \cdot g_e^f \cdot K_j^0$$

In the above three expressions, the superscript f denotes frequency dependent and the superscript 0 denotes a nominal value. The subscript j implies the j -th degree of freedom of the CBUSH element. The real term goes into the element stiffness matrix and the imaginary term goes into the element κ^4 matrix.

3.15.2 Bush1d Element (CBUSH1D)

The BUSH1D is a one dimensional version of the BUSH element (without the rigid offsets) and supports large displacements.

The element is defined with the CBUSH1D and a PBUSH1D entry. The user may define several spring or damping values on the PBUSH1D property entry. It is assumed that springs and dampers work in parallel. The element force is the sum of all springs and dampers.

The BUSH1D element has axial stiffness and axial damping. The element includes the effects of large deformation. The elastic forces and the damping forces follow the deformation of the element axis if there is no element coordinate system defined. The forces stay fixed in the x-direction of the element coordinate system if the user defines

such a system. Arbitrary nonlinear force-displacement and force-velocity functions are defined with tables and equations. A special input format is provided to model shock absorbers.

Benefits

The element damping follows large deformation. Arbitrary force deflection functions can be modeled conveniently. When the same components of two grid points must be connected, we recommend using force-deflection functions with the BUSH1D element instead of using NOLINi entries. The BUSH1D element produces tangent stiffness and tangent damping matrices, whereas the NOLINi entries do not produce tangent matrices. Therefore, BUSH1D elements are expected to converge better than NOLINi forces.

Output

The BUSH1D element puts out axial force, relative axial displacement and relative axial velocity. It also puts out stress and strain if stress and strain coefficients are defined. All element related output (forces, displacements, stresses) is requested with the STRESS Case Control command.

Guidelines

The element is available in all solution sequences. In static and normal modes solution sequences, the damping is ignored.

In linear dynamic solution sequences, the linear stiffness and damping is used. In linear dynamic solution sequences, the BUSH1D damping forces are not included in the element force output.

In nonlinear solution sequences, the linear stiffness and damping is used for the initial tangent stiffness and damping. When nonlinear force functions are defined and the stiffness needs to be updated, the tangents of the force-displacement and force-velocity curves are used for stiffness and damping. The BUSH1D element is considered to be nonlinear if a nonlinear force function is defined or if large deformation is turned on (PARAM,LGDISP,1). For a nonlinear BUSH1D element, the element force output is the sum of the elastic forces and the damping forces. The element is considered to be a linear element if only a linear stiffness and a linear damping are defined and large deformation is turned off.

Limitations

1. The BUSH1D element nonlinear forces are defined with table look ups and equations. Equations are only available if the default option ADAPT on the TSTEPNL entry is used, equations are not available for the options AUTO and TSTEP.
2. The table look ups are all single precision. In nonlinear, round-off errors may accumulate due to single precision table look ups.
3. For linear dynamic solution sequences, the damping forces are not included in the element force output.
4. The “LOG” option on the TABLED1 is not supported with the BUSH1D.

3.16 Interface Elements

The interface elements are primarily used when performing global local analyses. The current implementation is for p-elements.

For curve interfaces the following entries are used:

- GMBNDC -- Geometric Boundary - Curve
- GMINTC -- Geometric Interface - Curve
- PINTC -- Properties of Geometric Interface - Curve

For surface interfaces the following entries are used:

- GMBNDS -- Geometric Boundary - Surface
- GMINTS -- Geometric Interface - Surface
- PINTS -- Properties of Geometric Interface - Surface

The interface elements use a hybrid variational formulation with Lagrange multipliers, developed by NASA Langley Research Center. There are displacement variables defined on the interface element, in order to avoid making the interface too stiff, such as a rigid element. There are also Lagrange multipliers defined on each boundary, which represent the forces between the boundaries and the interface element. This formulation is energy-based and results in a compliant interface.

3.16.1 Curve Interface Elements

The addition of interface elements allows dissimilar meshes to be connected over a common geometric boundary, instead of using transition meshes or constraint conditions. Primary applications where the analyst specifies the interface elements manually include: facilitating global-local analysis, where a patch of elements may be removed from the global model and replaced by a denser patch for a local detail, without having to transition to the surrounding area; and connecting meshes built by different engineering organizations, such as a wing to the fuselage of an airplane.

Primary applications where the interface elements could be generated automatically are related to automeshers, which may be required to transition between large and small elements between mesh regions; and h-refinement, where subdivided elements may be adjacent to undivided elements without a transition area.

Dissimilar meshes created by the analyst can occur with global-local analysis, where part of the structure is modeled as the area of primary interest in which detailed stress distributions are required, and part of the structure is modeled as the area of secondary interest through which load paths are passed into the area of primary interest.

Generally the area of primary interest has a finer mesh than the area of secondary interest and, therefore, a transition area is required. Severe transitions generally produce elements that are heavily distorted, which can result in poor stresses and poor load transfer into the area of primary interest. An example of using interface elements to avoid such transitions is shown in [Figure 3-56](#). Similarly, a patch of elements may be removed from the global model and replaced by a denser patch for local detail.

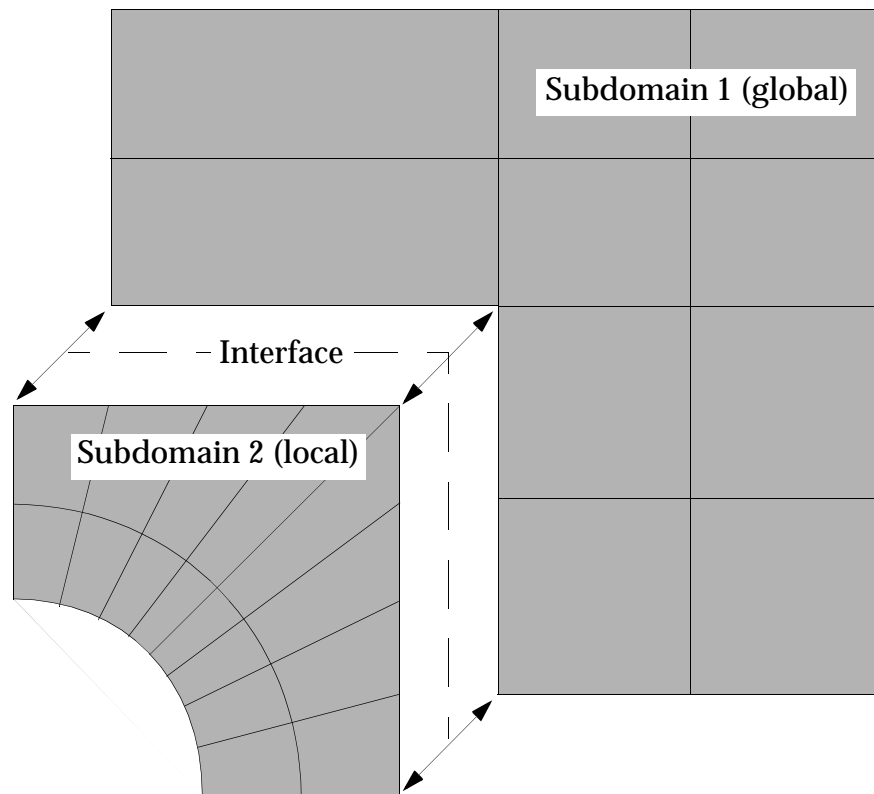


Figure 3-56 Example of Interface Elements (Exploded View)

In large system assemblies, different analysts or even different organizations may have created different components of the model, such as the wing and the fuselage of an airplane. Unless they have carefully coordinated their efforts, the finite element meshes of the different components may not match at the interfaces.

Dissimilar meshes generated by the analysis program can also arise with automeshers, which may be required to transition between large elements and small elements in a small area. Many automeshers generate tetrahedral meshes for solids, and distorted tetrahedra may be more susceptible to poor results. Interface elements are particularly useful when a transition is needed between large and small elements within a small area.

When mesh refinement is performed, subdivided elements may be adjacent to undivided elements with no room for a transition area. Without some kind of interface element, the subdivision would need to be carried out to the model boundary or otherwise transitioned out.

It is important to note that the interface elements provide a tool for connecting dissimilar meshes, but they do not increase the accuracy of the mesh. As with any interface formulation, the hybrid variational technology, which imposes continuity conditions in a weak form, can not increase the accuracy of the adjacent subdomains. For instance, if a single element edge on one boundary is connected to many element edges on the other boundary, the analysis is going to be limited to the accuracy of the boundary containing the single element edge. This restriction should be considered when deciding how close to the areas of primary interest to put the interface elements.

Guidelines

- The interface elements use the geometry of the boundary with the least number of p-element edges, which consist of cubic polynomials. If the other boundaries have widely varying geometry, poor answers may result. Warnings may be issued, but no geometrical adjustment is performed.
- Sharp corners within the interface element may degrade accuracy. An alternative is to specify multiple interface elements.
- Connecting few elements to many may result in very high p-levels along the interface. The limiting factor on the accuracy will be those few elements.
- Interface elements will only connect common fields of different element types. No kinematic constraints are enforced. Shell p-elements have five fields, and beam p-elements have six. Solid p-elements have three fields, but connecting edges of solid elements may lead to stress singularities.
- Constraining the boundaries may lead to unstable results. The most common constraint on a boundary is at an end point. Cases with constraints include multiple interfaces at the same point and end points of different boundaries connected together. Such unstable results are indicated by high epsilons or nonphysical modes.

- Superelements have not been implemented.
- The value of epsilon and the shapes of modes of primary interest should be checked to detect unstable results.
- For the rank deficient matrices, the sparse solver should be used for linear statics and the Lanczos solver for normal modes. `SYSTEM(166)=4` must also be set for normal modes.
- Since the boundaries are physically distinct, certain functions, such as shell normals and stress discontinuities in the error estimator, will not be applied across the interface.
- Contour plots may show differences across the interface because of the different view meshes, even though the solutions may be identical.

3.16.2 Surface Interface Elements

In finite element analysis the issue of connecting dissimilar meshes often arises, especially when refinement is performed. One method of connecting these dissimilar meshes is to use interface elements. An example in which patches of elements have been removed from the global model and replaced by denser patches for local details is shown in [Figure 3-57](#), where the boundaries of the patches are bold.

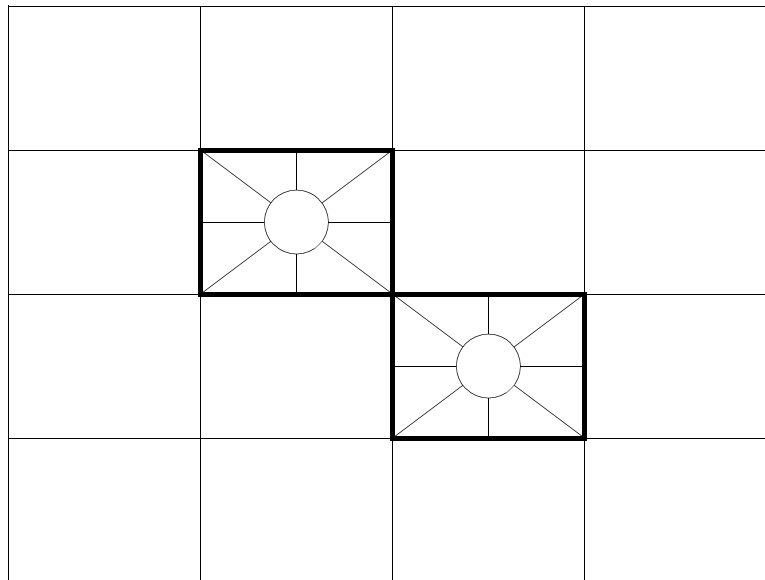


Figure 3-57 Example of Dissimilar Mesh from Global/Local Analysis

Implementation

Surface interface elements have been implemented for p-elements:

- The surface interface elements connect p-element faces. These can be either faces of solid p-elements or shell p-elements, and the faces may be either quadrilateral or triangular.
- The interface elements are geometry-based. They use the same geometry as the p-element boundaries.
- The interface elements connect only corresponding displacement fields. They do not have kinematic constraints, such as connecting shell rotations to solid translations.
- There are three methods of defining the subdomain boundaries of solid or shell p-element faces (GMBNDS). For the surface interface, each boundary may be defined using the GMSURF with which the finite element faces are associated; the FEFACEs defining the finite element faces; or in the most basic form, the GRIDs over the finite element faces.
- Once the boundaries have been defined, they must be associated with the interface elements. This is accomplished by referencing the boundaries in the interface element definition (GMINTS).
- Since the interface elements consist only of the differences in displacement components weighted by the Lagrange multipliers, there are no conventional element or material properties. The property Bulk Data entry (PINTS) specifies a tolerance for the interface elements, which defines the allowable distance between the subdomain boundaries; and a scaling factor, which may improve the conditioning of the Lagrange multipliers.

Output

The interface elements have no output of their own. However, they do cause changes in the customary output:

- Internal node, edge, and face degrees-of-freedom are generated for the interface elements. Because of their formulation, use of the parameter AUTOSPC, which is the default in the allowable solution sequences, detects them as singular. This can make the Grid Point Singularity Table (GPST) larger than expected. These degrees-of-freedom will also appear in the USET table.
- The interface elements may generate high or negative matrix/factor diagonal ratios. If there are no other model errors, these messages may be ignored and PARAM, BAILOUT, -1 may be used to continue.

Guidelines

- The interface elements use the geometry of the boundary with the least number of degrees-of-freedom, which consist of cubic polynomials. If the other boundaries have widely varying geometry, poor answers may result. Warnings may be issued, but no geometrical adjustment is performed.
- Sharp corners within the interface element may degrade accuracy. The preferred alternative is to specify multiple interface elements. An example of multiple interface elements is solved as the second example.
- Connecting few elements to many will not improve the accuracy along the interface. The limiting factor on the accuracy will be those few elements. For example, if one boundary has one element and the other boundary has four, the accuracy will be limited to that one element.
- Interface elements will only connect common displacement fields of different element types. No kinematic constraints are enforced. Shell p-elements have five fields, and solid p-elements have three. Therefore the rotational fields of the shell p-elements will not be connected to the solid p-elements.
- The sparse solver for linear statics and the Lanczos eigensolver for normal modes should be used. The sparse solver is the default solver for linear statics.
- The value of epsilon, which is the residual from the linear solution, and the shapes of modes of primary interest, which can best be evaluated graphically, should be checked to detect unstable results. Plots of displacements and stresses may also indicate unstable results. This would be visible as discontinuities in the displacements or stresses across the interface, which would imply a poor solution in that area.
- Since the boundaries are physically distinct, certain functions, such as shell normals and stress discontinuities in the error estimator, will not be applied across the interface.
- Contour plots may show differences across the interface because of the different view meshes. However, this is an indication of the results processing, not the original solution. A denser view mesh would reduce the differences.

Limitations

- Constraining the boundaries may lead to unstable results in certain cases. The most common constraint on a boundary is at an endpoint, such as a symmetry condition. Cases with constraints include multiple interfaces at the same point, such as a sharp corner; and endpoints of different boundaries connected together, such as an interior interface. Such unstable results are indicated by high epsilons or non-physical modes. They are also indicated by irregularities in the displacements or stresses.
- Superelements have not been implemented.

3.17 CWELD Element Enhancements

Introduction

The CWELD element was introduced in MSC.Nastran 2001. The CWELD can connect congruent as well as non congruent meshes. In MSC.Nastran 2004, we have extended the connectivity options to handle spot welds which connect more than one element per sheet and to handle connections other than surface patches.

Benefits

In MSC.Nastran 2001 the CWELD was limited to connect one element per shell sheet in patch-to-patch type connections using the ELEMID and GRIDID format, (see [Figure 3-58](#)). The connectivity could be unsymmetric so that results became unsymmetric for a model with symmetric mesh, loads, and boundary conditions. Two new patch-to-patch type connections are introduced, format ELPAT and PARTPAT, which can connect more than one element per sheet, see [Figure 3-59](#). The new formats preserve symmetry. In addition, the two element connectivity could miss elements if the mesh is so fine that the spot weld area spans over more than one element, see [Figure 3-60](#). The situation of meshes with element sizes smaller than spot weld sizes may occur in automotive models for crash analysis. The new formats can connect a maximum of 3x3 elements per shell sheet.

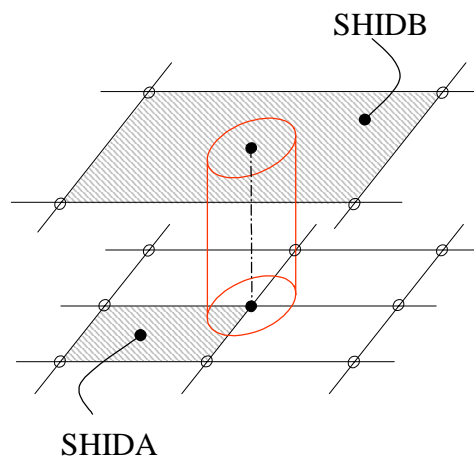


Figure 3-58 Connected elements for format ELEMID

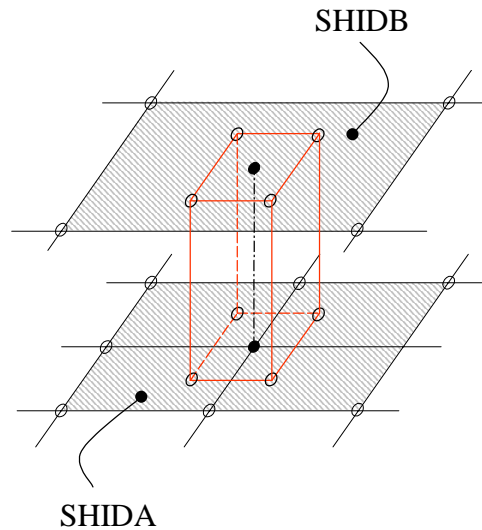


Figure 3-59 Connected elements for format ELPAT

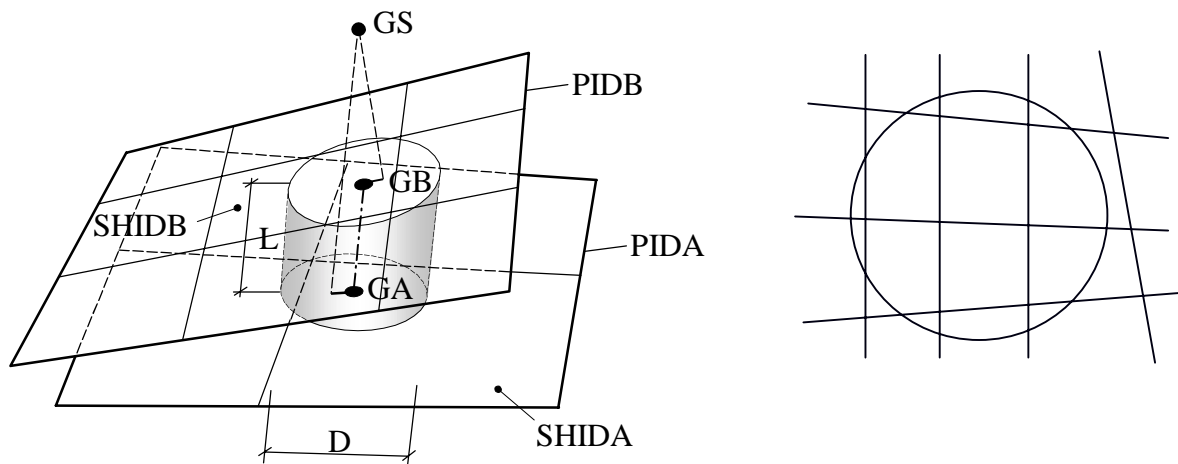


Figure 3-60 Spot weld covering a patch of elements.

In addition, the CWELD element in MSC.Nastran 2001 was restricted to handling connectivities of shell elements or surface patches made up of grid point sequences. In MSC.Nastran 2004, the point-to-patch type connectivity has been extended so that any element type, load or boundary condition can be connected to the CWELD. The element can now be used to model bearings of large masses, connectivities to springs, gaps, etc., see [Figure 3-61](#).

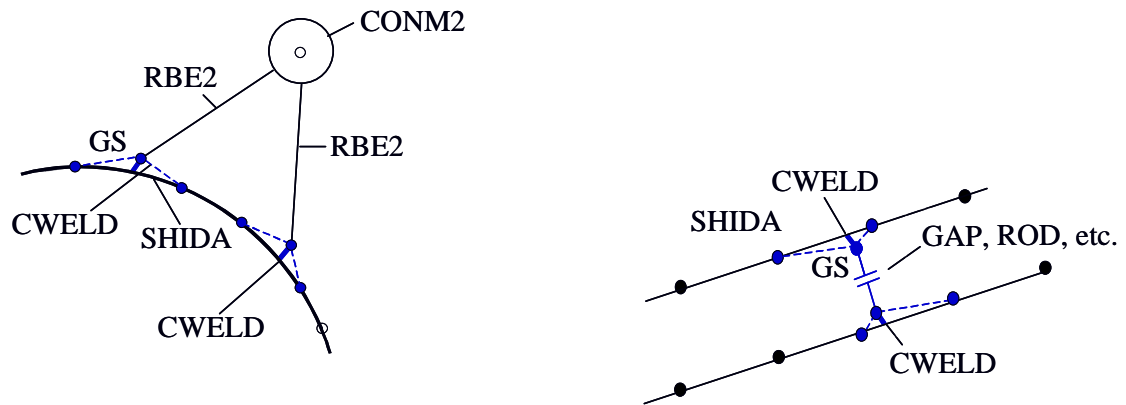


Figure 3-61 Modeling options using the point-to-patch type connectivity

Theory

Finding CWELD End Points GA and GB

If the CWELD end points GA and GB are not defined, then they are created from the locator point GS as follows. For the patch-to-patch type connectivity, the user supplied grid point GS is projected on to shell A and B according to [Figure 3-62](#). The end points GA and GB are created by a normal projection of GS to shell A and B if GS lies between shell A and B. A two step projection is used if GS lies outside of shell A and B. First, the auxiliary points GA' and GB' are created by a normal projection of GS to shell A and B. From the midpoint GC' of section GA',GB' a second normal projection is done that produces the final end points GA and GB. All the points described in this paragraph do not have degrees of freedom attached.

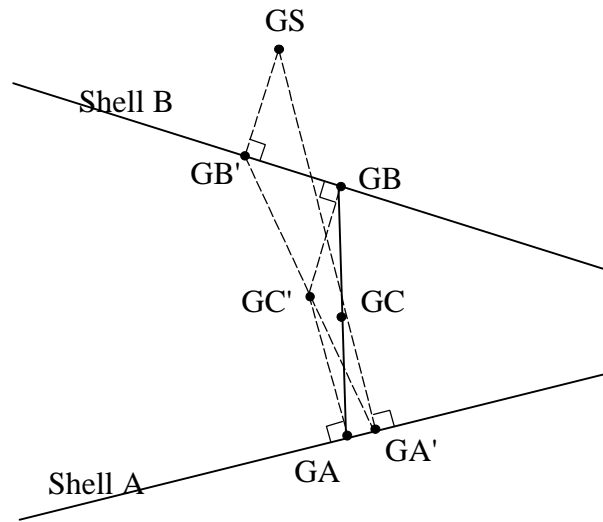


Figure 3-62 Projection of GS to determine end points GA and GB.

Element Coordinate System

The element x-axis points from GA to GB.

$$e_1 = \frac{x_B - x_A}{\|x_B - x_A\|} \quad \text{element x-axis}$$

In case of zero length, the normal of shell A is taken. All vector components are in basic unless noted otherwise.

Find the smallest component j of e_1

$$e_1^j = \min_{i=1,2,3} \{e_1^i\}.$$

In case of two equal components we take the one with the smaller i . The corresponding basic vector

$$b_j, \text{ e.g., for } j=3, b_3 = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix}$$

provides a good directional choice for e_2 . In addition, the vector e_2 must be orthogonal to e_1 .

$$\tilde{e}_2 = b_j - \frac{e_1^T b_j}{e_1^T e_1} e_1 \quad e_2 = \frac{\tilde{e}_2}{\|\tilde{e}_2\|} \text{ element y-axis}$$

and e_3 is just the cross product

$$e_3 = e_1 \times e_2 \quad \text{element z-axis}$$

The final transformation matrix is

$$T_{be} = \begin{bmatrix} | & | & | \\ e_1 & e_2 & e_3 \\ | & | & | \end{bmatrix}$$



Figure 3-63 Element Coordinate System.

Element Stiffness

The elastic part of the CWELD is a short beam from GA to GB with 2x6 degrees of freedom. The element is a shear flexible Timoshenko type beam. The Young's and shear modulus is taken from the material defined on the PWELD property entry. The cross sectional properties are calculated from the user supplied spot weld diameter D defined on the PWELD property entry,

$$E = \text{Young's modulus}$$

$$G = \text{shear modulus} = \frac{E}{2(1+\nu)}$$

$$A = \text{area} = \pi D^2/4$$

$$I_{zz} = I_{yy} \text{ moment of inertia} = \frac{\pi D^4}{64}$$

$$J = \frac{\pi D^4}{32}$$

The length L of the beam element is from end points GA to GB. If the user defines the type as SPOT on the PWELD property entry, then the effective element length L_e is always

$$L_e = \frac{1}{2}(t_A + t_B)$$

regardless of the true length L . The user supplied values for the Young's and shear modulus E and G are scaled by the ratio of true length to effective length

$$\tilde{E} = E \frac{L}{L_e} \quad \tilde{G} = G \frac{L}{L_e}$$

With the scaling, the spot weld stiffness is approximately constant for all elements. Extremely stiff elements from short lengths L and extremely soft elements from long lengths L are avoided. For zero length L , the beam element degenerates to 6 uncoupled springs. If the user does not define SPOT as the type, the true length is used if it is inside the range of

$$LDMIN \leq \frac{L}{D} \leq LDMAX$$

where $LDMIN$ and $LDMAX$ are user supplied parameters on the PWELD entry. By default $LDMIN = 0.2$ and $LDMAX = 5.0$. If the user does not want the true length adjusted at all, $LDMIN = 0$. and $LDMAX = 1.e+12$ may be specified.

Constraints Connecting GA and GB to Shell Grids

The end points GA and GB are connected to the shell grids of shell A and B, respectively, using constraint equations. The 3 translations at grid GA are connected to the 3 translations of the shell grid points using the interpolation functions of the corresponding shell surface. The 3 rotations at grid GA are connected to the 3 translations of the shell grid points using Kirchhoff conditions.

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix}_A = \sum N_I(\xi_A, \eta_A) \cdot \begin{Bmatrix} u \\ v \\ w \end{Bmatrix}_I$$

$$\theta_x^A = \frac{\partial w}{\partial y} = \sum N_{I,y} \cdot w_I$$

$$\theta_y^A = -\frac{\partial w}{\partial x} = -\sum N_{I,x} \cdot w_I$$

$$\theta_z^A = \frac{1}{2} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) = \frac{1}{2} (\sum N_{I,x} \cdot v_I - \sum N_{I,y} \cdot u_I)$$

For the formats GRIDID and ELEMID on the CWELD entry, the grid points I are grid points of shell elements or grid points of solid element faces. For the new formats ELPAT and PARTPAT, the grid points I are auxiliary points GAHI and GBHI, $I=1,4$, constructed according to [Figure 3-64](#) and [Figure 3-65](#). The auxiliary points are connected to shell element grids with a second set of constraints

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix}_I = \sum_K G_{IK} \cdot \begin{Bmatrix} u \\ v \\ w \end{Bmatrix}_K$$

where G_{IK} is a coefficient matrix derived from RBE3 type constraints. The grid points K are shell grid points.

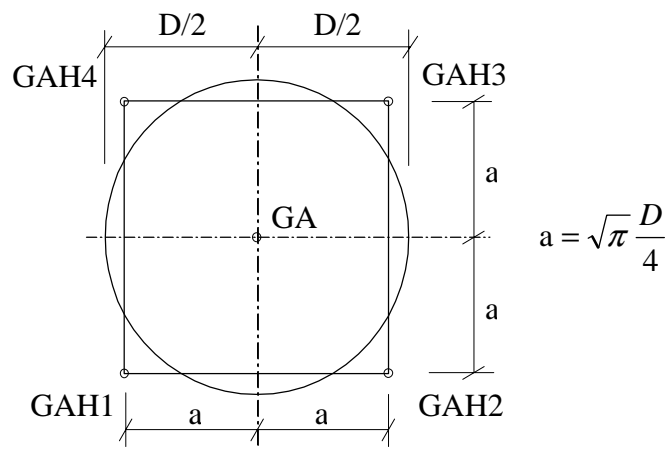


Figure 3-64 Equivalent square cross sectional area for a spot weld of diameter D

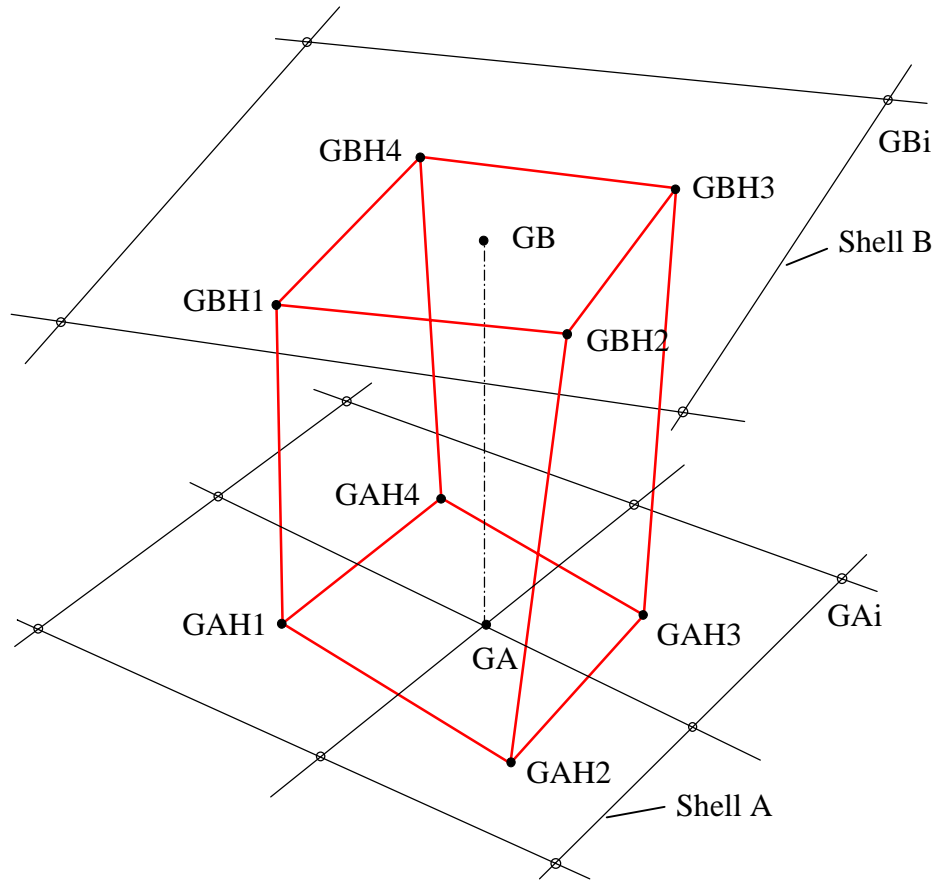


Figure 3-65 Auxiliary points GAH1 and GBH1 for formats ELPAT and PARTPAT

With the new formats ELPAT and PARTPAT, the element can connect from one to 3x3 elements on shell A and B, respectively, see [Figure 3-66](#).

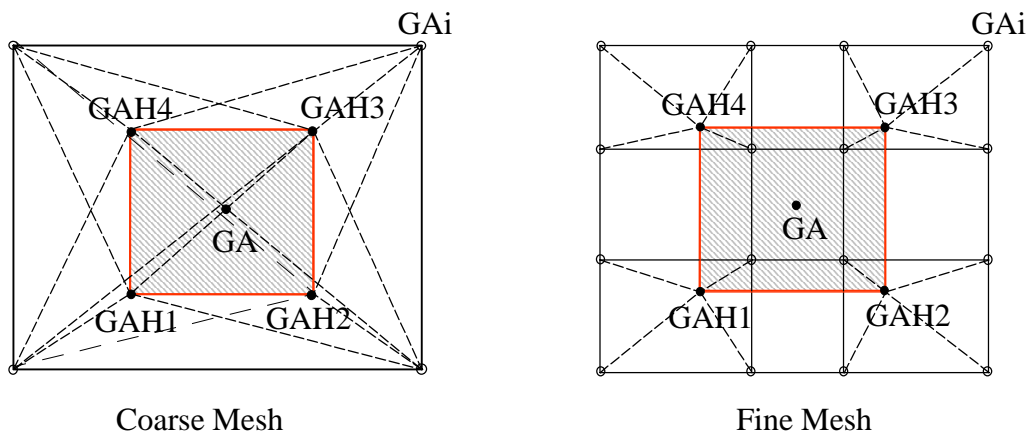


Figure 3-66 Connectivity for formats ELPAT and PARTPAT

With the constraint equations above, the degrees of freedom of grid points GA, GB and the auxiliary points GAHI, GBHI are condensed out on the element level. The final stiffness matrix of the CWELD consists of translational degrees of freedom of grid points on shell A and B, respectively. There are no extra degrees of freedom generated except for the case where the user requests explicit constraints by setting MSET=ON on the PWELD entry. The flag MSET=ON works only for the formats ELEMID and GRIDID.

Inputs

There are two new formats added to the three existing formats. The first new format, ELPAT, has the same input parameters as the format ELEMID, except for the new character string ELPAT.

```
CWELD, ELID, PWID, GS, , , ELPAT, , , +
+, SHIDA, SHIDB
```

where SHIDA and SHIDB are element ids of shell A and B, respectively. With ELPAT, the connectivity is extended to all neighboring shell elements which fall into the cross sectional area of the spot weld.

For the second new format, PARTPAT, the user defines shell A and B by property id PSIDA and PSIDB.

```
CWELD, ELID, PWID, GS, , , PARTPAT, , , +
+, PSIDA, PSIDB
```

With PARTPAT, all shell elements are connected which fall into the cross sectional area of the spot weld.

Outputs

The force output remains the same as in MSC.Nastran 2001. For internal processing, the following element types are created depending on the user specified formats.

Element Name	Element Number	User Specification
CWELD	200	ALIGN ELEMID, GRIDID with MSET=ON
CWELDC	117	ELEMID, GRIDID with MSET=OFF
CWELDP	118	ELPAT, PARTPAT

The output record of the CWELD forces is identical to the CBAR output record. The output is in the element coordinate system, see the explanations above how the element coordinate system is constructed. The forces are output at the end points GA and GB, see **Figure 3-67**.

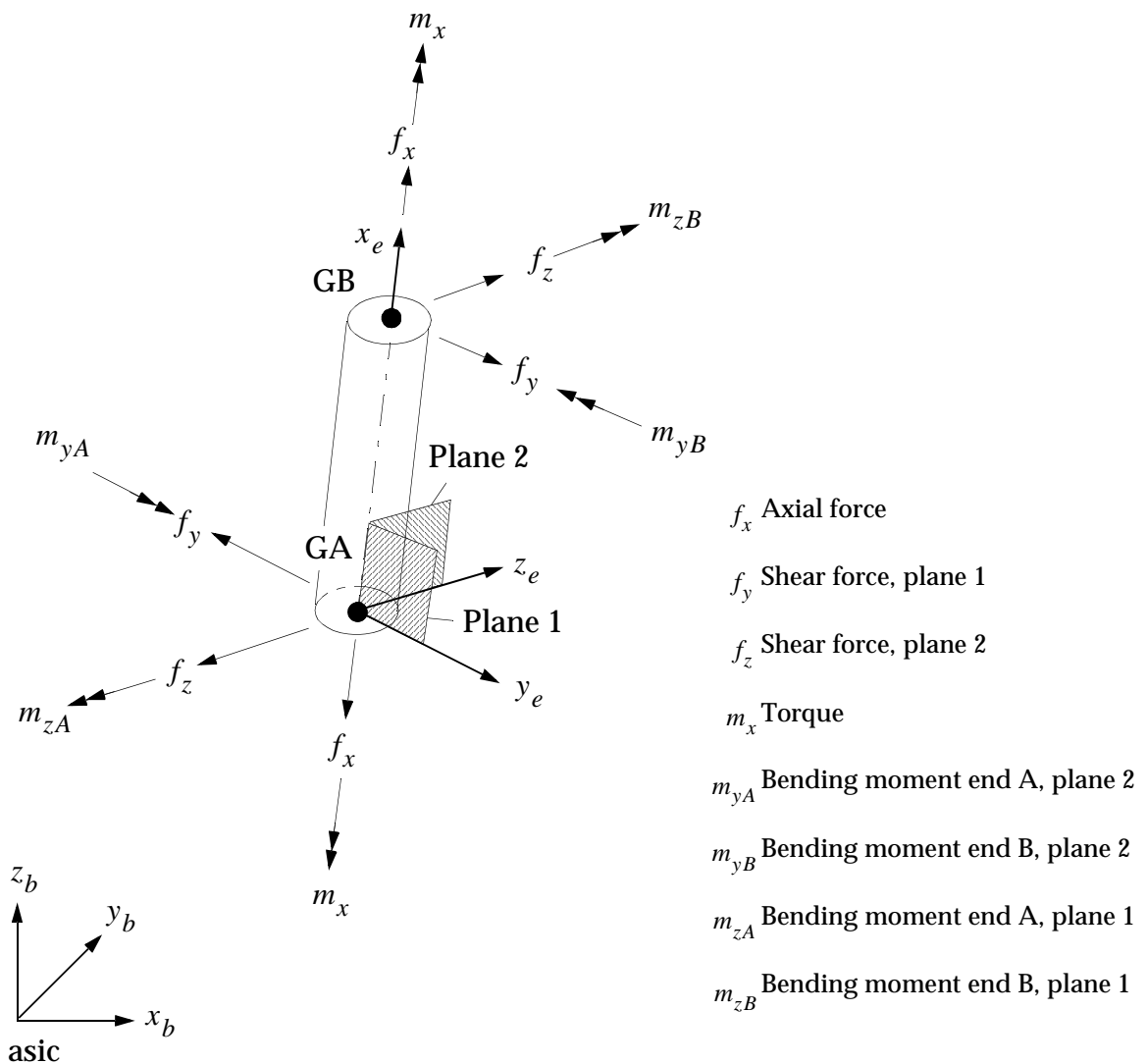


Figure 3-67 CWELD Element Forces

Guidelines and Limitations

Finding the connectivity is not straight forward for a complex topology. The user has an option to do a check out run first. In addition, the user may choose to let the program move GS in case a projection can not be found. Furthermore, the user may choose to reduce the diameter so that the spot weld can be placed near edges and corners. All parameters for the connectivity search are lumped on a new SWLDPRM Bulk Data entry. It is recommended to use the defaults first. If there are problems, it is recommended that the check out option be used next and request diagnostic prints.

The CWELD element is not designed to provide accurate stresses at the spot weld location. The stresses and spot weld forces are mesh dependent. The mesh dependency is more pronounced when the mesh size difference between shell A and B is large, for example, when the element connects a coarse mesh on shell A with a fine mesh on shell B.

A couple of limitations exist which will be removed in future versions.

- The CWELD is not available for nonlinear material and it stays linear in geometric nonlinear analysis.
- The element does not take preloads or initial stresses.
- Temperature strains are not available in the element.

The MSET option on the PWELD entry is only available for the formats ELEMID and GRIDID. The MSET default was ON in MSC.Nastran 2001, that means external m-set constraints were generated by default. In MSC.Nastran 2004, the default is MSET=OFF, that means the constraints are eliminated on the element level and there are no external m-set constraints generated. For the new formats ELPAT and PARTPAT, all constraints are eliminated on the element level, there are no external constraints generated.

Example

A symmetric hat profile with symmetric mesh, loads, and boundary conditions is used to demonstrate the difference between the old ELEMID and the new ELPAT formats, see [Figure 3-68](#). For the old ELEMID format, it may happen that the connectivity is defined in an unsymmetric manner, if the spot weld locator point GS is on a corner of an element, see [Figure 3-69](#). Then the format ELEMID produces unsymmetric results. With the new ELPAT format, even with a user defined unsymmetric connectivity, the internal connectivity is symmetric and the results are symmetric, see the following print out.

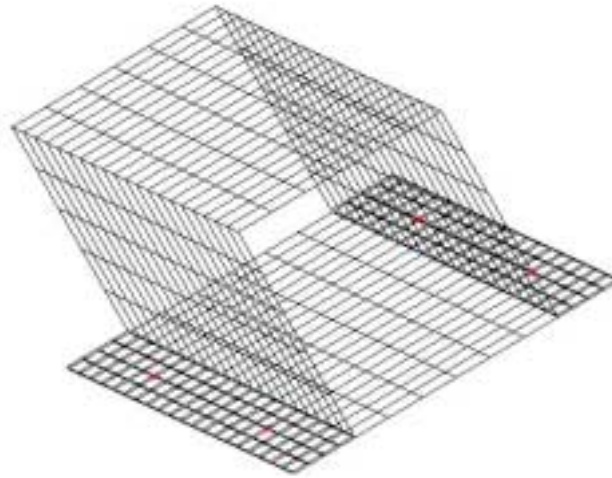


Figure 3-68 Hat Profile

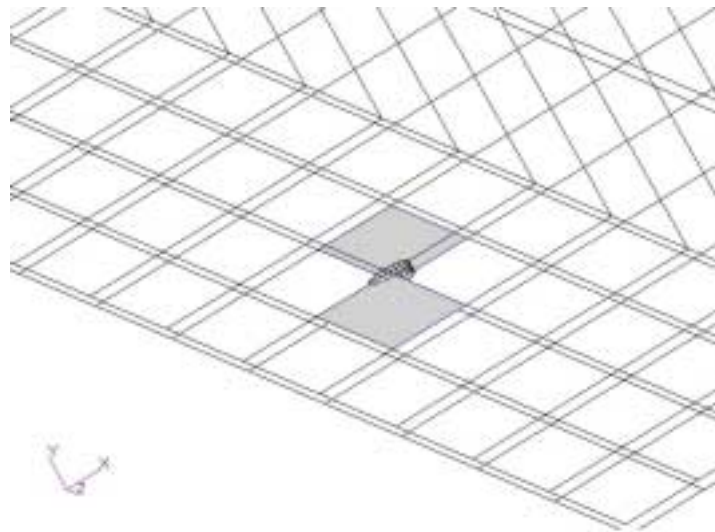


Figure 3-69 Unsymmetric Connectivity for Format ELEMID

The following is the echo of the input for the formats ELEMID and ELPAT:

```

$ Weld definition with format ELEMID
PWELD    10      1      3.
CWELD    761     10     10001  ELEMID
         259     22
CWELD    762     10     10002  ELEMID
         299     62
CWELD    763     10     10003  ELEMID
    
```

	639	402		
CWELD	764	10	10004	ELEMID
	679	442		

\$ Weld definition with format ELPAT

PWELD	10	1	3.	
CWELD	761	10	10001	ELPAT
	259	22		
CWELD	762	10	10002	ELPAT
	299	62		
CWELD	763	10	10003	ELPAT
	639	402		
CWELD	764	10	10004	ELPAT
	679	442		

Below is the print out for the element forces. The forces are not symmetric for the format ELEMID, see the first part with the title CWELDC. The forces come out symmetric for the format ELPAT, see the second part with the title CWELDP.

1 HAT SECTION WELDED BEAM, NON-SYM ELEMID OPTION		MAY 16, 2003 MSC.NASTRAN 5/15/03						
PAGE 16								
MZ								
0		SUBCASE 3						
FORCES IN WELD ELEMENTS (CWELDC)								
ELEMENT ID	BEND-MOMENT END-A		BEND-MOMENT END-B		- SHEAR -		AXIAL FORCE FX	
	PLANE 1 (MZ)	PLANE 2 (MY)	PLANE 1 (MZ)	PLANE 2 (MY)	PLANE 1 (FY)	PLANE 2 (FZ)		
TORQUE MX								
761	-1.422846E-02	1.642372E-01	-8.496036E-03	-3.994170E-01	-5.732428E-03	5.636542E-01	-5.534064E-04	
4.985318E-02								
762	-1.146957E-02	1.216688E-01	-1.872424E-02	-4.326421E-01	7.254668E-03	5.543109E-01	8.758141E-05	
4.411347E-02								
763	-2.167158E-02	-1.589983E-01	-1.698017E-02	4.059372E-01	-4.691408E-03	-5.649355E-01	5.010240E-04	
4.672464E-02								
764	-2.205327E-02	-1.208619E-01	-3.047966E-02	4.344850E-01	8.426391E-03	-5.553468E-01	-1.337890E-04	
4.600000E-02								
1 HAT SECTION WELDED BEAM, NON-SYM ELPAT OPTION		MAY 16, 2003 MSC.NASTRAN 5/15/03						
PAGE 16								
MZ								
0		SUBCASE 3						
FORCES IN WELD ELEMENTS (CWELDP)								
ELEMENT ID	BEND-MOMENT END-A		BEND-MOMENT END-B		- SHEAR -		AXIAL FORCE FX	
	PLANE 1 (MZ)	PLANE 2 (MY)	PLANE 1 (MZ)	PLANE 2 (MY)	PLANE 1 (FY)	PLANE 2 (FZ)		
TORQUE MX								
761	-9.314521E-04	1.277454E-01	4.733397E-03	-4.993157E-01	-5.664849E-03	6.270611E-01	-2.275192E-04	
5.223577E-02								
762	9.314521E-04	1.277454E-01	-4.733397E-03	-4.993157E-01	5.664849E-03	6.270611E-01	2.275192E-04	
5.223577E-02								
763	-9.314521E-04	-1.277454E-01	4.733397E-03	4.993157E-01	-5.664849E-03	-6.270611E-01	2.275192E-04	
5.223577E-02								
764	9.314521E-04	-1.277454E-01	-4.733397E-03	4.993157E-01	5.664849E-03	-6.270611E-01	-2.275192E-04	
5.223577E-02								

3.18 Element Geometry Checks

Overview

Mathematical models are usually composed of several different types of finite elements depending upon the simulation being performed and the type of structure being investigated. For each of these finite element formulations, thorough evaluations of the geometry presented are performed prior to and during element matrix generation operations. These geometry tests are typically used to determine whether (1) the element geometry is adequate for finite element matrix generation at all, and if so, (2) how close the geometry could be to producing poor formulations. Taking a beam element for example, a fatal geometry condition would be locations of the two end points that result in an element with zero length. An example of the second type of geometry check for a beam would be an excessive ratio of the length of the beam with offset vectors to the length of the beam without them. In the first case, the geometry check is fatal because the finite element cannot be formulated from the given geometry and the simulation cannot be done. In the second case, the geometry check is informational and the simulation analysis can proceed. It is the user's responsibility to inspect these informational messages and determine if the results are acceptable.

Several different basic geometry tests are performed; for example, duplicate grid point location tests, mid-side node location tests, warped element tests, and aspect ratio tests. One or more of these types of tests is performed using the geometry supplied for each finite element. Obviously, not all tests apply to every element type. Each one of the geometry tests is accompanied by a message that can generate up to four lines of informational output for an element if the test tolerance is violated. For models that employ a small number of elements, this amount of output is not overwhelming. However, the size and complexity of mathematical simulation models have continued to grow over time. It is no longer surprising to see models that generate hundreds of thousands of lines of output from these tests alone. In an effort to reduce the amount of output generated by geometry checks, an alternative output for reporting the results of the element geometry tests is available. The results for all tests for a single element are reported on a single line together with a visual indication of which tests have exceeded their tolerances and the severity of the failure. Default values for all test tolerances are defined such that geometry tests result in the same type of pass/fail decision as previous releases. In a few cases, additional tests are performed. At the same time, the GEOMCHECK Executive statement is available to give users more control over the test tolerances, their severity levels and the number of messages output.

GEOMCHECK Statement

The GEOMCHECK statement provides users with a way to override the default values supplied for all tests. (Refer to the Executive Statement, “**GEOMCHECK**” on page 128 of the *MSC.Nastran Quick Reference Guide*.) Except for a very few isolated instances, bad geometry that does not allow the formulation of finite element matrix data is always fatal and is not under user control. In these cases, the informational message formats have not been modified. Every test that fails will produce a message for every element. For the cases where geometry does not prevent finite element matrix generation, additional tests that evaluate various geometry parameters are performed and the results reported to the user. For these tests, the user can control both the test tolerance and the severity level of test failure. Using the CBAR element as an example, there is a test performed that evaluates the ratio of the CBAR’s length with offset effects to the length without offset effects. The tolerance for this test is 15%, meaning that if the length with offset is different from the length without the offset by more than fifteen percent, an informational message is issued and element processing continues. The user can change the tolerance if desired. The severity of the test failure can also be changed from informational to fatal if desired. In that case, any CBAR element that exceeds the offset ratio test tolerance will cause the job to stop after element matrix generation has been attempted for all elements.

The GEOMCHECK statement can only be used to modify geometry test activities for the CQUAD4, CQUADR, CTRIA3, CTRIAR, CBAR, CBEAM, CHEXA, CPENTA, and CTETRA finite elements at this time. As stated previously, only informational tests can be affected using the GEOMCHECK statement. Results of geometry generated by MSC.Nastran that prevent the generation of a finite element matrix cannot be modified using the GEOMCHECK statement.

Keywords are available that allow control over the number of messages to be generated (MSGLIMIT) as well as the severity of the message (MSGTYPE) and its associated effect on the job. Only a severity of FATAL will cause the job to abort after the element matrix generation module has executed. In this case, the identifier associated with failed tests is **FAIL** and the run will also terminate.

Geometry Test Descriptions

Several different types of geometry tests are performed. For each finite element type, the geometry must be tested to ensure that it is adequate to allow generation of element matrices. No attempt is made here to describe those tests. Rather, tests that optionally evaluate certain characteristics of the geometry are described. These tests

fall into several categories depending upon the element type. Thus, not all tests are applicable to every element type. The discussion that follows summarizes the optional geometry tests that are applicable to various element types.

CBAR and CBEAM

The only optional test available for these element types is the offset test. The length of the element offset is compared to the original length of the element. If the ratio of these lengths is greater than the tolerance, an informational message is issued identifying the element and its length with and without the effects of offset.

CQUAD4 and CQUADR

Four optional tests are performed on the quadrilateral shaped elements. These tests are:

1. Interior Angle test: This test evaluates the interior angles measured at each of the four corner grid points. If any one of the four angles exceeds minimum or maximum tolerance levels, an informational message is produced.
2. Taper test: Taper ratio for the quadrilateral element is defined to be the ratio of the area of the triangle formed at each corner grid point to one half the area of the quadrilateral. The largest of the four ratios is compared against the tolerance value. It may be noted that as the ratio approaches 1.0, the shape approaches a rectangle.
3. Skew test: This test evaluates the distortion or “parallelogram shape effect” by measuring the angle between lines that join the midpoints of opposite sides of the element. If the angle exceeds the tolerance, an informational message is produced.
4. Warp test: This test evaluates how far out of plane the four corner grid points are by measuring the distance of each point from a “mean” plane passing through the locations of the four points. The corner points are alternately H units above and H units below this mean plane. If the lengths of the diagonals of the element are denoted by D1 and D2, the warping coefficient is obtained from the equation $WC = H / 2(D1+D2)$. If this value exceeds the tolerance, an informational message is produced.

CTRIA3 and CTRIAR

Only one optional test is performed on the triangular shaped planar elements. It is the same as the interior angle test discussed previously under the CQUAD4 and CQUADR test description.

CHEXA, CPENTA and CTETRA

Five optional tests are performed on the solid element family. Not all tests are applicable to every element type. These tests are:

1. Aspect Ratio test: This test evaluates the ratio of the longest length (edge or height) to the shortest length (edge or height) encountered in the element. If the ratio exceeds the tolerance, an informational message is produced.
2. Edge Point Length Ratio test: This test evaluates the location of the “mid-side” nodes if any are present. The node should be located on the line connecting the two adjacent corner grid points at approximately the mid-way point. This determination is made in two ways. First, the distance of the node from the two corner nodes is measured and if the ratio of the distance from one node to the other exceeds the tolerance, an informational message is issued. Next, the angles between the lines connecting the mid-side node to its two adjacent corner nodes, and the line connecting the corner nodes themselves, is evaluated. If either of the angles exceeds 30 degrees, an informational message is issued.
3. Integration Point Jacobian Determinant test: This test evaluates the determinant of the Jacobian at each integration point. If it is zero, or changes sign from integration point to integration point, an informational message is issued. Note that detection of a zero value should always be fatal because element matrices calculated from such geometry rarely produce satisfactory analysis results.
4. Warped Face test: This test is applicable to CHEXA and CPENTA elements. It evaluates the warping coefficient of the quadrilateral faces of such elements in a manner similar to that discussed previously under the CQUAD4 element warping test description. If the warping coefficient exceeds the tolerance, an informational message is issued.
5. Grid Point Jacobian Determinant test: This test is applicable to CTETRA elements only. The test is the same as that done for the integration point test discussed previously except it uses the location of the corner grid points to perform the test. If any determinant is zero or changes sign, an informational message is produced.

Sample Output:

The following output from a small example provides a sample of the new message formats that are produced for geometry tests that can be controlled using the GEOMCHECK statement. For this output, most of the default test tolerances were modified so that the effect could be observed in the tolerance information lines of the

messages that were produced. Also note that a small table is generated at the end of all messages that summarizes the number of tests that actually exceeded the tolerance value for each element type as well as a list of the elements that produced the worst violations. The summary table itself can be produced by using the keyword `SUMMARY` on the `GEOMCHECK` statement.

*** USER INFORMATION MESSAGE 7555 (GMTSTD)
 FINITE ELEMENT GEOMETRY CHECK RESULTS EXCEED TOLERANCE LEVELS FOR THE FOLLOWING ELEMENTS.
 User Action: Use the GEOMCHECK (Executive Control Statement) keyword=value to change tolerance values if desired.
 A MINIMUM OF 100 OFFSET LENGTH RATIO TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.

TOLERANCE LIMIT IS: BAR_/BEAM_OFF(SET) = .15 (xxxx = LIMIT VIOLATED)
 ELEMENT TYPE ID LENGTH W/O OFFSET LENGTH W/ OFFSET OFFSET LENGTH RATIO
 BAR 7101 1.00000E+00 3.66367E+00 2.66 xxxx

*** USER INFORMATION MESSAGE 7555 (GMTSTD)
 FINITE ELEMENT GEOMETRY CHECK RESULTS EXCEED TOLERANCE LEVELS FOR THE FOLLOWING ELEMENTS.
 A MINIMUM OF 4 OFFSET LENGTH RATIO TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY WARN.

TOLERANCE LIMIT IS: BAR_/BEAM_OFF(SET) = -.02 (WARN = LIMIT VIOLATED)
 ELEMENT TYPE ID LENGTH W/O OFFSET LENGTH W/ OFFSET OFFSET LENGTH RATIO
 BEAM 8101 1.00000E+00 1.10372E+01 10.04 WARN
 BEAM 8102 1.00000E+00 1.00000E+00 .00 WARN

*** USER INFORMATION MESSAGE 7555 (EHEXGD)
 FINITE ELEMENT GEOMETRY CHECK RESULTS EXCEED TOLERANCE LEVELS FOR THE FOLLOWING ELEMENTS.
 A MINIMUM OF 100 EDGE LENGTH ASPECT RATIO (AR) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 EDGE NODE POINT LENGTH RATIO (ER) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 EDGE ANGLE VALUES ARE INDICATED BY AN * AFTER THE VALUE. THE TOLERANCE LIMIT IS 30 DEGREES.
 ALL NEG/ZERO DET(JACOBIAN) VALUE (DJ) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY FAIL.
 A MINIMUM OF 100 WARPED FACE COEFFICIENT TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.

TOLERANCE LIMITS ARE: HEX_AR = 300.00, HEX_EPLR = .40, HEX_DETJ = .01, HEX_WARP = .65 (xxxx = LIMIT VIOLATED)

ELEMENT TYPE	ID	LONGEST EDGE	SHORTEST EDGE	ASPECT RATIO	EDGE POINT LENGTH RATIO	MIN. JACOBIAN DETERMINANT	FACE WARP COEFFICIENT
HEXA	8602	4.00	1.00	4.00	N/A- ONLY 8 NODE	.92	.52 xxxx
HEXA	8620	4.00	1.00	4.00	.60xxxx	.50	1.00

*** USER INFORMATION MESSAGE 7555 (GMTSTD)
 FINITE ELEMENT GEOMETRY CHECK RESULTS EXCEED TOLERANCE LEVELS FOR THE FOLLOWING ELEMENTS.
 A MINIMUM OF 100 SKEW ANGLE (SA) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 MIN INT. ANGLE (IA) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 MAX INT. ANGLE (IA) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 WARPING FACTOR (WF) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 TAPER RATIO (TR) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.

TOLERANCE LIMITS ARE: SA = 30.00, IA(MIN) = 30.00, IA(MAX) = 150.00, WF = .05, TR = .50 (xxxx = LIMIT VIOLATED)

ELEMENT TYPE	ID	SKEW ANGLE	MIN INT. ANGLE	MAX INT. ANGLE	WARPING FACTOR	TAPER RATIO
QUAD4	105	29.74 xxxx	29.05 xxxx	150.95 xxxx	.00	.05
QUAD4	106	28.39 xxxx	27.76 xxxx	152.24 xxxx	.00	.05
QUAD4	107	27.15 xxxx	26.57 xxxx	153.43 xxxx	.00	.05
QUAD4	108	26.00 xxxx	25.46 xxxx	154.54 xxxx	.00	.05
QUAD4	109	24.94 xxxx	24.44 xxxx	155.56 xxxx	.00	.05
QUAD4	2105	29.74 xxxx	29.05 xxxx	150.95 xxxx	.00	.05
QUAD4	2106	28.39 xxxx	27.76 xxxx	152.24 xxxx	.00	.05
QUAD4	2107	27.15 xxxx	26.57 xxxx	153.43 xxxx	.00	.05
QUAD4	2108	26.00 xxxx	25.46 xxxx	154.54 xxxx	.00	.05
QUAD4	2109	24.94 xxxx	24.44 xxxx	155.56 xxxx	.00	.05

*** USER INFORMATION MESSAGE 7555 (GMTSTD)
 FINITE ELEMENT GEOMETRY CHECK RESULTS EXCEED TOLERANCE LEVELS FOR THE FOLLOWING ELEMENTS.
 A MINIMUM OF 100 SKEW ANGLE (SA) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 MIN INT. ANGLE (IA) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 MAX INT. ANGLE (IA) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 WARPING FACTOR (WF) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.
 A MINIMUM OF 100 TAPER RATIO (TR) TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY xxxx.

TOLERANCE LIMITS ARE: SA = 30.00, IA(MIN) = 30.00, IA(MAX) = 150.00, WF = .05, TR = .50 (xxxx = LIMIT VIOLATED)

ELEMENT TYPE	ID	SKEW ANGLE	MIN INT. ANGLE	MAX INT. ANGLE	WARPING FACTOR	TAPER RATIO
QUADR	1105	29.74 xxxx	29.05 xxxx	150.95 xxxx	.00	.05
QUADR	1106	28.39 xxxx	27.76 xxxx	152.24 xxxx	.00	.05
QUADR	1107	27.15 xxxx	26.57 xxxx	153.43 xxxx	.00	.05
QUADR	1108	26.00 xxxx	25.46 xxxx	154.54 xxxx	.00	.05
QUADR	1109	24.94 xxxx	24.44 xxxx	155.56 xxxx	.00	.05

*** USER INFORMATION MESSAGE 7555 (ETETGD)
FINITE ELEMENT GEOMETRY CHECK RESULTS EXCEED TOLERANCE LEVELS FOR THE FOLLOWING ELEMENTS.

A MINIMUM OF 4 EDGE LENGTH TO HEIGHT RATIO (AR)TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY WARN.

A MINIMUM OF 4 EDGE NODE POINT LENGTH RATIO (ER)TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY WARN.

EDGE ANGLE VALUES ARE INDICATED BY AN * AFTER THE VALUE. THE TOLERANCE LIMIT IS 30 DEGREES.

A MINIMUM OF 4 NEG/ZERO DET(JACOBIAN) VALUE (DJ)TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY WARN.

A MINIMUM OF 4 NEG DET(JAC) AT VERTEX NODE (DG)TOLERANCE LIMIT VIOLATIONS WILL BE IDENTIFIED AND INDICATED BY WARN.

TOLERANCE LIMITS ARE: TET_AR = 351.00, TET_EPLR = .45, TET_DETJ = -.01, TET_DETG = -.02 (xxxx = LIMIT VIOLATED)

ELEMENT TYPE	ID	LONGEST EDGE	SHORTEST HEIGHT	ASPECT RATIO	EDGE POINT LENGTH RATIO	MIN. JACOBIAN DETERMINANT	MIN. DET(JAC) AT VERTEX
TETRA	6601	1.41	.58	2.45	.25 WARN	.14	-.20 WARN

E L E M E N T G E O M E T R Y T E S T R E S U L T S S U M M A R Y

TOTAL NUMBER OF TIMES TOLERANCES WERE EXCEEDED

JACOBIAN DETERMINANT	ASPECT/	MINIMUM	MAXIMUM	SURFACE/FACE	EDGE POINT		
ELEMENT TYPE	SKEW ANGLE	TAPER RATIO	INTER. ANGLE	INTER. ANGLE	WARP FACTOR	OFFSET RATIO	LENGTH RATIO
BAR	N/A	N/A	N/A	N/A	N/A	0	N/A
BEAM	N/A	N/A	N/A	N/A	N/A	0	N/A
HEXA	N/A	0	N/A	N/A	N/A	1	N/A
PENTA	N/A	0	N/A	N/A	N/A	0	N/A
QUAD4	10	0	10	10	0	N/A	N/A
QUADR	5	0	5	5	0	N/A	N/A
TETRA	N/A	0	N/A	N/A	N/A	0	N/A
TRIA3	0	N/A	N/A	0	N/A	N/A	N/A
TRIAR	0	N/A	N/A	0	N/A	N/A	N/A

N/A IN THE ABOVE TABLE INDICATES TESTS THAT ARE NOT APPLICABLE TO THE ELEMENT TYPE AND WERE NOT PERFORMED. FOR ALL ELEMENTS WHERE GEOMETRY TEST RESULTS HAVE EXCEEDED TOLERANCES,

HEXA	ELEMENT ID	8620	PRODUCED	SMALLEST	EDGE	COS(ANGLE)	OF	.60	(TOLERANCE =	.87).
HEXA	ELEMENT ID	8602	PRODUCED	SMALLEST	FACE	WARP FACTOR	OF	.52	(TOLERANCE =	.65).
PENTA	ELEMENT ID	3915	PRODUCED	SMALLEST	EDGE	COS(ANGLE)	OF	.78	(TOLERANCE =	.87).
QUAD4	ELEMENT ID	109	PRODUCED	SMALLEST	SKEW ANGLE		OF	24.94	(TOLERANCE =	30.00).
QUAD4	ELEMENT ID	109	PRODUCED	SMALLEST	INTERIOR ANGLE		OF	24.44	(TOLERANCE =	30.00).
QUAD4	ELEMENT ID	109	PRODUCED	LARGEST	INTERIOR ANGLE		OF	155.56	(TOLERANCE =	150.00).
QUADR	ELEMENT ID	1109	PRODUCED	SMALLEST	SKEW ANGLE		OF	24.94	(TOLERANCE =	30.00).
QUADR	ELEMENT ID	1109	PRODUCED	SMALLEST	INTERIOR ANGLE		OF	24.44	(TOLERANCE =	30.00).
QUADR	ELEMENT ID	1109	PRODUCED	LARGEST	INTERIOR ANGLE		OF	155.56	(TOLERANCE =	150.00).
TETRA	ELEMENT ID	6601	PRODUCED	LARGEST	EDGE POINT LR		OF	.25	(TOLERANCE =	.45).
TETRA	ELEMENT ID	6601	PRODUCED	SMALLEST	VERTEX DET(JAC)	-UWM 6828-	OF	-.20	(TOLERANCE =	-.20).

Introduction

A long-standing key feature of MSC.Nastran is the ability of the PSHELL(, PCOMP), PBAR(, PBARL), PBEAM(, PBEAML, PBCOMP), PROD(, CONROD), PBEND, PSHEAR, PTUBE, PCONEAX, and PRAC2D property entries to include nonstructural

mass. The five Bulk Data entries--NSM, NSM1, NSML, NSML1, and NSMADD--distribute nonstructural mass by element lists or specific property lists associated with the above fourteen property entries.

Theory

NSML and NSML1 Entries

The NSML and NSML1 entries compute a nonstructural mass coefficient value for “area” elements identified by an element list or property list or both by the relationship:

$$\text{NSM_value} = \frac{\text{Lumped_non_structural_mass_value}}{\sum_{\text{elements}} \text{AREA}}$$

The NSML and NSML1 entries compute a nonstructural mass coefficient value for “line” elements identified by an element list or property list or both by the relationship:

$$\text{NSM_value} = \frac{\text{Lumped_non_structural_mass_value}}{\sum_{\text{elements}} \text{LENGTH}}$$

In the above two expressions, AREA corresponds to the area of each individual area element (CQUAD4 for example) and LENGTH corresponds to the length of each individual length element (CBAR for example).

The NSML and NSML1 entries are then converted internally to NSM and NSM1 entries.

Inputs

Case Control Commands

NSM Case Control command allows for the selection of different NSM sets for residual and superelements.

Bulk Data Entries

NSM and its alternate form NSM1 allows the user to allocate an NSM_value to selected sets of elements.

NSML and its alternate form NSML1 allows the user to allocate and smear a lumped nonstructural mass value to selected sets of elements.

NSMADD allows the user to form combinations of NSM, NSM1, NSML, and NSML1 sets and sum their results to selected sets of elements.

Guidelines and Limitations

The following requirements are associated with the new capability:

1. The NSML and NSML1 entries cannot mix “area” and “line” elements on the same entry.
2. Area elements are: CQUAD4, CQUAD8, CQUADR, CTRIA3, CTRIA6, CTRIAR, CSHEAR, and CRAC2D. (The CCONEAX is a stand alone element.)
3. Line elements are: CBAR, CBEAM, CBEND, CTUBE, CROD, and CONROD.
4. The CCONEAX is excluded from the NSML and NSML1 entries.

Example

The following example partial input demonstrates the form of the new entries and the expected result:

Partial Input File

```

SOL 108 $
CEND
  DISPL = ALL
  .
  .
  .
  ELSUM = ALL
  NSM = 1222
BEGIN BULK
$
.
.
.
$
$      2      3      4      5      6      7      8      9      0
NSML   9      PBEAM  211    .026   311    .028   411    .030
NSML   9      PBEND  21     .4869  31     .4869
NSML  10      PRAC2D  1     .132
NSML1 10      PBCOMP  .126   21     THRU   41
NSML1 11      PBEAML  .138   212    312    412
NSML  11      PROD   12     .17
NSML  12      CONROD  12     .19
NSML  12      PSHELL  51     4.5
NSML1 13      PSHELL  132.   ALL
NSML1 13      PSHELL  107.8  50     THRU   52
NSML1 14      PSHELL  7.5    57

```

NSML1	14	ELEMENT	.594	20	30	40	201	202
	301	302	401	402				
NSML1	15	PSHEAR	.264	25	THRU	45	BY	10
NSML	15	PBAR	22	.22				
NSML1	15	PBARL	.52	ALL				
NSML	16	ELEMENT	101	.077				
NSML	16	ELEMENT	27	1.021	13	.67		
NSML	17	ELEMENT	14	.41				
NSML	17	PTUBE	14	.43				
NSML1	18	ELEMENT	16.2	41	81234			
NSML1	18	PSHEAR	1.476	ALL				
NSML	18	PSHEAR	1	1.476				
NSML1	19	ELEMENT	.112	201	402			
NSML1	19	ELEMENT	4.2	3123	6134	134		
\$								
NSMADD	1222	9	11	12	14	13	15	
	16	1718	19	2010	12			

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

\$

FINAL
DISTRIBUTED

PROPERTY
VALUE

ELM	NSM	ID	PID	PROPERTY	NSM
----	---	--	---	-----	----
CBAR	.061	25	22	- PBAR	.013
CBAR	.039	26	26	- PBARL	.013
CBEAM	.065	20	21	- PBCOMP	.011
	.065				
CBEAM	.065	30	31	- PBCOMP	.011
	.065				
CBEAM	.065	40	41	- PBCOMP	.011
	.065				
CBEAM	.085	201	211	- PBEAM	.011,.012,.014
	.086	is1			
	.088	is2	all others zero		
	.088	end B			
CBEAM	.058	301	311	- PBEAM	.011,.012,.014
	.059	is1			
	.061	is2	all others zero		
	.061	end B			
CBEAM	.059	401	411	- PBEAM	.011,.012,.014
	.060	is1			
	.062	is2	all others zero		
	.062	end B			
CBEAM	.0671	202	212	- PBEAML	.0111,.0112,.0113,.0114,.0115
	.0672	is1			
	.0673	is2			
	.0674	is3	all others zero		
	.0675	is4			
	.0675	end B			
CBEAM	.0671	302	312	- PBEAML	.0111,.0112,.0113,.0114,.0115
	.0672	is1			
	.0673	is2			
	.0674	is3	all others zero		
	.0675	is4			
	.0675	end B			


```

$ CBEAM      .0951  402    412    - PBEAML  .0111,.0112,.0113,.0114,.0115
$           .0952  is1
$           .0953  is2
$           .0954  is3 all others zero
$           .0955  is4
$           .0955  end B
$ CBEND      .109   27     21     - PBEND   .013
$ CBEND      .044   37     31     - PBEND   .013
$ CBEND      .013   47     41     - PBEND   .013
$ CONROD     .032   12     1      - CONROD  .013
$ CQUAD4     .257   41     57     - PCOMP   .013
$ CQUAD4     .178  41234   50     - PSHELL  .013
$ CQUAD4     .178  4123456 52     - PSHELL  .013
$ CQUAD4     .193  4123456751 - PSHELL  .013
$ CQUAD8     .178   81     50     - PSHELL  .013
$ CQUAD8     .259  81234   50     - PSHELL  .013
$ CQUAD8     .178  8123456 52     - PSHELL  .013
$ CQUAD8     .193  8123456751 - PSHELL  .013
$ CQUADR     .178   1      50     - PSHELL  .013
$ CQUADR     .178  1234    50     - PSHELL  .013
$ CQUADR     .178  123456 52     - PSHELL  .013
$ CQUADR     .193  1234567 51     - PSHELL  .013
$ CROD       .097   13     12     - PROD    .013
$ CSHEAR     .158   25     1      - PSHEAR  .013
$ CSHEAR     .158   35     1      - PSHEAR  .013
$ CSHEAR     .158   45     1      - PSHEAR  .013
$ CTRIA3     .206  3123    50     - PSHELL  .013
$ CTRIA3     .178  3134    50     - PSHELL  .013
$ CTRIA6     .178  6123    50     - PSHELL  .013
$ CTRIA6     .206  6134    50     - PSHELL  .013
$ CTRIAR     .178  123     50     - PSHELL  .013
$ CTRIAR     .206  134     50     - PSHELL  .013
$ CTUBE      .097   14     14     - PTUBE   .013
$
$
.
.
.
$
ENDDATA

```

Typical Element mass summary is given as:

3.19 ELSUM Case Control Command

3.19.1 Introduction

The ELSUM Case Control command controls the generation of a printed table of informational properties for the various element types present in the input data file. The information produced includes element measures (e.g. length, thickness, area, volume) and mass property data. The information is grouped according to the finite element type. Several new options have been added to the ELSUM command giving users more control over the amount of output produced and another view of the data grouped by element property.

3.19.2 Method and Theory

For each element present in the model, physical metrics appropriate to the element type are computed. These metrics may include length, thickness, area, and volume. Mass properties are computed for supported element types. For the CONM1 element, the mass is taken to be the average of the M11, M22 and M33 values on the CONM1 Bulk Data entry. In addition, references to material properties and element properties are gathered for each element. For bar, beam and shell elements, an attempt is made to determine whether the property reference is to a basic property entry, or has been derived from one of the available alternate input entries (PBARL, PBEAML, PBCOMP, PCOMP). If it is determined that the property is derived, a character (L for PBARL or PBEAML, P for PBCOMP or PCOMP) is appended to the property identification number on output. For the case of the element type grouping, each element type is summarized in turn. For each element type, information for each element is displayed. Sub-totals are generated for mass property information. For the case of the property type grouping, each property type referenced by an element type is summarized in turn. All of the property ids referenced within the type are summarized in ascending property id order. Within this summary, information for each element referencing the property id is displayed. Various subtotals are accumulated and displayed.

3.19.3 Inputs

The ELSUM Case Control command has been enhanced to give the user new controls over the amount and ordering of the element summary information produced. The general format of the command is:

$$\text{ELSUM}([\text{EID}, \text{PID}, \text{BOTH}, \text{PIDSUM}, \text{EIDSUM}]) = \left\{ \begin{array}{c} \text{ALL} \\ \mathbf{n} \\ \text{NONE} \end{array} \right\}$$

The complete description of the ELSUM command can be found in “[Statements, Commands, Entries, and Parameters](#)” on page 621. The EID and PID keywords are used to select the particular grouping desired for the element summary output. The EID keyword requests grouping by element type while the PID keyword requests grouping by element property type. The BOTH keyword requests both groupings. The PIDSUM keyword requests that only mass property totals be output for the PID grouping. The EIDSUM keyword requests that only mass property totals be output for the EID grouping.

3.19.4 Outputs

The ELSUM Case Control command causes printed output to be generated. [Figure 3-70](#) is an example of the EID grouping output. [Figure 3-71](#) is an example of the PID grouping output. If PIDSUM or EIDSUM were used, then only the subtotals and totals information would be present in the figures.

ELEMENT PROPERTY SUMMARY (BY ELEMENT TYPE / ID)									
ELEMENT TYPE = BAR									
ELEM ID	PROP ID	MATL ID	LENGTH	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS
3401	3401	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01
3403	3403	1	3.00000E+00	1.00000E+00	3.00000E+00	1.50000E-01	1.50000E+00	1.65000E+00	1.65000E+00
3410	3403	1	2.00000E+00	1.00000E+00	2.00000E+00	1.00000E-01	1.00000E+00	1.10000E+00	1.10000E+00
17101	17103 L	1	1.00000E+00	1.01034E+00	1.01034E+00	5.05168E-02	0.00000E+00	5.05168E-02	5.05168E-02
17103	17103 L	1	3.00000E+00	1.01034E+00	3.03101E+00	1.51550E-01	0.00000E+00	1.51550E-01	1.51550E-01
17110	17103 L	1	2.00000E+00	1.01034E+00	2.02067E+00	1.01034E-01	0.00000E+00	1.01034E-01	1.01034E-01
SUBTOTAL MASS FOR ALL BAR						6.03101E-01	3.00000E+00	3.60310E+00	3.60310E+00
.									
.									
.									
ELEMENT TYPE = BEAM									
ELEM ID	PROP ID	MATL ID	LENGTH	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS
200	200	1	1.00000E+00	1.00000E+02	1.00000E+02	5.00000E+00	0.00000E+00	5.00000E+00	5.00000E+00
17000	17000 L	1	1.00000E+00	9.99987E+01	9.99987E+01	4.99994E+00	0.00000E+00	4.99994E+00	4.99994E+00
SUBTOTAL MASS FOR ALL BEAM						9.99994E+00	0.00000E+00	9.99994E+00	9.99994E+00
.									
.									
.									
ELEMENT TYPE = PENTA									
ELEM ID	PROP ID	MATL ID			VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS
6801	6801	2			4.00000E-02	2.00000E-03	0.00000E+00	2.00000E-03	2.00000E-03
SUBTOTAL MASS FOR ALL PENTA						2.00000E-03	0.00000E+00	2.00000E-03	2.00000E-03
.									
.									
.									
ELEMENT TYPE = QUAD4									
ELEM ID	PROP ID	MATL ID	THICKNESS	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS
3301	3301	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01
3313	3313	1	6.12009E-01	6.36864E+00	3.89766E+00	1.94883E-01	1.91715E+00	2.11203E+00	2.11203E+00
3321	3321 P	3321	1.00000E+00	1.00000E+02	1.00000E+02	1.00000E+02	0.00000E+00	1.00000E+02	1.00000E+02
17503	17503	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	0.00000E+00	5.00000E-02	5.00000E-02
SUBTOTAL MASS FOR ALL QUAD4						1.00295E+02	2.41715E+00	1.02712E+02	1.02712E+02
.									
.									
.									
ELEMENT TYPE = ROD									
ELEM ID	PROP ID	MATL ID	LENGTH	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS
101	101	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01
107	101	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01
SUBTOTAL MASS FOR ALL ROD						1.00000E-01	1.00000E+00	1.10000E+00	1.10000E+00
.									
.									
.									
ELEMENT TYPE = TETRA									
ELEM ID	PROP ID	MATL ID			VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS
3901	3901	1			5.77350E-01	2.88675E-02	0.00000E+00	2.88675E-02	2.88675E-02
SUBTOTAL MASS FOR ALL TETRA						2.88675E-02	0.00000E+00	2.88675E-02	2.88675E-02
.									
.									
.									
TOTAL MASS FOR ALL SUPPORTED ELEMENT TYPES						1.38647E+02	1.53344E+01	1.53981E+02	1.53981E+02

Figure 3-70 Sample Element Summary by Element Type Output

ELEMENT PROPERTY SUMMARY (BY PROPERTY TYPE / ID)										
PROPERTY TYPE = PBAR, ID = 3401 *****										
ELEM ID	ELEM TYPE	MATL ID	LENGTH	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS	
3401	BAR	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01	
SUBTOTAL MASS FOR ALL BAR			ELEMENTS	FOR PBAR, ID = 3401		5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01	
.
PROPERTY TYPE = PBEAM, ID = 200 *****										
ELEM ID	ELEM TYPE	MATL ID	LENGTH	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS	
200	BEAM	1	1.00000E+00	1.00000E+02	1.00000E+02	5.00000E+00	0.00000E+00	5.00000E+00	5.00000E+00	
SUBTOTAL MASS FOR ALL BEAM			ELEMENTS	FOR PBEAM, ID = 200		5.00000E+00	0.00000E+00	5.00000E+00	5.00000E+00	
.
PROPERTY TYPE = PSOLID, ID = 3901 *****										
ELEM ID	ELEM TYPE	MATL ID			VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS	
3901	TETRA	1			5.77350E-01	2.88675E-02	0.00000E+00	2.88675E-02	2.88675E-02	
SUBTOTAL MASS FOR ALL SOLID			ELEMENTS	FOR PSOLID, ID = 3901		2.88675E-02	0.00000E+00	2.88675E-02	2.88675E-02	
.
PROPERTY TYPE = (NONE) *****										
ELEM ID	ELEM TYPE	MATL ID				STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS	
2901	CONM1	0				1.00000E+00	0.00000E+00	1.00000E+00	1.00000E+00	
2904	CONM1	0				2.20000E+00	0.00000E+00	2.20000E+00	2.20000E+00	
SUBTOTAL MASS FOR ALL CONM1			ELEMENTS			3.20000E+00	0.00000E+00	3.20000E+00	3.20000E+00	
.
PROPERTY TYPE = PSHELL, ID = 3301 *****										
ELEM ID	ELEM TYPE	MATL ID	THICKNESS	AREA	VOLUME	STRUCT.MASS	NON-STR.MASS	TOTAL MASS	TM*WTMASS	
3301	QUAD4	1	1.00000E+00	1.00000E+00	1.00000E+00	5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01	
SUBTOTAL MASS FOR ALL SHELL			ELEMENTS	FOR PSHELL, ID = 3301		5.00000E-02	5.00000E-01	5.50000E-01	5.50000E-01	
.
TOTAL MASS FOR ALL SUPPORTED ELEMENT TYPES						1.38647E+02	1.53344E+01	1.53981E+02	1.53981E+02	
						=====	=====	=====	=====	

Figure 3-71 Sample Element Summary by Element Property Type Output

3.19.5 Guidelines and Limitations

- The ELSUM command keywords can be used in any combination. However, if both EID and EIDSUM are present, EIDSUM takes precedence over EID and only the mass property totals summary will be generated for the grouping by element type. This is also true for the PID and PIDSUM keywords. PIDSUM takes precedence over PID and only the mass property totals will be generated for the grouping by element property type.
- Only the following element types produce mass property information: CBAR, CBEAM, CBEND, CHEXA, CMASSi, CONM1, CONM2, CONROD, CPENTA, CQUAD4, CQUAD8, CQUADR, CRAC2D, CRAC3D, CROD, CSHEAR, CTETRA, CTRIA3, CTRIA6, CTRIAR, CTRIAX6 and CTUBE.
- The ELSUM command is ignored in heat transfer solution sequences.

3.19.6 Example

No new example is presented for this enhanced ELSUM Case Control command since any existing input data file available can be used to demonstrate the new features. One simply inserts an ELSUM command with the desired keywords into the Case Control Section of the input file. Element summary information will be generated and placed on the standard .f06 output file.

CHAPTER

4

Material Properties

- Introduction
- Isotropic Material
- Orthotropic Material
- Anisotropic Material
- Nonuniform Material and Composite Properties for Shells
- Hyperelastic Material

4.1 Introduction

The material properties used in structural analysis are described in this section. Material properties for heat transfer are described in the *MSC.Nastran Thermal Analysis User's Guide* and the material properties for nonlinear and fluid structure interaction are described in “ ” on page 568 and “**Coupled Fluid-Structure Interaction**” on page 753, respectively. The material definitions describe the stiffness or constitutive relationships, strength, density, and thermal expansion characteristics of the material. Material Bulk Data entries can be classified by type as isotropic (MAT1), orthotropic (MAT3, MAT8), or anisotropic (MAT2, MAT9).

The materials input data types MAT1, MAT2, MAT8 and MAT9 entries can be used to describe temperature-dependent material when used in conjunction with MATT1, MATT2, MATT8 and MATT9 entries, respectively. In addition, the MATS1 can be used with the MAT1entry to describe nonlinear material behavior.

Also included in this section is a description of composite materials (PBCOMP, PCOMP) which are used in the formulation of equivalent material constitutive relations. The method used to model composites is also used in the analysis of materials with nonhomogeneous cross sections. Composite material properties are used in the analysis of the composite (nonuniform) beams and shells.

Table 4-1 summarizes the relationships between elements and material input data types.

Table 4-1 Correspondence Between Element and Material Type

Element Type	Material Type					Hyper- elastic MATHP	Stress Dependent MATS1	Temperature Dependent MATTi	Composite
	MAT1	MAT2	MAT3	MAT8	MAT9				
CBAR	X							X	
CBEAM	X						X	X	X
CBEND	X							X	
CCONEAX	X	X						X	
CRAC2D	X							X	
CRAC3D	X							X	
CHEXA	X				X	X	X	X	
CPENTA	X				X	X	X	X	
CQUAD						X			
CQUAD4	X	X		X		X	X	X	X

Table 4-1 Correspondence Between Element and Material Type (continued)

Element	Material Type					Hyper-elastic	Stress Dependent	Temperature Dependent	Composite
	MAT1	MAT2	MAT3	MAT8	MAT9				
CQUAD8	X	X		X		X		X	X
CQUADR	X	X		X				X	
CQUADX						X			
CROD	X						X	X	
CSHEAR	X							X	
CTETRA	X				X	X	X	X	
CTRIA3	X	X		X		X	X	X	X
CTRIA6	X	X		X		X		X	X
CTRIAR	X	X		X				X	
CTRIAX						X			
CTRIAX6	X		X						
CTUBE	X						X	X	

4.2 Isotropic Material

The MAT1 entry is used to describe the isotropic material properties for general, plain stress, plain strain, and axisymmetric isotropic material behavior. This section describes the MAT1 Bulk Data entries used to define isotropic materials and how that definition relates to the constitutive relations. See [Table 4-1](#) for a summary of elements that use MAT1 entries.

The general three-dimensional isotropic material stress-strain constitutive relationship is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} = \begin{bmatrix} \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & \frac{\nu E}{(1+\nu)(1-2\nu)} & \frac{\nu E}{(1+\nu)(1-2\nu)} & 0 & 0 & 0 \\ & \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & \frac{\nu E}{(1+\nu)(1-2\nu)} & 0 & 0 & 0 \\ & & \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & 0 & 0 & 0 \\ & & & \frac{E}{2(1+\nu)} & 0 & 0 \\ & & & & \frac{E}{2(1+\nu)} & 0 \\ & & & & & \frac{E}{2(1+\nu)} \end{bmatrix} \quad \text{Eq. 4-1}$$

symmetric

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} = (T - T_{ref}) \begin{Bmatrix} A \\ A \\ A \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$

where $\{\sigma\}$, $\{\varepsilon\}$, and $\{A\}$ are the stress, strain, and thermal expansions coefficients, respectively. E , ν , G , and $(T - T_{ref})$ are Young's modulus, Poisson's ratio, shear modulus, and the change in element temperature used to calculate initial element thermal expansions.

For MAT1, $G = E/2(1+\nu)$ is the relationship used to determine the isotropic elastic material constants when only two of the three are input on the MAT1 entry. When all three are given, G will not be required to satisfy this relationship (see [Table 4-2](#) for the use of the elastic constants with respect to the finite elements usage).

Isotropic material constitutive relationship for plane stress behavior is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{E}{1-\nu^2} & \frac{\nu E}{1-\nu^2} & 0 \\ \frac{\nu E}{1-\nu^2} & \frac{E}{1-\nu^2} & 0 \\ 0 & 0 & \frac{E}{2(1+\nu)} \end{bmatrix} \left\{ \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} - (T - T_{ref}) \begin{Bmatrix} A \\ A \\ 0 \end{Bmatrix} \right\} \quad \text{Eq. 4-2}$$

Isotropic material constitutive relationship for plane strain behavior is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & \frac{\nu E}{(1+\nu)(1-2\nu)} & 0 \\ \frac{\nu E}{(1+\nu)(1-2\nu)} & \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & 0 \\ 0 & 0 & \frac{E}{2(1+\nu)} \end{bmatrix} \left\{ \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} - (T - T_{ref})(1+\nu) \begin{Bmatrix} A \\ A \\ 0 \end{Bmatrix} \right\} \quad \text{Eq. 4-3}$$

When MAT1 is used for transverse shear

$$\begin{pmatrix} \sigma_{zx} \\ \sigma_{zy} \end{pmatrix} = \begin{bmatrix} G & 0 \\ 0 & G \end{bmatrix} \begin{pmatrix} \gamma_{zx} \\ \gamma_{zy} \end{pmatrix} \quad \text{Eq. 4-4}$$

Table 4-2 MAT1 – Material Property Usage Versus Element Types

Element Entry	E	NU	G
CROD CBEAM CBAR	Extension and Bending	Not Used	Torsion Transverse Shear
CQUADi CTRIAi CCONEAX	Membrane and Bending		Transverse Shear
CSHEAR	Not Used		Shear
CRAC2D	All Terms		Not Used

Table 4-2 MAT1 – Material Property Usage Versus Element Types

Element Entry	E	NU	G
CHEXA CPENTA CTETRA CRAC3D	All Terms		Not Used
CTRIAX6	Radial, Axial, Circumferential	All Coupled Ratios	Shear

4.3 Orthotropic Material

This section describes the MAT3 and MAT8 Bulk Data entries which define orthotropic materials and their corresponding constitutive relations. The MAT3 and MAT8 entries are used to describe the orthotropic material properties for two-dimensional and three-dimensional element orthotropic material behavior. See [Table 4-1](#) for a summary of elements that use MAT3 and MAT8 entries.

The MAT3 entry is used to define orthotropic material matrix for axisymmetric solid elements in the stress-strain relationship as given by:

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_\theta \\ \varepsilon_z \\ \gamma_{zx} \end{Bmatrix} = \begin{bmatrix} \frac{1}{E_x} & -\frac{\nu_{\theta x}}{E_\theta} & -\frac{\nu_{zx}}{E_z} & 0 \\ -\frac{\nu_{x\theta}}{E_x} & \frac{1}{E_\theta} & -\frac{\nu_{z\theta}}{E_z} & 0 \\ -\frac{\nu_{xz}}{E_x} & -\frac{\nu_{\theta z}}{E_\theta} & \frac{1}{E_z} & 0 \\ 0 & 0 & 0 & \frac{1}{G_{zx}} \end{bmatrix} \begin{Bmatrix} \sigma_x \\ \sigma_\theta \\ \sigma_z \\ \tau_{zx} \end{Bmatrix} + (T - T_{ref}) \cdot \begin{Bmatrix} A_x \\ A_\theta \\ A_z \\ 0 \end{Bmatrix} \quad \text{Eq. 4-5}$$

where $\{\sigma\}$, $\{\varepsilon\}$, and $\{A\}$ are the stress, strain, and thermal expansions coefficients, respectively. E , ν , G , and $(T - T_{ref})$ are Young's modulus, Poisson's ratio, shear modulus, and the change in element temperature used to calculate initial element thermal expansions.

Where $\nu_{x\theta}/E_x = \nu_{\theta z}/E_\theta$, $\nu_{zx}/E_z = \nu_{xz}/E_x$, and $\nu_{\theta z}/E_\theta = \nu_{z\theta}/E_z$ must be satisfied to preserve symmetry. Only the $\nu_{\theta x}$, ν_{xz} , and $\nu_{z\theta}$ terms are used in the input.

The MAT8 entry is used to define orthotropic material matrix for shell elements in the stress-strain relationship as given by:

$$\begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \\ \gamma_{1z} \\ \gamma_{2z} \end{Bmatrix} = \begin{bmatrix} \frac{1}{E_1} & \frac{-\nu_{21}}{E_2} & 0 & 0 & 0 \\ \frac{-\nu_{12}}{E_1} & \frac{1}{E_2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{G_{12}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{1z}} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{2z}} \end{bmatrix} \begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \\ \tau_{1z} \\ \tau_{2z} \end{Bmatrix} + (T - T_{ref}) \begin{Bmatrix} A_1 \\ A_2 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad \text{Eq. 4-6}$$

where $\{\sigma\}$, $\{\varepsilon\}$, and $\{A\}$ are the stress, strain, and thermal expansions coefficients, respectively. E , ν , G , and $(T - T_{ref})$ are Young's modulus, Poisson's ratio, shear modulus, and the change in element temperature used to calculate initial element thermal expansions.

Where $\nu_{21}/E_2 = \nu_{12}/E_1$ must be satisfied to preserve symmetry. Only the ν_{12} term is used in the input.

When either G_{1z} or G_{2z} is input as zero, it implies that the corresponding shear modulus is infinite. When this is the case, the transverse shear flexibility effects are neglected in the shell formulation.

4.4 Anisotropic Material

This section describes the MAT2 and MAT9 entries that define anisotropic materials and their corresponding constitutive relations. The MAT2 and MAT9 entries are used to describe the anisotropic material properties for two-dimensional and three-dimensional element anisotropic material behavior. See [Table 4-1](#) for a summary of elements that use MAT2 and MAT9 entries.

The MAT2 entry used to define two-dimensional anisotropic constitutive stress-strain relationship using the anisotropic material matrix, $[G]$. The stress-strain relationship is given by:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{12} & G_{22} & G_{23} \\ G_{13} & G_{23} & G_{33} \end{bmatrix} \left\{ \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix} - (T - T_{ref}) \begin{Bmatrix} A_1 \\ A_2 \\ A_{12} \end{Bmatrix} \right\} \quad \text{Eq. 4-7}$$

where $\{\sigma\}$, $\{\varepsilon\}$, and $\{A\}$ are the stress, strain, and thermal expansions coefficients, respectively. E , ν , G , and $(T - T_{ref})$ are Young's modulus, Poisson's ratio, shear modulus, and the change in element temperature used to calculate initial element thermal expansions.

The MAT2 entry is also used to define two-dimensional orthotropic constitutive stress-strain relation using the anisotropic material matrix, $[G]$. The stress-strain relationship is given by:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} & 0 \\ G_{12} & G_{22} & 0 \\ 0 & 0 & G_{33} \end{bmatrix} \left\{ \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix} - (T - T_{ref}) \begin{Bmatrix} A_1 \\ A_2 \\ 0 \end{Bmatrix} \right\} \quad \text{Eq. 4-8}$$

where the elastic coefficients are given by engineering constants used to describe orthotropic material properties as:

$$G_{11} = \frac{E_1}{(1 - \nu_{12}\nu_{21})} \quad \text{Eq. 4-9}$$

$$G_{22} = \frac{E_2}{(1 - \nu_{12}\nu_{21})} \quad \text{Eq. 4-10}$$

$$G_{12} = \frac{\nu_{12}E_2}{(1 - \nu_{12}\nu_{21})} = \frac{\nu_{21}E_1}{(1 - \nu_{12}\nu_{21})} \quad \text{Eq. 4-11}$$

When MAT2 is used for transverse shear

$$\begin{pmatrix} \sigma_{zx} \\ \sigma_{zy} \end{pmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{12} & G_{22} \end{bmatrix} \begin{pmatrix} \gamma_{zx} \\ \gamma_{zy} \end{pmatrix} \quad \text{Eq. 4-12}$$

G_{33} must be set to 0.

The MAT9 entry is used to define general three-dimensional anisotropic constitutive stress-strain relationship using the anisotropic material matrix, $[G]$, the stress-strain constitutive relationship is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} & G_{14} & G_{15} & G_{16} \\ & G_{22} & G_{23} & G_{24} & G_{25} & G_{26} \\ & & G_{33} & G_{34} & G_{35} & G_{36} \\ & & & G_{44} & G_{45} & G_{46} \\ \text{symmetric} & & & & G_{55} & G_{56} \\ & & & & & G_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} - (T - T_{ref}) \begin{Bmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \end{Bmatrix} \quad \text{Eq. 4-13}$$

where $\{\sigma\}$, ε , and $\{A\}$ are the stress, strain and thermal expansions coefficients, respectively, and E , ν , G , and $(T - T_{ref})$ are Young's modulus, Poisson's ratio, shear modulus, and the change in element temperature used to calculate initial element thermal expansions.

The MAT9 entry is also used to define three-dimensional orthotropic constitutive stress-strain relation using the anisotropic material matrix $[G]$. The stress-strain relationship is given by:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} & 0 & 0 & 0 \\ & G_{22} & G_{23} & 0 & 0 & 0 \\ & & G_{33} & 0 & 0 & 0 \\ & & & G_{44} & 0 & 0 \\ \text{symmetric} & & & & G_{55} & 0 \\ & & & & & G_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} - (T - T_{ref}) \begin{Bmatrix} A_1 \\ A_2 \\ A_3 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad \text{Eq. 4-14}$$

4.5 Nonuniform Material and Composite Properties for Shells

See “[Composites](#)” on page 531.

4.6 Hyperelastic Material

This section describes the MATHP Bulk Data entry, which defines hyperelastic analysis of rubber-like material.

The generalized Mooney-Rivlin strain energy function may be expressed as follows:

$$U(J, \bar{I}_1, \bar{I}_2) = \sum_{i+j=1}^{NA} A_{ij} (\bar{I}_1 - 3)^i (\bar{I}_2 - 3)^j + \sum_{i=1}^{ND} D_i (J - 1 - AV(T - T_0))^{2i}$$

$$A_{00} = 0$$

where \bar{I}_1 and \bar{I}_2 are the first and second distortional strain invariants, respectively; $J = \det F$ is the determinant of the deformation gradient; and $2D_1 = K$ and $2(A_{10} + A_{01}) = G$ at small strains, in which K is the bulk modulus and G is the shear modulus. The model reduces to a Mooney-Rivlin material $NA = 1$ and to a Neo-Hookean material if $NA = 1$ and $A_{01} = 0.0$. For Neo-Hookean or Mooney-Rivlin materials no continuation entry is required. T is the current temperature and T_0 is the initial temperature.

Conventional Mooney-Rivlin and Neo-Hookean materials are fully incompressible. Full incompressibility is not presently available but may be simulated with a large enough value of D_1 . A value of D_1 higher than $10^3 \cdot (A_{10} + A_{01})$ is, however, not recommended.

A_{ij} (material constants related to distortional deformation) and D_i (material constants related to volumetric deformation) are obtained from least squares fitting of experimental data. One or more of four experiments (TAB1 to TAB4) may be used to obtain A_{ij} . D_i may be obtained from pure volumetric compression data (TABD). If all TAB1 through TAB4 are blank, A_{ij} must be specified by the user. Parameter estimation, specified through any of the TABLES1 entries, supersedes the manual input of the parameters.

If $ND = 10$ and a nonzero value of D_1 is provided or is obtained from experimental data in TABD, then the parameter estimation of the material constants A_{ij} takes compressibility into account in the cases of simple tension/compression, equibiaxial tension, and general biaxial deformation. Otherwise, full incompressibility is assumed in estimating the material constants.

CHAPTER

5

Applied Loads

- Static Loads
- Point Loads
- Distributed Loads
- Inertia Loads
- Thermal and Enforced Element Deformation Loads
- Loads Due to Enforced Motion
- Frequency-Dependent Loads
- Time-Dependent Loads
- Simplified Static Loading Data in Dynamic Analysis
- Combining of Loads
- p-Element Loads and Constraints
- Summary of Nonlinear Static Loads
- Summary of Nonlinear Transient Loads

5.1 Static Loads

In MSC.Nastran, static loads are applied to geometric and scalar points in a variety of ways, including

- Loads applied directly to grid points.
- Pressure loads on surfaces.
- Distributed and concentrated loads on CBAR, CBEAM, and CBEND elements.
- Gravity loads.
- Centrifugal loads due to steady rotation.
- Tangential loads due to angular acceleration.
- Loads resulting from thermal expansion.
- Loads resulting from enforced deformations of a structural element.
- Loads resulting from enforced displacements at a grid point.

Additional information on static loads is given in Section 3.6 of *The NASTRAN Theoretical Manual*. Any number of load sets can be defined in the Bulk Data Section. However, only those sets selected in the Case Control Section, as described in “[Case Control Section](#)” on page 11, will be used in the problem solution. The manner of selecting each type of load is specified on the associated Bulk Data entry description in the “[Bulk Data Entries](#)” on page 849 of the *MSC.Nastran Quick Reference Guide*.

5.2 Point Loads

The FORCE entry is used to define a static load applied to a geometric grid point in terms of components defined by a local coordinate system. The orientation of the load components depends on the type of local coordinate system used to define the load. The directions of the load components are the same as those indicated on **Figure 2-2** of “**Grid Point and Coordinate System Definition**” on page 41 for displacement components. The FORCE1 entry is used if the direction is determined by a vector connecting two grid points, and a FORCE2 entry is used if the direction is specified by the cross product of two such vectors. The MOMENT, MOMENT1 and MOMENT2 entries are used in a similar fashion to define the application of a concentrated moment at a geometric grid point. The SLOAD entry is used to define a load at a scalar point. In this case, only the magnitude is specified since only one component of motion exists at a scalar point.

The FORCEAX and MOMAX entries are used to define the loading of specified harmonics on rings of conical shell elements. FORCE and MOMENT entries may be used to apply concentrated loads or moments to conical shell elements, providing that such points have been defined with a POINTAX entry.

5.3 Distributed Loads

The PLOAD1 entry is used to define either distributed or concentrated loads on CBAR and CBEAM elements. This entry is only used to define loads at intermediate points on CBAR and CBEAM elements. Applied loads at end points are defined with either FORCE or MOMENT entries. The CBARAO entry is used to define intermediate points on CBAR elements for which stress and/or force output is desired. The PLOAD1 entry is also used to define only distributed loads for the CBEND element. The distributed load is always applied along the entire length of the element and does not define loads at intermediate points on the CBEND. The PLOAD1 entry cannot be used to define concentrated loads at intermediate stations on the CBEND element.

Pressure loads on three-node triangular and four-node quadrilateral elements are defined with a PLOAD2 entry. The positive direction of the loading is determined by the order of the grid points on the element connection entry, using the right hand rule. The magnitude and direction of the load is automatically computed from the value of the pressure and the coordinates of the connected grid points. The load is applied to the connected grid points. The PLOAD entry is used in a similar fashion to define the loading of any three or four grid points regardless of whether they are connected with two-dimensional elements. The PLOAD4 entry may be used with CQUAD4, CQUAD8, CQUADR, CTRIA3, CTRIA6, and CTRIAR elements. Since the surface hyperelastic elements CQUAD4, CQUAD8, CQUAD, CTRIA3, and CTRIA6 are plane strain elements, no pressure load may be applied on these elements.

The PRESAX entry is used to define pressure loading on the conical shell element. The PLOADX1 entry is used to define pressure loading on the CTRIAX6 element and on the hyperelastic CTRIAX and CQUADX elements.

Pressure loads on the CHEXA, CPENTA, and CTETRA elements are defined with the PLOAD4 entry. The magnitude and direction of the equivalent grid point forces are automatically computed using the isoparametric shape functions of the element to which the load has been applied.

5.4 Inertia Loads

The GRAV entry is used to specify a gravity load (or any other type of acceleration) by providing the components of the gravity vector in any defined coordinate system. The gravity load is obtained from the gravity vector and the mass matrix. The gravitational acceleration is not calculated at scalar points. The user is required to introduce gravity loads at scalar points directly.

The RFORCE entry is used to define a static loading condition due to angular velocity and/or acceleration. These loads are specified by the designation of a grid point that lies on the axis of rotation and by the components of rotational velocity or angular acceleration in any defined coordinate system. There are two methods to compute these loads. In the first method, the mass matrix is regarded as pertaining to a set of distinct rigid bodies connected to grid points. Deviations from this viewpoint, such as the use of scalar points or the use of mass coupling between grid points, can result in errors. The second method allows mass coupling, but does not allow offsets. The second method is the only method used for angular acceleration.

5.5 Thermal and Enforced Element Deformation Loads

Temperatures may be specified for selected elements. The temperatures for CROD, CBAR, CBEAM, CBEND, CONROD, or TUBE elements are specified on the TEMPRB Bulk Data entry. This entry specifies the average temperature on both ends and, in the case of the CBAR, CBEAM and CBEND elements, is used to define temperature gradients over the cross section. Temperatures for two-dimensional plate and membrane elements are specified on a TEMPP1, TEMPP2, or TEMPP3 Bulk Data entry. The user-defined average temperature over the volume is used to produce in-plane loads and stresses. Thermal gradients over the depth of the bending elements, or the resulting moments, may be used to produce bending loads and stresses.

If no thermal element data is given for an element, the temperatures of the connected grid points given on the TEMP, TEMPD, or TEMPAX entries are simply averaged to produce an average temperature for the element. The thermal expansion coefficients are defined on the material definition entries. Regardless of the type of thermal data, if the material coefficients for an element are temperature dependent by use of the MATTi entry, they are always calculated from the “average” temperature of the element. The presence of a thermal field does not imply the application of a thermal load. A thermal load will not be applied unless the user makes a specific request in the Case Control Section.

Enforced axial deformations can be applied to CROD, CBAR, and CBEAM elements. Enforced deformations, for example, are useful in the simulation of misfit or misalignment in engineering structures. As in the case of thermal expansion, the equivalent loads are calculated by separate subroutines for each type of structural element and are applied to the connected grid points. The magnitude of the axial deformation is specified on a DEFORM entry.

Temperature Loads

If temperature loads are present in one or more of the subcases, the thermal strain must be taken into account in the calculation of element stresses and forces. That is, a body subjected to a temperature field will deform in a stress free manner in the absence of constraints. However, if the deformation is resisted through boundary conditions and/or the presence of applied loads, the total actual strain, at a point must be reduced by the thermal strain, $\alpha(T - T_o)$, in order to obtain the stress at the point. For example, the stress-strain relation for a rod that undergoes thermal expansion and also sustains an axial force is written in the form

$$\varepsilon - \alpha(T - T_o) = \frac{\sigma}{E} \quad \text{Eq. 5-1}$$

Element stresses and forces are calculated on an element-by-element basis from a knowledge of the displacement vector and the temperature field; therefore, in a SUBCOM or a SYMCOM, the user must supply a definition of the temperature field whenever element stresses and forces are required.

Consider the earlier example, with the modification that a temperature load is specified in SUBCASE 2.

```

      :
SUBCASE 1
  LOAD=2
SUBCASE 2
  TEMP ( LOAD ) =4
SUBCASE 10
  LOAD=10
SUBCOM 20
  SUBSEQ= .5 , 1 . , 1 . 6
  TEMP ( LOAD ) =4
      :

```

In this example, the multiplier for SUBCASE 2 is unity so that the temperature distribution required to calculate element forces and stresses for the combined loading case is precisely TEMP(LOAD)=4. If, on the other hand, the SUBSEQ entry is modified as indicated below, an additional TEMP(LOAD) must be defined.

```

      :
SUBCASE 1
  LOAD=1
SUBCASE 2
  TEMP ( LOAD ) =4
SUBCASE 10
  LOAD=10
SUBCOM 20
  SUBSEQ= .5 , .75 , 1 . 6
  TEMP ( LOAD ) =40

```

Here, TEMP(LOAD)=40 must reflect a temperature distribution of the form

$$(T_{40} - T_o) = .75(T_4 - T_o) \quad \text{Eq. 5-2}$$

$$T_{40} = T_o + .75(T_4 - T_o) \quad \text{Eq. 5-3}$$

This new TEMP(LOAD) must be supplied by the user through standard Bulk Data entries such as TEMPD, TEMP, TEMPP1, and TEMPP3.

In general, with n temperature loading conditions in the subcases immediately preceding a SUBCOM, the temperature distribution that must be defined under the SUBCOM if element force or stress output is desired is of the form

$$T_{SUBCOM} = T_o + \sum_{i=1}^n a_i(T_i - T_o) \quad \text{Eq. 5-4}$$

where the coefficients a_i are defined on the SUBSEQ entry under the SUBCOM in question. The reference temperature T_o , defined on MATi Bulk Data entries, can conveniently be defined as zero in order to simplify the calculation of the temperature distribution for the combined loading condition of the SUBCOM.

The same approach is used with combined loading condition defined through a SYMCOM Case Control entry. However, the use of SYM subcases and SYMCOM entries implies the use of the concepts of reflective symmetry, in which case it is strongly recommended that the DIH option of cyclic symmetry be used (“**Cyclic Symmetry**” on page 803). The appropriate boundary conditions are generated internally in MSC.Nastran, and the user simply defines the actual temperature distribution on each portion of the structure without regard to decomposing the temperatures and/or mechanical loads into symmetric and antisymmetric components.

Element Deformation

The DEFORM Bulk Data entry enables the user to define enforced axial deformations for the CBAR, CBEAM, CROD, and CTUBE elements. The nature of this enforced deformation is analogous to a turnbuckle that may be adjusted to provide a desired axial deformation in an element. The problem is analogous to the temperature problem discussed in the previous section and, for example, the appropriate stress-strain relation for a rod element of length that undergoes an enforced axial deformation, δ is given by the following equation:

$$\varepsilon - \frac{\delta}{l} = \frac{\sigma}{E} \quad \text{Eq. 5-5}$$

Consider the final example of the previous section.

```
      ⋮  
SUBCASE 1  
  LOAD=1  
SUBCASE 2  
  DEFORM=5  
SUBCASE 10  
  LOAD=10  
SUBCOM 20  
  
SUBSEQ=.5, .75, 1.6  
  DEFORM=50  
      ⋮
```

Here DEFORM = 50 must reflect a set of enforced deformations of the form

$$\delta_{50} = .75\delta_5 \quad \text{Eq. 5-6}$$

For n DEFORM loading conditions in the subcases that immediately precede a SUBCOM, the DEFORM loading condition that must be defined under the SUBCOM, if element force and stress output is desired, is of the form

$$\delta_{SUBCOM} = \sum_{i=1}^n a_i \delta_i \quad \text{Eq. 5-7}$$

5.6 Loads Due to Enforced Motion

Statics

Enforced zero displacements may be specified on GRID, SPC, or SPC1 entries. Zero displacements that result in nonzero forces of constraint are usually specified on SPC or SPC1 entries. If GRID entries are used, the constraints become part of the structural model and modifications cannot be made at the subcase level.

Nonzero enforced displacements may be specified on SPC or SPCD entries. The SPC entry specifies both the component to be constrained and the magnitude of the enforced displacement. The SPCD entry specifies only the magnitude of the enforced displacement. When an SPCD entry is used, the component to be constrained must be specified on either an SPC or SPC1 entry. The use of the SPCD entry avoids the decomposition of the stiffness matrix when changes are only made in the magnitudes of the enforced displacements.

The equivalent loads resulting from enforced displacements of grid points are calculated by the program and added to the other applied loads. The magnitudes of the enforced displacements are specified on SPC entries (SPCAX in the case of conical shell problems) in the global coordinate system. The application of the load is automatic when the user selects the associated SPC set in the Case Control Section.

The SPCD is selected by a load request in the Case Control Section.

Dynamics

Enforced motion in prior versions of MSC.Nastran used a nonzero value in the TYPE field (field 5) on the TLOAD1 and TLOAD2 entries to specify whether an enforced displacement, velocity, or acceleration were to be applied using the large mass method. Although these integer values are still allowed, the field has now been generalized to permit character input, and a similar TYPE field has been included in field 8 of the RLOAD1 and RLOAD2 entries to support enforced motion in frequency response analysis. The new TYPE field for these dynamic loading/enforced motion entries now reads as follows:

Value	Excitation Function
0, or LOAD	Applied load (Default)
1, or DISP	Enforced displacement using large mass or SPC/SPCD data
2, or VELO	Enforced velocity using large mass or SPC/SPCD data
3, or ACCE	Enforced acceleration using large mass or SPC/SPCD data

The character fields may be shortened to as little as a single character on input, if desired.

In order to allow maximum input flexibility and to support existing input data files that had used the large mass method, the decision regarding whether to use the older, large mass method, or the newer direct enforced motion will be made based on a search for SPC/SPC1/SPCD data as follows: If the TYPE field on the TLOAD1/2 entries indicates an enforced motion, the EXCITEID field will first be assumed to point to SPC-type data. If not present, MSC.Nastran will then assume the excitation is enforced motion using large masses and will then look for DAREA and various static loading data, just as in the case of applied loads. RLOAD1 and RLOAD2 entries will only look for SPC-type data in the case of enforced motion. For all entries (TLOAD1/2, RLOAD1/2), if the TYPE field indicates an applied load, the program will search only for static loading data.

With this feature, enforced motion data is supplied via the SPC/SPC1/SPCD Bulk Data entries. (Note that components specified in SPCD data must be also be referenced on SPC or SPC1 entries.) Enforced motion could be defined solely with SPC data (without the use of SPCD's), but such usage is discouraged in favor of SPC/SPC1 data in connection with SPCD's. In this way, SPCD entries are selected via the EXCITEID field in the dynamic load data, while the SPC/SPC1 sets are used to identify the constrained displacement set in the Case Control Section. With SPC data alone, enforced motion is applied solely via the Case Control Section. The entries referenced by the EXCITEID field on the TLOAD1/2 or RLOAD1/2 entries do not have to exist. This method of specifying enforced motion is less elegant and somewhat more difficult to interpret.

Procedure for Using the Enforced Motion Feature

In summary, the procedure for applying enforced motion in a dynamic analysis:

1. Specifies the appropriate type of enforced motion to be applied via the TYPE field in RLOAD1/RLOAD2 (field 8) or TLOAD1/TLOAD2 (field 5) Bulk Data entry, as appropriate.
2. Defines the desired enforced motion using the SPCD Bulk Data entry. The set IDs of these SPCD data must match the IDs appearing in the EXCITEID fields of the corresponding dynamic load data in step 1.
3. Ensures that the components referenced in the SPCD Bulk Data entry above are also specified in the SPC1 Bulk Data entry and this SPC1 is also selected in the Case Control Section.

5.7 Frequency-Dependent Loads

A discussion of frequency response calculations is given in “[Frequency Response Analysis](#)” on page 103 of the *MSC.Nastran Basic Dynamics User’s Guide*. The DLOAD entry is used to define linear combinations of frequency dependent loads that are defined on RLOAD1 or RLOAD2 entries. The RLOAD1 entry defines a frequency dependent load of the form

$$\{P(f)\} = A[C(f) + iD(f)]e^{i(\theta - 2\pi f\tau)} \quad \text{Eq. 5-8}$$

where A is defined on a DAREA entry, $C(f)$ and $D(f)$ are defined on TABLEDi entries, θ is defined on a DPHASE entry, τ is defined on a DELAY entry, and f is defined on a FREQ, FREQ1 or FREQ2 entry. The RLOAD2 entry defines a frequency-dependent load of the form

$$\{P(f)\} = AB(f)e^{i\{\phi(f) + \theta - 2\pi f\tau\}} \quad \text{Eq. 5-9}$$

where A is defined on a DAREA entry, $B(f)$ and $\phi(f)$ are defined on TABLEDi entries, θ is defined on a DPHASE entry, and τ is defined on a DELAY entry. The coefficients on the DAREA, DELAY, and DPHASE entries may be different for each loaded degree-of-freedom. The loads are applied to the specified components in the global coordinate system.

Static loading data may be used to supply DAREA values. Using a LOADSET Case Control request with LSEQ Bulk Data entries, the user can build DAREA load vectors from any valid static load set. This option is available in the frequency response solutions and in superelement analysis.

A discussion of random response calculations is given in Section 12.2 of *The NASTRAN Theoretical Manual*. The RANDPS entry defines load set power spectral density factors for use in random analysis of the form

$$S_{jk}(f) = (X + iY)G(f) \quad \text{Eq. 5-10}$$

where $G(f)$ is defined on a TABRNDi entry. The subscripts j and k define the subcase numbers of the load definitions. If the applied loads are independent, only the diagonal terms ($j = k$) need be defined. The RANDT1 entry is used to specify the time lag constants for use in the computation of the autocorrelation functions.

5.8 Time-Dependent Loads

A discussion of transient response calculations is given in “[Transient Response Analysis](#)” on page 149 of the *MSC.Nastran Basic Dynamics User’s Guide*. The DLOAD entry is used to define linear combinations of time-dependent loads that are defined on TLOAD1 and TLOAD2 entries. The TLOAD1 entry defines a time-dependent load of the form

$$\{P(t)\} = \{AF(t - \tau)\} \quad \text{Eq. 5-11}$$

where A is defined on a DAREA entry, τ is defined on a DELAY entry, and $F(t - \tau)$ is defined on a TABLEDi entry. The TLOAD2 entry defines a time-dependent load of the form

$$\{P(t)\} = \begin{cases} 0 & , t < (T_1 + \tau) \text{ or } t > (T_2 + \tau) \\ A\tilde{t}^B e^{C\tilde{t}} \cos(2\pi F\tilde{t} + P) & , (T_1 + \tau) \leq t \leq (T_2 + \tau) \end{cases} \quad \text{Eq. 5-12}$$

where T_1 and T_2 are time constants, $\tilde{t} = t - T_1 - \tau$, A and τ are the same as on the TLOAD1 entry, B is the growth coefficient, C is the exponential coefficient, F is the frequency, and P is the phase angle.

The coefficients on the DAREA and DELAY entries may be different for each loaded degree-of-freedom. The loads are applied to the specified components in the global coordinate system.

Two methods are available to automatically create DAREA values from static load data. Simple static loads that supply load values directly to grid points (such as FORCE, MOMENT, SLOAD, and PLOAD Bulk Data entries) will add to the DAREA factors when the load identification matches the requested set.

More complicated loads may be generated using the LOADSET Case Control command. The requested LSEQ Bulk Data entry matches DAREA sets to static loading sets. Any type of static load (e.g., LOAD, GRAV, RFORCE, and SPCD Bulk Data entries) may be generated for either superelement or residual only (non-superelement) models.

Nonlinear effects are treated as an additional applied load vector, for which the components are functions of the degrees-of-freedom. This additional load vector is added to the right side of the equations of motion and treated along with the applied load vector during numerical integration. It is required that the points to which the nonlinear loads are applied and the degrees-of-freedom on which they depend be members of the solution set; i.e., that they cannot be degrees-of-freedom eliminated by

constraints. It is further required that, if a modal formulation is used, the points referenced by the nonlinear loads be members of the set of extra scalar points introduced for dynamic analysis (see Section 11.2 of *The NASTRAN Theoretical Manual*).

There are four different types of nonlinear transient forcing functions: arbitrary function generators, multiplies, positive power functions, and negative power functions. For a discussion of nonlinear forcing functions, see Section 11.2 of *The NASTRAN Theoretical Manual*.

The NOLIN1 entry defines a nonlinear load of the form

$$P_i(t) = S_i T(u_j) \quad \text{Eq. 5-13}$$

where P_i is the load applied to u_i , S_i is a scale factor, $T(u_j)$ is a tabulated function defined with a TABLEDi entry, and u_j is any permissible displacement component or velocity component.

The NOLIN2 entry defines a nonlinear load of the form

$$P_i(t) = S_i u_j u_k \quad \text{Eq. 5-14}$$

where u_j and u_k are any permissible pair of displacement components and/or velocity components. They may be the same.

The NOLIN3 entry defines a nonlinear load of the form

$$P_i(t) = \begin{cases} -S_i(u_j)^A & , u_j > 0 \\ 0 & , u_j \leq 0 \end{cases} \quad \text{Eq. 5-15}$$

where A is an exponent and u_j is any permissible displacement component or velocity component.

The NOLIN4 entry defines a nonlinear load of the form

$$P_i(t) = \begin{cases} -S_i(u_j)^A & , u_j < 0 \\ 0 & , u_j \geq 0 \end{cases} \quad \text{Eq. 5-16}$$

where A is an exponent and u_j is any permissible displacement component or velocity component.

5.9 Simplified Static Loading Data in Dynamic Analysis

Prior to MSC.Nastran 2001, static loading data in dynamic loads had to be included through the use of the LOADSET/LSEQ combination. The LOADSET command, which is selected in the Case Control Section, points to a LSEQ Bulk Data entry. The LSEQ Bulk Data entry, in turn, acts as a "bridge" between the DAREA ID and the static load ID/thermal load ID. If the DAREA ID of the selected dynamic load (RLOAD1/RLOAD2/TLOAD1/TLOAD2/ACSRCE) matches the DAREA ID on the selected LSEQ entry, then all static loads and thermal loads referenced in the selected LSEQ Bulk Data entry will be selected automatically. Note that the DAREA entry does even have to exist in this case.

With MSC.Nastran 2001, it is no longer necessary for the user to explicitly specify LOADSET/LSEQ combination in order to employ static loading data in dynamic analysis. Instead, when the user selects a dynamic load, all static loads and thermal loads that have the same ID as the DAREA ID on the dynamic load entry are automatically selected. The usage of this feature is illustrated below.

Pre-2001 Usage

```
.
CEND
.
$ THE FOLLOWING REQUEST SELECTS LSEQ ENTRY 1000
LOADSET = 1000
$ THE FOLLOWING REQUEST SELECTS RLOAD1 ENTRY 150
DLOAD = 150
.
BEGIN BULK
$ THE FOLLOWING ENTRY, WHICH ACTS AS A "BRIDGE"
$ BETWEEN EXCITEID ID 100 AND STATIC LOAD ID 200,
$ CAUSES THE SELECTION OF PLOAD4,200
LSEQ,1000,100,200
PLOAD4,200,...
RLOAD1,150,100,...
.
```

2001 Usage

```

.
CEND
.
$ THE FOLLOWING REQUEST SELECTS RLOAD1 ENTRY 150
DLOAD = 150
.
BEGIN BULK
$ THE FOLLOWING PLOAD4 IS AUTOMATICALLY SELECTED
$ BECAUSE ITS ID OF 100 MATCHES THE EXCITEID ID OF THE
$ SELECTED RLOAD1 ENTRY
PLOAD4,100,...
RLOAD1,150,100,...
$ THE ABOVE USAGE IS EQUIVALENT TO THE USER
$ SELECTING AN LSEQ BULK DATA OF THE FORM:-
$ LSEQ,SET_ID,100,100,100
.

```

The 2001 design should alleviate the need for the LOADSET/LSEQ combination. However, in order to support existing models, the LOADSET/LSEQ feature will continue to be honored and supported as in earlier versions of MSC.Nastran. It should also be pointed out that the new feature will be activated *if and only if* the user does *not* have a LOADSET/LSEQ selection.

The DAREA Bulk Data entry is used to specify point loads in dynamic analysis. In the case of grid points, these loads are implicitly assumed to be in the displacement (or local) coordinate systems of those points. Until now, the usage of these data in superelement analysis had been restricted to the residual structure. Prior to MSC.Nastran 2001, to specify DAREA-type loads for upstream superelements in dynamic analysis, the user was forced to specify the LOADSET/LSEQ combination in conjunction with the appropriate FORCE, FORCE1, FORCE2, MOMENT, MOMENT1, MOMENT2, or SLOAD Bulk Data entry. An enhancement incorporated into MSC.Nastran avoids this problem by automatically converting all DAREA Bulk Data entries for grid and scalar points into the appropriate equivalent FORCE/MOMENT/SLOAD Bulk Data entries.

Two additional, important advantages of the above enhancement follow.

- When performing dynamic analysis using the modal approach, it is very often desirable to employ residual vectors to improve the quality of the solution. The user must specify static loads at those points that are dynamically excited. However, with the above automatic conversion feature, it is not necessary to explicitly specify static loads for the purpose of residual vector calculations. Such loads are automatically generated by the MSC.Nastran.

- Prior to MSC.Nastran 2001, the usage of the DAREA Bulk Data entry had been restricted to dynamic analysis. The above automatic conversion feature permits the DAREA entry to be used in static analysis as well. This is particularly advantageous when the user wishes to apply loads at grid points in the displacement (or local) coordinate systems of those grid points.

When the above conversion is made, a message is issued at the end of the Preface module informing the user of the conversion. This message also outputs an image not only of each converted DAREA Bulk Data entry, but also of the corresponding FORCE/MOMENT/SLOAD Bulk Data entry into which it has been converted.

5.10 Combining of Loads

The LOADCYH Bulk Data entry is used to define the harmonic coefficients of a load in cyclic symmetry analysis. The LOADCYN entry is used to define a physical load in cyclic symmetry analysis, and the LOADCYT entry specifies a tabular load input for axisymmetric cyclic symmetry problems.

The LOAD entry defines a static loading condition that is a linear combination of load sets consisting of loads applied directly to grid points, pressure loads, gravity loads, and centrifugal forces. This entry must be used if gravity loads are to be used in combination with loads applied directly to grid points, pressure loads, or centrifugal forces. The application of the combined loading condition is requested in the Case Control Section by selecting the set number of the LOAD combination.

It should be noted that the equivalent loads (thermal, enforced deformation, and enforced displacement) must have unique set identification numbers and be separately selected in Case Control. For any particular solution, the total static load will be the sum of the applied loads (grid point loading, pressure loading, gravity loading, and centrifugal forces) and the equivalent loads.

The LSEQ entry defines static load sets that may be referenced by dynamic load entries RLOAD1, RLOAD2, TLOAD1, and TLOAD2. The LSEQ entry may be used to select static loading conditions in the superelement static solution sequences and is used to correlate static loading conditions with dynamic loads in dynamic analysis.

The loads actually applied to points may be requested by use of the OLOAD entry.

Temperature Loads and Enforced Deformations in Subcase Combinations (SUBCOMS)

The use of the Case Control Section entries SUBCOM and SYMCOM in static analysis provides the user with a relatively straightforward method to combine the results of two or more immediately preceding subcases or SYM cases respectively. For example, the following portion of a Case Control Section illustrates the use of the SUBCOM.

```

      :
SUBCASE 1
  LOAD=1
  TEMP (LOAD) =7
SUBCASE 2
  LOAD=2
SUBCASE 10
  LOAD=10
  DEFORM=11
SUBCOM 20
  SUBSEQ= .5 , 1 . , 1 .6
      :

```

Here, SUBCOM 20 provides results for a combined loading condition of the form

$$0.5 \cdot (\text{LOAD} = 1 + \text{TEMP}(\text{LOAD}) = 7) + 1.0 \cdot (\text{LOAD} = 2) + 1.6 \cdot (\text{LOAD} = 10 + \text{DEFORM} = 11) \quad \text{Eq. 5-17}$$

In MSC.Nastran, the results for this combined loading condition are efficiently obtained by forming the linear combination of the displacement vectors obtained in the previous subcases. Thus,

$$\{U\}_{\text{SUBCOM 20}} = 0.5 \cdot \{U\}_{\text{SUBCASE 1}} + 1.0 \cdot \{U\}_{\text{SUBCASE 2}} + 1.6 \cdot \{U\}_{\text{SUBCASE 10}} \quad \text{Eq. 5-18}$$

and the element forces and stresses are calculated in the usual manner from the displacement vector for the combined loading condition. This technique is quite straightforward when the several loading conditions are composed of forces and moments. However, if any of the previous subcases contain thermal loads and/or enforced element deformations, the user must exercise some care in order to obtain correct element stresses and forces for the combined loading condition defined through the SUBCOM.

5.11 p-Element Loads and Constraints

Point Loads

The FORCE entry can be used to define point loads applied to grid points. No point loads can be applied to finite element edges or faces.

Line and Pressure Loads in p-Element Analysis

The GMLOAD entry is used to specify line and pressure loads by:

1. Applying directly to the finite elements entities, FEEDGEs and FEFACEs.
2. Applying to the geometry entities GMCURV and GMSURF.

Body Loads

Body loads consists of thermal loads and gravity loads.

For thermal loading, the TEMP Bulk Data entry is used for a temperature distribution that is trilinear over the element, whereas the TEMPF entry is used over a set of elements. The initial or reference temperature of a body must be supplied using these entries.

Gravity loads are defined by the GRAV entry.

Boundary Conditions

The SPC and SPCD entries are used for the point constraints that are allowed only on corner GRID points.

The GMSPC entry is used to define zero constraints for FEEDGEs, FEFACEs, GMCURVs, and GMSURFs.

For non-zero constraints, the GMBC entry is used.

In general, for multiple input data for FEEDGE, FEFACE, GMCURV, and GMSURF entities, the hierarchy set to resolve these conflicts is:

1. GRIDs
2. FEEDGEs
3. GMCURVs
4. FEFACEs
5. GMSURFs

These priorities apply for the constraints and the enforced displacement input data.

The rules for nonzero boundary constraints:

1. Displacement values specified for each component of a given GMSURF entry are applied to the same component of all GRID, FEEDGE and FEFACE degrees-of-freedom that lie within the GMSURF.
2. Displacement values specified for each component of different GMSURF entries are averaged and applied to the same component of all GRID, FEEDGE degrees-of-freedom that are shared by (common to) the multiple GMSURFs.
3. Displacement values specified for a given FEFACE entry are applied to all GRID, FEEDGE and FEFACE degrees-of-freedom that lie within the FEFACE. This data overrides the data that is specified for all the components of the given GRID, FEEDGE and FEFACE degrees-of-freedom that lie within the FEFACE by using GMSURF entries.
4. Displacement values specified for each component of different FEFACE entries are averaged and applied to the same component of all GRID, FEEDGE degrees-of-freedom that are shared by (common to) the multiple FEFACEs. This data overrides the data that is specified for all the components of the given FEEDGE and GRIDs by using GMSURF entries.
5. Displacement values specified for each component of a given GMCURV entry are applied to the same component of all GRID, FEEDGE, degrees-of-freedom that lie within the GMCURV. This data overrides the data for all components that is specified for the given FEEDGE and GRIDs by using GMSURF and FEFACE entries.
6. Displacement values specified for each component of different CURV entries are averaged and applied to the same component of all GRID degrees-of-freedom that are shared by (common to) the multiple GMCURVs. This data overrides the data for all the components that is specified for the given FEEDGE and GRIDS by using GMSURF and FEFACE entries.
7. Displacement values specified for each component of a given FEEDGE entry are applied to the same component of all GRID, FEEDGE, degrees-of-freedom that lie within the FEEDGE. This data overrides the data for all components that is specified for the given FEEDGE by using GMCURV or FEFACE or GMSURF entries.

8. Displacement values specified for each component of different FEEDGE entries are averaged and applied to the same component of all GRID degrees-of-freedom that are shared by (common to) the multiple FEEDGEs. This data overrides the data for all the components that is specified for the given FEEDGE by using GMCURV or FEFACE or GMSURF entries.
9. Grids have the highest priority, i.e., any value/property specified using a GRID entry overrides all other information associated with that GRID. If multiple entries are used for a given GRID, e.g., multiple SPCs, then the existing rules of MSC.Nastran govern (SPCs are combined, FORCE fare added, SPCDs for the same component are not allowed).
10. It is important to recall that these displacements are assumed to be in the global coordinate system and that the interconsistency of the output coordinate systems of the various GRIDs, FEEDGEs, FEFACEs is not checked.

Resolution of Conflicting Zero Constraints

In general, the hierarchy set to resolve the conflicts arising in the enforced displacement input data is the same as for the constraints:

1. GRIDs
2. FEEDGEs
3. GMCURVs
4. FEFACEs
5. GMSURFs

This means that:

1. The specified component of a given GMSURF entry that is to be constrained is also constrained on all GRID, FEEDGE and FEFACE degrees-of-freedom that lie within the GMSURF.
2. The specified component of different SURF entries are combined and applied to all GRID, FEEDGE degrees-of-freedom that are shared by (common to) the multiple GMSURFs.
3. The specified component of a given FEFACE entry that is to be constrained is also constrained on all GRID, FEEDGE and FEFACE degrees-of-freedom that lie within the FEFACE. This data overrides the data that is specified for the given FEFACE by using GMSURF entries.

4. The specified component of different FEFACE entries are combined and applied to all GRID, FEEDGE degrees-of-freedom that are shared by (common to) the multiple FEFACE. This data overrides the data that is specified for the given FEEDGE and GRIDs by using GMSURF entries.
5. The specified component of a given GMCURV entry that is to be constrained is also constrained on all GRID, FEEDGE degrees-of-freedom that lie within the GMCURV. This data overrides the data that is specified for the given FEEDGE by using GMSURF or FEFACE entries.
6. The specified component of different GMCURV entries are combined and applied to all GRID degrees-of-freedom that are shared by (common to) the multiple GMCURVs. This data overrides the data that is specified for the given FEEDGE by using GMCURV or FEFACE entries.
7. The specified component of a given FEEDGE entry that is to be constrained is also constrained on also constrained on all GRID, FEEDGE degrees-of-freedom that lie within the FEEDGE. This data overrides the data that is specified for the given FEEDGE by using GMCURV or FEFACE or GMSURF entries.
8. The specified component of different FEEDGE entries are combined and applied to all GRID degrees-of-freedom that are shared by (common to) the multiple FEEDGES. This data overrides the data that is specified for the given FEEDGE by using GMCURV or FEFACE or GMSURF entries.
9. Grids have the highest priority, i.e., any value/property specified using a GRID entry overrides all other information associated with that GRID. If multiple entries are used for a given GRID, e.g., multiple SPCs, then the existing rules of MSC.Nastran govern (SPCs are combined, FORCE is added, SPCDs for the same component are not allowed).
10. It is important to recall that the interconsistency of the output coordinate systems of the various GRIDs, FEEDGES, FEFACEs is not checked.

5.12 Summary of Nonlinear Static Loads

Loads Overview

The static loads in nonlinear analysis consist of a subset of the total MSC.Nastran static load set. Most of the relevant loads data applicable to the linear static analysis are also applicable to nonlinear static analysis except for the Bulk Data entry DEFORM. The enforced displacements can be applied as loads using the Bulk Data entry SPCD, that defines an enforced motion of a grid in a specific fixed direction. Each grid point with an enforced displacement must also appear on an SPC or SPC1 entry.

The loads are selected in each subcase by the LOAD Case Control command. The load for a subcase is subdivided into the number of increments specified for the subcase. The solution strategy in nonlinear is to apply the loads in an incremental fashion until the desired load level is reached. The algorithms “remember” the loads from one subcase to the next. If the load reaches the desired level in a subcase and if the load description is left out in the subsequent subcase inadvertently, the nonlinear solution algorithm will begin in an incremental fashion to remove the load. The loads described below make up the valid subset applicable for nonlinear static analysis.

Concentrated Loads

The concentrated loads consist of two categories: those that are stationary in direction throughout the analysis and those that follow the grid motion. The first category consists of the following:

FORCE
MOMENT
SLOAD

The second category consists of the following:

FORCE1
FORCE2
MOMENT1
MOMENT2

The three FORCE entries differ only in the manner in which the direction of the force is specified. FORCE uses the components of a vector. FORCE1 uses two grid points, not necessarily the same as the loaded grid points. FORCE2 defines the direction of the force as a vector that is the vector product of two other vectors. The distinctions between the three MOMENT entries are similar. The SLOAD entry is used to apply loads to scalar points.

Distributed Loads

The distributed loads consist of two categories: those that are stationary in direction throughout the analysis and those that follow the element motion. The first category consists of the PLOAD1.

The second category consists of the following:

PLOAD
PLOAD2
PLOAD4

Loads may be applied to the interiors of finite elements. The PLOAD1 entry defines concentrated and linearly distributed forces and moments to the CBEAM element. However, the stress calculation for nonlinear CBEAM element ignores the distributed loads, but lump them onto the nodal points. The PLOAD entry is exceptional in that it references three or four arbitrarily located grids rather than an element. The PLOAD2 entry defines a uniform static pressure load applied to the CQUAD4 or CTRIA3 elements. The PLOAD4 defines a pressure load on surfaces of CHEXA, CPENTA, CTETRA, CTRIA3, and CQUAD4 elements that need not be normal to the surface. Since the surface hyperelastic elements CQUAD4, CQUAD8, CQUAD, CTRIA3, and CTRIA6 are plane strain elements, no pressure load may be applied on these elements.

Mass Related Loads (GRAV and RFORCE Entries)

The mass related loads are gravity and centrifugal loads, which require specification of mass data. The gravity loads that can be specified on the GRAV Bulk Data entry are stationary in direction throughout the analysis. The GRAV entry is used to define the direction and magnitude of the gravity vector in some identified coordinate system. The components of the gravity vector are multiplied by the mass matrix to obtain the components of the gravity force at each grid point.

Centrifugal loads are specified on the RFORCE Bulk Data entry. RFORCE defines the components of a spin vector that is used internally to compute centrifugal forces. Each component of the spin vector is multiplied by a scale factor.

Thermal Loads

Thermal loads are selected in each subcase by the Case Control command TEMP(LOAD). Thermal loading that is a follower loading is accomplished through the following entries:

TEMP
TEMPD

TEMPP1
 TEMPP3
 TEMPRB

The TEMP and TEMPD provide for grid point temperature specification which are then interpolated to points within elements. For the nonlinear CQUAD4 and CTRIA3 elements the TEMPP1 provides for an average temperature and thermal gradient in the thickness direction and the TEMPP3 provides for temperature at points through the thickness. TEMPP1 and TEMPP3 are not available for the plane strain hyperelastic elements CQUAD4, CQUAD8, CQUAD, CTRIA3, and CTRIA6. The TEMPRB provides for average temperature at ends A and B for the CROD and CBEAM element. Average temperature specified directly for an element will take precedence over the temperature interpolated from the elements connected grid points.

The Case Control command TEMP(INIT) is required above subcase level to specify an initial temperature distribution. The use of TEMP(MAT) or TEMP(BOTH) will cause a fatal error. Subcases that do not contain a TEMP(LOAD) will default to TEMP(INIT), which implies no thermal load. Thermal loads are often used to provide initial strains.

Load Combinations

Various load specifications can be combined using the LOAD, CLOAD, and LSEQ entries.

The LOAD provides for the linear combination of static load sets. The static loads (LOAD, FORCE, etc.) applied to the upstream superelements cannot be referenced by the LOAD Case Control command. The CLOAD entry is designed to apply static loads to upstream superelements by combining loads defined in LSEQ entries. The (CLOAD, LSEQ) combination allows the nonlinear algorithms to apply in an incremental fashion on the upstream superelement loads to the boundary of the residual structure.

The CLOAD defines a static load combination for superelement loads acting on residual boundary and the LSEQ defines a static load combination for superelement upstream loads. The LSEQ assigns load vectors to the superelements and numerically labels them via the DAREA field entry value. The LID and TID field entries point to the appropriate load and temperature distribution entries. The CLOAD defines a static load as a linear combination of previously calculated superelement loads defined by the LSEQ. The IDVi fields correspond directly to the numeric labels defined via the DAREA fields of the LSEQ entries.

The CLOAD must be selected in the residual solution subcase via the Case Control command, CLOAD. Any loads referenced by CLOAD should not be again referenced by LOAD until Version 66, otherwise the load will be doubled: e.g., GRAV and TEMP.

This deficiency has been corrected in Version 67. The LSEQ is selected by a LOADSET Case Control command above any residual subcase. Only one LOADSET may appear in Case Control. The hierarchy of the loads data is shown schematically in [Figure 5-0](#). An example for the Case Control set-up follows:


```
      :
SEALL=ALL
LOADSET = 1000 $ Selects LSEQ 1000 for upstream loads
SUPER = ALL $ Identify SEs to process
      :
      DISP = ALL
      :
      :
      :
$ NONLINEAR SOLUTIONS FOR RESIDUAL SUPERELEMENTS
SUBCASE 10
      CLOAD = 1001 $ Refers to CLOAD Bulk Data
      NLPARM = 12 $ Iteration control
      :
SUBCASE 20
      CLOAD = 1002
      LOAD = 10 $ Residual SE load
      NLPARM = 22
      :
```

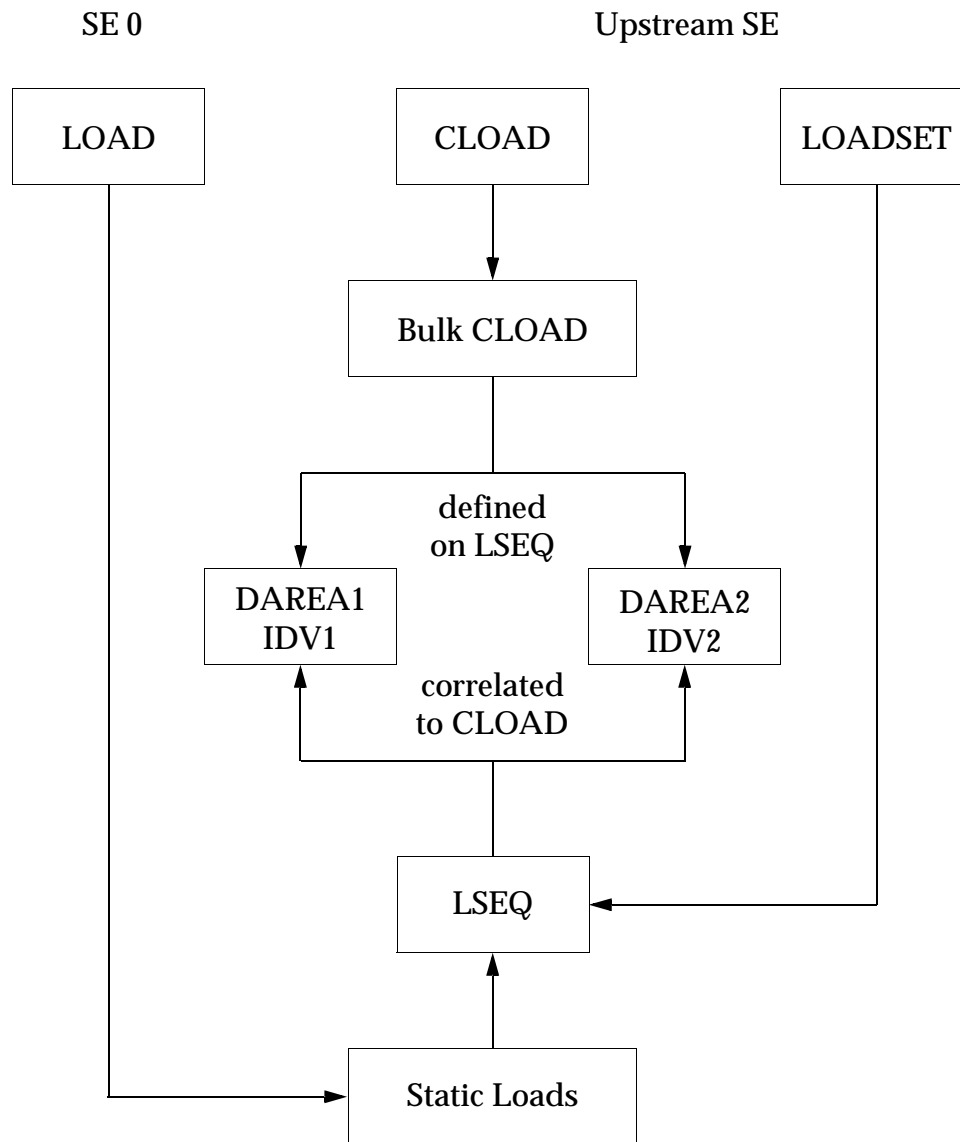


Figure 5-0 Hierarchy of Load Data

5.13 Summary of Nonlinear Transient Loads

Transient Loads And Initial Conditions

The methods employed to define loads in nonlinear transient analysis are similar to those used in the linear solutions. A single degree-of-freedom or a set of grid points may be loaded with force pattern that varies with time. Functions may be tabular such as an earthquake or a booster liftoff, or they may be simple analytic functions such as a sine wave. Simple static load sets generated in both upstream superelements and the default residual may be used to create the dynamic loads. They may be scaled and combined with other loads to simulate complex loading problems.

For special problems involving simple scalar nonlinearities, the traditional nonlinear loads (NOLINi option) may be used in both types of transient solutions. These are useful for simulating nonlinear damping mechanisms and rotational coupling not provided by the CGAP elements.

Transient loads define the loadings as functions of time and the location. They can be a load applied at a particular degree-of-freedom, pressure over the surface area, or the body force simulating an acceleration. The time history is provided by the TLOADi Bulk Data entries and the static loads may be converted to dynamic loads using LSEQ entry. The transient dynamic loads are selected in the Case Control Section by the following commands:

DLOAD	Selects TLOADi or DLOAD Bulk Data entries. Each stage of the transient solution may be defined with a separate SUBCASE, each with its own DLOAD request. The DLOAD entry is restricted to act only in the time defined in its subcase.
LOADSET	Selects LSEQ entries. Must be placed above the solution subcases and/or in the Superelement Subcases.
NONLINEAR	Selects NOLINi entries. May not be changed between solution subcases.
NLLOAD	Required for NOLINi output request.

Bulk Data entries associated with the transient loads in SOL 99 (or 129) are briefly described below:

TLOAD1, TLOAD2	Required data defining the coupling between load factors (DAREA entry) and functions of time (TABLEDi, DELAY, or coefficients). Specifies the time functions, $F_i(t)$, for dynamic load i . TLOAD1 defines the load history in TABLEDi format while TLOAD2 defines the load history by an analytic expression. Note that each TLOADi entry must have a unique set identification number.
DAREA	Defines load scale factors, A_{ij} , for each degree-of-freedom, u_j requested in set i . The DAREA factors may also be derived from static load entries with the LSEQ Bulk Data entry.
LSEQ	Generates transient load history for static loads and controls the assembly of static load vectors to be used in dynamic analysis. Each set is equivalent to a subcase in Case Control. It will cause a static load vector A_j to be generated and will label it with the DAREA identification. This is useful for modeling problems with distributed transient loading such as GRAV loads or pressurized areas.
DLOAD	Combines different TLOADi functions into a single set and scales each by a factor, C_{ki} as shown in Eq. 5-30. Performs the same function as the static LOAD Bulk Data input. Unlike statics, this is the only method to combine loads in dynamic analysis.
TABLEDi, i=1,2,3,4	Used with TLOAD1 entries to specify general time varying tabular functions. Since extrapolation is used beyond the range of data the user is cautioned to be careful with the end points.
DELAY	Specifies the delay time for applying the forcing function defined in TLOADi to each GRID point. The DELAY entry is associated with the degree-of-freedom and not the forcing function in each TLOADi. This option is useful for defining loads that travel across a structure such as an oblique wave along a dam or a bump in the road exciting the wheels of a traveling vehicle. This saves the analyst the effort of defining multiple TLOADi entries for simple time lags.

NOLIN i , $i=1,2,3,4$	Define specific nonlinear scalar loads as functions of velocity and/or displacement. Basic analog nonlinear transfer functions are defined. However, these functions do not generate tangent matrix terms and the Newton iteration method may experience difficulties in convergence. Line search and BFGS solution methods are recommended for these models.
NOLIN1	Nonlinear transient load as a tabular function.
NOLIN2	Nonlinear transient load as products of two variables.
NOLIN3	Nonlinear transient load as a positive variable raised to a power.
NOLIN4	Nonlinear transient load as a negative variable raised to a power.
NLRGAP	The NLRGAP is an idealization of the contact between any circular shaft enclosed by a circular housing (Figure 5-1). In this idealization, grid point A represents the outer surface of the shaft while grid point B represents the inner surface of the housing. Grid points A and B are coincident and must have a parallel displacement coordinate system. A rudimentary friction capability is included to account for the friction-induced torque and friction-induced lateral loads that occur with a spinning shaft. However, friction can be neglected, in which case the solution is consistent with that for a nonrotating shaft in a frictionless housing.

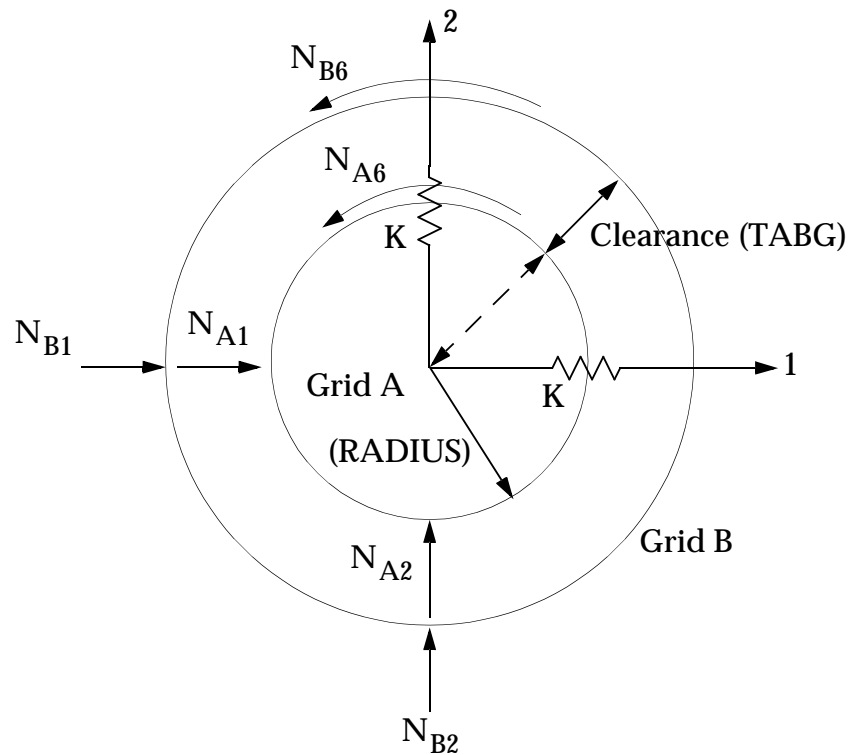


Figure 5-1 Radial Gap Orientations and Nonlinear Load Sign Conventions

The equations underlying the NLRGAP measure the relative radial displacement between the shaft grid and the housing grid in the displacement coordinate system of grid points A and B. When the relative radial displacement is greater than the clearance (i.e. when contact has occurred), contact forces as well as frictional forces are automatically applied to the grids. The contact and frictional forces are dependent on the contact stiffness and the coefficient of friction.

NLRGAP Underlying Equations

Definitions

- $K(t,u)$ Contact stiffness, function of time or displacement
- $G(t)$ Radial clearance, function of time
- $\mu(t)$ Coefficient of friction, function of time. Positive value implies shaft rotation from axis 1 towards axis 2.
- R Shaft radius for friction induced torque calculation

Equations

The NLRGAP is internally composed of the following equations. It is helpful to visualize Grid A as belonging to the shaft and Grid B as belonging to the housing. Consider each grid to have DOF 1 and 2 in the plane of action (e.g., DOF 1 = X and DOF 2 = Y for XY orientation).

The element relative displacement in directions 1 and 2 are:

$$\Delta U_1 = U_{A1} - U_{B1} \quad \text{Eq. 5-19}$$

$$\Delta U_2 = U_{A2} - U_{B2} \quad \text{Eq. 5-20}$$

The relative radial displacement is:

$$\Delta r = \sqrt{\Delta U_1^2 + \Delta U_2^2} \quad \text{Eq. 5-21}$$

The force in the gap when it is open ($\Delta r \leq G(t)$) is zero. When the gap is closed ($\Delta r > G(t)$), the penetration is defined as:

$$P = \Delta r - G(t) \quad \text{Eq. 5-22}$$

The nonlinear gap forces are:

$$S = K(t,u) \left(1 - \frac{G(t)}{\Delta r} \right) \quad \text{Eq. 5-23}$$

$$N_{A1} = -\Delta U_1 S + \Delta U_2 S \mu(t) \quad \text{Eq. 5-24}$$

$$N_{B1} = \Delta U_1 S - \Delta U_2 S \mu(t) \quad \text{Eq. 5-25}$$

$$N_{A2} = -\Delta U_2 S - \Delta U_1 S \mu(t) \quad \text{Eq. 5-26}$$

$$N_{B2} = \Delta U_2 S + \Delta U_1 S \mu(t) \quad \text{Eq. 5-27}$$

If the shaft radius R is input, then the following friction induced torque loads are generated.

$$N_{A6} = -R \Delta r S \mu(t) \quad \text{Eq. 5-28}$$

$$N_{B6} = R \Delta r S \mu(t) \quad \text{Eq. 5-29}$$

Benefit

Allows you to model contact between two coaxial cylinders, for example shaft and housing.

Input

The Bulk Data entry NLRGAP is used to model the radial gap.

The contact stiffness can be input either as a function of time or as a function of penetration. In both cases, the stiffness curves are input on TABLEDi Bulk Data entries which are referenced by the NLRGAP entry. Inputting stiffness as a function of time allows the user to model special situations where structural changes over time cause the contact stiffness to vary over time. Inputting stiffness as a function of penetration allows the user to input a nonlinear stiffness curve which might be required, for example, if the shaft or housing were covered by some nonlinear-elastic material.

As with stiffness, both the coefficient of friction and the clearance can be functions of time. The time-varying coefficient of friction might be used where the quality of lubrication varied with time, or if experimental data showed that the surface characteristics of the shaft and housing varied with time. The clearance can be input as a function of time to model situations where an accelerating, spinning rotor expands due to centripetal loads.

The NLRGAP entry can also approximate the torque on the shaft and housing due to friction. If this effect is desired, the RADIUS field of the NLRGAP must be specified. If not specified, then friction effects will still be included in the lateral loads as shown in Eq. 5-24 through Eq. 5-27. The torque loads, Eq. 5-28 and Eq. 5-29, will not be computed nor applied if RADIUS is input as 0.0 or left blank.

As with the NOLINs, the NLRGAP is selected by the NONLINEAR Case Control command.

Output

The output is the nonlinear loads applied to the NLRGAP grids to simulate the contact. The loads can be recovered with the NLLOAD Case Control command.

Guidelines

1. There should be mass on both grids of the NLRGAP. Otherwise erratic results and divergence can occur. A small amount of damping (e.g. CDAMPi) between the two grids may improve stability for some problems.

2. The two grids listed on the NLRGAP should be coincident and have parallel displacement coordinate systems.
3. The friction capability is somewhat arbitrary in sign. Referring to **Figure 5-1**, a positive coefficient of friction is consistent with shaft rotation from axis 1 towards axis 2 (counter-clockwise). If the shaft is rotating in the opposite direction, then a negative coefficient of friction must be input in order for the friction induced loads to have the proper signs.
4. In SOL 129, setting ADJUST = 0 on the TSTEPNL entry will turn off the adaptive time step. This is useful if large contact stiffness is causing an excessively small time step.
5. As with the NOLINs, the forces applied to the NLRGAP grids to simulate the radial contact are based upon the relative displacements of the grids from the previous time step. Therefore, the larger the contact stiffness, the smaller the necessary time steps. The smaller the contact stiffness, the larger the penetration. Use the smallest contact stiffness possible to model the contact to the accuracy desired. The time step may need to be decreased by 10 times or more compared to the same model without NLRGAPs and NOLINs.
6. The NLRGAP is based upon small displacement theory. Therefore, for most accurate results, especially with respect to friction induced torque, the clearance should be small relative to the housing diameter. The friction induced torque applied to grids A and B will be equal and opposite in direction.

Limitations

1. The NLRGAP can only be positioned in the XY, YZ or ZX plane of the displacement coordinates of the referenced grids.
2. The DOF in the plane of the NLRGAP must be in the d-set. This means that these DOF cannot be constrained by SPCs or be dependent DOF on MPCs and RBE type elements.
3. The NLRGAP can only be used in SOLution sequences 9, 109 and 129.

A diagram of input data linkages is shown in **Figure 5-2**. The general form of the transient load, P_{kj} , for each degree-of-freedom, u_j , is:

$$P_{kj} = C_{ki} \cdot A_j \cdot F_i(t) \quad \text{Eq. 5-30}$$

where the factors are defined by various Bulk Data entries as shown in **Figure 5-3**.

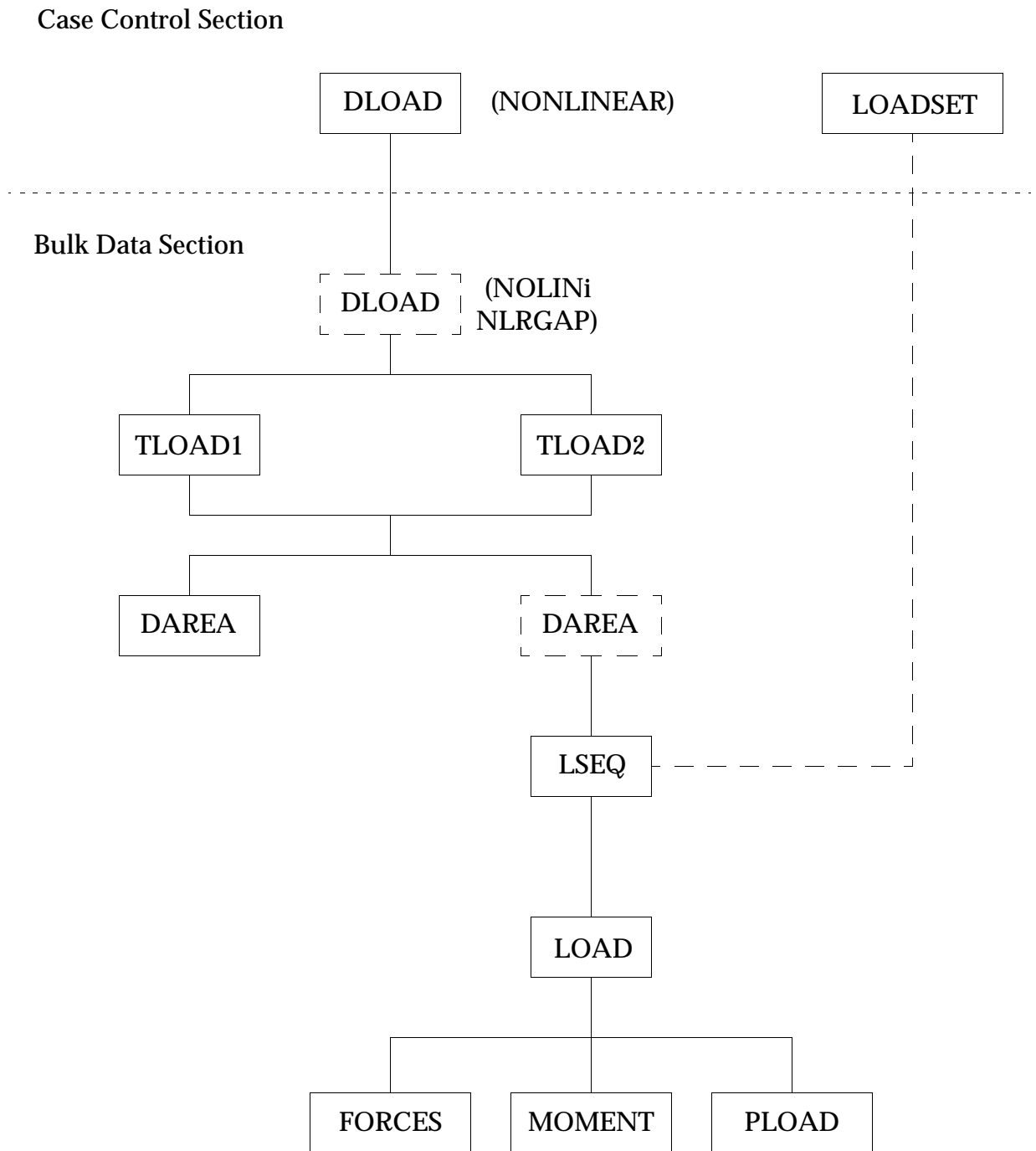
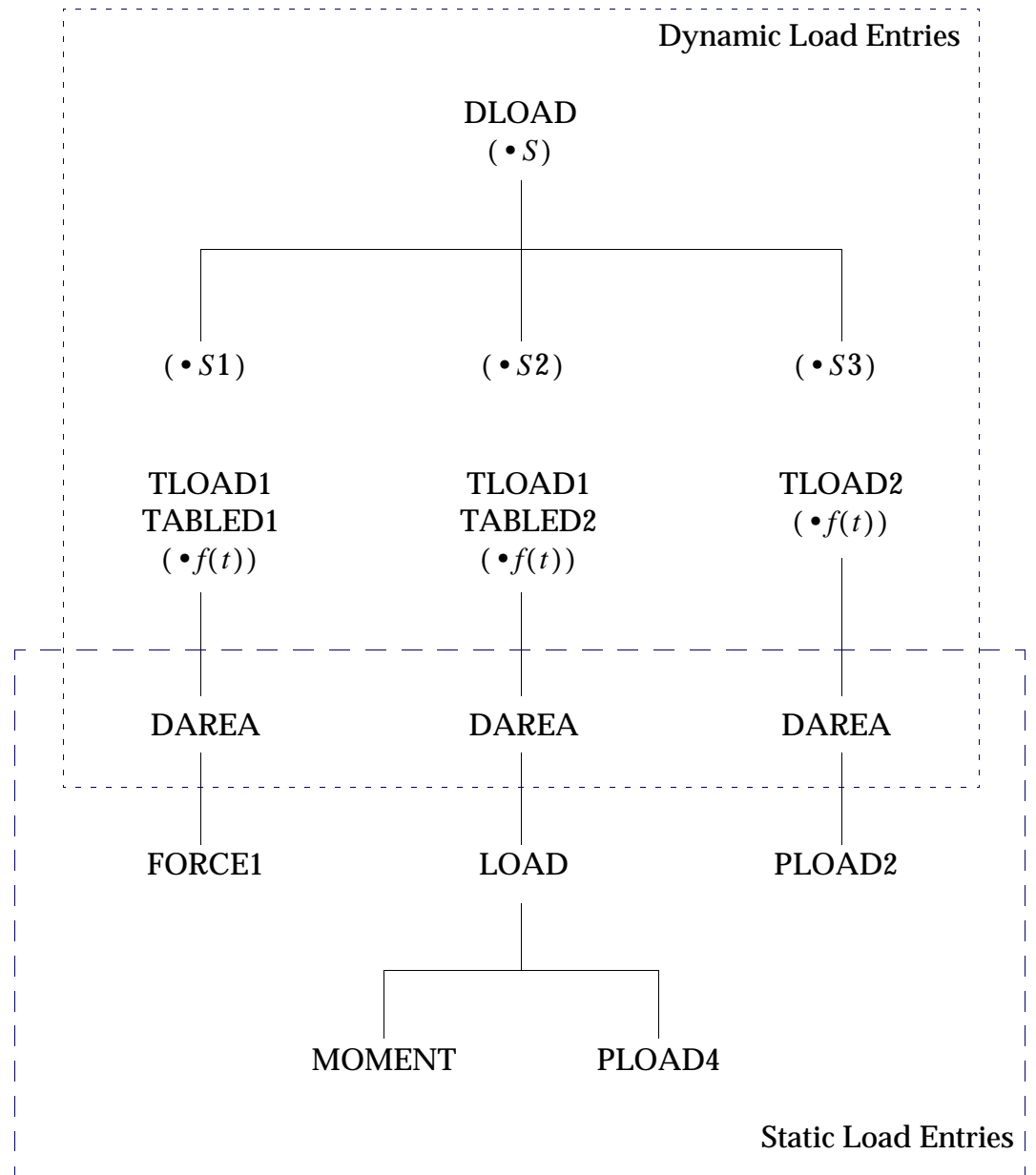


Figure 5-2 Dynamic Loads for Transient Analysis



Note that each load is scaled by $S \cdot S_i \cdot f(t)$ at each time step in dynamic loads.

Figure 5-3 Example of Combining Loads

CHAPTER

6

Constraint and Set Notation

- Degree-of-Freedom Sets
- Multipoint Constraints
- Single Point Constraints
- Rigid Body Supports
- Sets for Dynamic Reduction
- Sets for Aerodynamics
- Rules of Sets for Undefined Degrees-of-Freedom
- Output Selection Via Set Specification

6.1 Degree-of-Freedom Sets

Structural matrices are initially assembled in terms of all structural points, which excludes only the extra points introduced for dynamic analysis. These matrices are generated with six degrees-of-freedom for each geometric grid point and a single degree-of-freedom for each scalar point. These degrees-of-freedom are partitioned into sets, based on user inputs. The sets are used to successively eliminate variables during the solution process, as described in “[General Operations](#)” on page 374.

6.1.1 Degree-of-Freedom Set Definitions

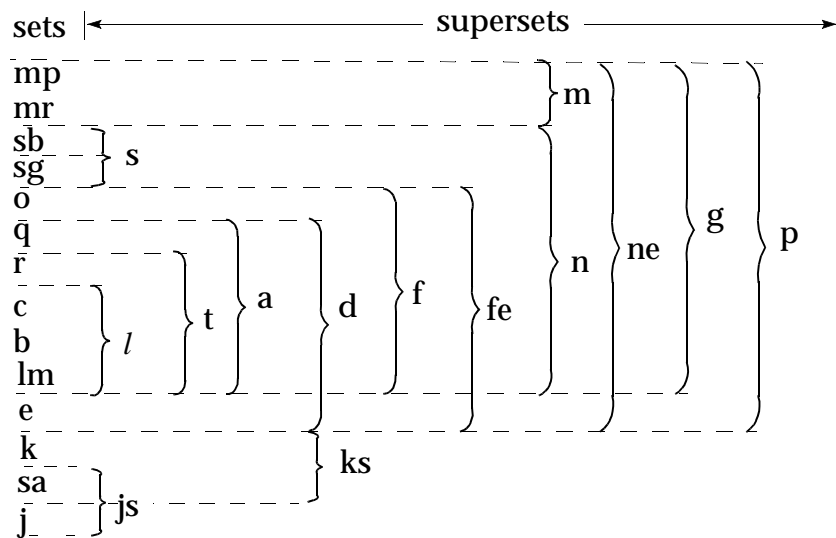
Each degree-of-freedom is a member of one mutually exclusive set. Set names have the following definitions:

Set Name	Definition
mp	Degrees-of-freedom eliminated by <u>m</u> ultipoint constraints.
mr	Degrees-of-freedom eliminated by <u>m</u> ultipoint constraints created by the <u>r</u> igid elements using the LGELIM method on the Case Control command RIGID.
sb*	Degrees-of-freedom eliminated by <u>s</u> ingle-point constraints that are included in <u>b</u> oundary condition changes and by the AUTOSPC feature.
sg*	Degrees-of-freedom eliminated by <u>s</u> ingle-point constraints that are specified on the PS field on <u>G</u> RID Bulk Data entries.
o	Degrees-of-freedom <u>o</u> mitted by structural matrix partitioning.
q	Generalized degrees-of-freedom assigned to component modes and residual vectors.
r	<u>r</u> eference degrees-of-freedom used to determine free body motion.
c	Degrees-of-freedom that are free during component mode synthesis or dynamic reduction.
b	Degrees-of-freedom fixed during component mode analysis or dynamic reduction.
lm	<u>L</u> agrange <u>m</u> ultiplier degrees-of-freedom created by the rigid elements using the LAGR method on the Case Control command, RIGID.
e	<u>e</u> xtra degrees-of-freedom introduced in dynamic analysis.
sa	Permanently constrained aerodynamic degrees-of-freedom.

Set Name	Definition
k	Aerodynamic mesh point set for forces and displacements on the aero mesh.
j	Aerodynamic mesh collocation point set (exact physical interpretation is dependent on the aerodynamic theory).

*Strictly speaking, sb and sg are not exclusive with respect to one another. Degrees-of-freedom may exist in both sets simultaneously. Since these sets are not used explicitly in the solution sequences, this need not concern the user. However, those who use these sets in their own DMAPs should avoid redundant specifications when using these sets for partitioning or merging operations. That is, a degree-of-freedom should not be specified on both a PS field of a GRID entry (sg set) and on a selected SPC entry (sb set). Redundant specifications will cause UFM 2120 in the VEC module and behavior listed in *MSC.Nastran 2005 DMAP Programmer's Guide* for the UPARTN module. These sets are exclusive, however, from the other mutually exclusive sets.

Each degree-of-freedom is also a member of one or more combined sets called "supersets." Supersets have the following definitions:



Set Name	Meaning (+ indicates union of two sets)
$s = sb + sg$	all degrees-of-freedom eliminated by <u>s</u> ingle point constraints
$l = b + c + lm$	the structural degrees-of-freedom remaining after the reference degrees-of-freedom are removed (degrees-of-freedom <u>l</u> eft over)
$t = l + r$	the <u>t</u> otal set of physical boundary degrees-of-freedom for superelements
$a = t + q$	the set <u>a</u> ssembled in superelement analysis
$d = a + e$	the set used in dynamic analysis by the <u>d</u> irect method
$f = a + o$	unconstrained (<u>f</u> ree) structural degrees-of-freedom
$fe = f + e$	<u>f</u> ree structural degrees-of-freedom plus <u>e</u> xtra degrees-of-freedom
$n = f + s$	all structural degrees-of-freedom <u>n</u> ot constrained by multipoint constraints
$ne = n + e$	all structural degrees-of-freedom <u>n</u> ot constrained by multipoint constraints plus <u>e</u> xtra degrees-of-freedom
$m = mp + mr$	all degrees-of-freedom eliminated by <u>m</u> ultiple constraints
$g = n + m$	all structural (<u>g</u> rid) degrees-of-freedom including scalar degrees-of-freedom
$p = g + e$	all <u>p</u> hysical degrees-of-freedom
$ks = k + sa$	the union of k and the re-used s-set (6 dof per grid)
$js = j + sa$	the union of j and the re-used s-set (6 dof per grid)
$fr = o + l$	statically independent set minus the statically determinate supports ($fr = f - q - r$)
$v = o + c + r$	the set free to <u>v</u> ibrate in dynamic reduction and component mode synthesis

The a-set and o-set are created in the following ways:

1. If only OMITi entries are present, then the o-set consists of degrees-of-freedom listed explicitly on OMITi entries. The remaining f-set degrees-of-freedom are placed in the b-set, which is a subset of the a-set.

2. If ASETi or QSETi entries are present, then the a-set consists of all degrees-of-freedom listed on ASETi entries and any entries listing its subsets, such as QSETi, SUPORTi, CSETi, and BSETi entries. Any OMITi entries are redundant. The remaining f-set degrees-of-freedom are placed in the o-set.
3. If there are no ASETi, QSETi, or OMITi entries present but there are SUPORTi, BSETi, or CSETi entries present, then the entire f-set is placed in the a-set and the o-set is not created.
4. There must be at least one explicit ASETi, QSETi, or OMITi entry for the o-set to exist, even if the ASETi, QSETi, or OMITi entry is redundant.

In dynamic analysis, additional vector sets are obtained by a modal transformation derived from real eigenvalue analysis of the a-set. These sets are as follows:

ξ_o = rigid body (zero frequency) modal degrees-of-freedom

ξ_f = finite frequency modal degrees-of-freedom

ξ_i = $\xi_o + \xi_f$, the set of all modal degrees-of-freedom

One vector set is defined that combines physical and modal degrees-of-freedom:

u_h = $\xi_i + u_e$, the set of all modal degrees-of-freedom

The membership of each degree-of-freedom can be printed by use of the Bulk Data entries PARAM,USETPRT and PARAM,USETSEL.

6.1.2 Degree-of-Freedom Set Bulk Data Entries

Degrees-of-freedom are placed in sets as specified by the user on the following Bulk Data entries:

Name	Bulk Data Entry Name
m	MPC, MPCADD, MPCAX, POINTAX, RBAR, RBE1, RBE2, RBE3, RROD, RSPLINE, RTRPLT, GMBC, GMSPC*
sb	SPC, SPC1, SPCADD, SPCAX, FLSYM, GMSPC*, BNDGRID, (PARAM,AUTOSPC,YES)
sg	GRID, GRIDB, GRDSET (PS field)
o	OMIT, OMIT1, OMITAX, GRID (SEID field), SESET
q	QSET, QSET1

Name	Bulk Data Entry Name
r	SUPPORT, SUPPORT1, SUPAX
c	CSET, CSET1
b	BSET, BSET1
e	EPOINT
sa	CAEROi
k	CAEROi
a	ASET, ASET1, Superelement exterior degrees-of-freedom, CSUPEXT

*Placed in set only if constraints are not specified in the basic coordinate system.

In superelement analysis, the appropriate entry names are preceded by the letters SE, and have a field reserved for the superelement identification number. This identification is used because a boundary (exterior) grid point may be in one mutually exclusive set in one superelement and in a different set in the adjoining superelement. The SE-type entries are internally translated to the following types of entry for the referenced superelement:

Entry Type	Equivalent Type
SEQSETi	QSETi
SESUP	SUPPORT
SECSETi	CSETi
SEBSETi	BSETi

6.2 Multipoint Constraints

Each multipoint constraint is described by a single equation that specifies a linear relationship for two or more degrees-of-freedom. In static analysis, multiple sets of multipoint constraints can be provided in the Bulk Data Section, with selections made at execution time by using the subcase structure in Case Control as explained in “[Case Control Section](#)” on page 11. Multipoint constraints are also discussed in “[Executive Control Section](#)” on page 10 of this manual and Section 5.4 of *The NASTRAN Theoretical Manual*.

Multipoint constraints are defined on MPC, MPCADD, and MPCAX entries. MPC is the basic entry for defining multipoint constraints. The first component specified on the entry is the dependent degree-of-freedom, i.e., that degree-of-freedom that is removed from the equations of motion. Dependent degrees-of-freedom may appear as independent terms in other equations of the set; however, they may appear as dependent terms in only a single equation. The MPCADD entry defines a union of multipoint constraints. The MPCAX defines multipoint constraints in conical shell problems. Some uses of multipoint constraints are:

- To describe rigid elements and mechanisms such as levers, pulleys, and gear trains. In this application, the degrees-of-freedom associated with the rigid element that are in excess of those needed to describe rigid body motion are eliminated with multipoint constraint equations. Treating very stiff members as rigid elements eliminates the ill-conditioning associated with their treatment as ordinary elastic elements.
- To enforce zero motion in directions other than those corresponding with components of the global coordinate system. In this case, the multipoint constraint will involve only the degrees-of-freedom at a single grid point. The constraint equation relates the displacement in the direction of zero motion to the displacement components in the global system at the grid point.
- To be used with scalar elements to generate nonstandard structural elements and other special effects.
- To describe parts of a structure by local vibration modes. This application is treated in Section 14.1 of *The NASTRAN Theoretical Manual*. The general idea is that the matrix of local eigenvectors represents a set of constraints relating physical coordinates to modal coordinates.

In general, the user must provide the coefficients in the multipoint constraint equations. However, several rigid elements have been introduced which will generate the MPC equations for some applications:

RROD	A pin-ended rod which is rigid in extension.
RBAR	A rigid bending element with six degrees-of-freedom at each end.
RTRPLT	A rigid triangular plate with six degrees-of-freedom at each vertex.
RBE1	A rigid body connected to an arbitrary number of grid points. The independent and dependent degrees-of-freedom can be arbitrarily selected by the user.
RBE2	A rigid body connected to an arbitrary number of grid points. The independent degrees-of-freedom are the six components of motion at a single grid point. The dependent degrees-of-freedom are specified at an arbitrary number of grid points.
RBE3	Defines the motion at a reference grid point as the weighted average of the motions at a set of other grid points.
RSPLINE	Defines multipoint constraints for the interpolation of displacements at grid points.
RSSCON	Defines multipoint constraints to model clamped connections of shell-to-solid elements.

The rigid elements will always meet equilibrium and continuity requirements, whereas this is a user responsibility for MPC equations. (See “**Rigid Elements and Multipoint Constraints (R-type, MPC)**” on page 167 for further information on rigid elements.)

Multipoint forces of constraint may be output with MPCFORCE Case Control command in SOLution Sequences 101 through 200.

6.3 Single Point Constraints

A single point constraint (SPC) applies a fixed value to a translational or rotational component at a geometric grid point or to a scalar point. Common uses of single point constraints are to specify the boundary conditions of a structural model by fixing the appropriate degrees-of-freedom and to eliminate unwanted degrees-of-freedom with zero stiffness. Multiple sets of single point constraints can be provided in the Bulk Data Section, with selections made at execution time by using the subcase structure in the Case Control Section as explained in “[Case Control Section](#)” on page 11. This procedure is particularly useful in the solution of problems having one or more planes of symmetry.

The elements connected to a grid point may not provide resistance to motion in certain directions, causing the stiffness matrix to be singular. Single point constraints are used to remove these degrees-of-freedom from the stiffness matrix. A typical example is a planar structure composed of membrane and extensional elements. The translations normal to the plane and all three rotational degrees-of-freedom must be constrained since the corresponding stiffness matrix terms are all zero. If a grid point has a direction of zero stiffness, the single point constraint needs not be exactly in that direction, but only needs to have a component in that direction. This allows the use of single point constraints for the removal of such singularities regardless of the orientation of the global coordinate system. Although the displacements will depend on the direction of the constraint, the internal forces will be unaffected.

One of the tasks performed by the Grid Point Singularity Processor (GPSP) (see “[Constraint and Mechanism Problem Identification in SubDMAP SEKR](#)” on page 383) is to examine the stiffness matrix for singularities at the grid point level. Singularities remaining at this level, following the application of the multipoint and single point constraints, are listed in the Grid Point Singularity Table (GPST), which is automatically printed. The GPST lists all singular degrees-of-freedom, in the global coordinate system, and the ratio of stiffness between the softest and stiffest degree-of-freedom for the grid point. The user may request that single point constraints be generated for all identified singularities by use of the Bulk Data entry `PARAM,AUTOSPC,YES`.

Single point constraints are defined on `SPC`, `SPC1`, `SPCADD` and `SPCAX` entries. The `SPC` entry is the most general way of specifying single point constraints. The `SPC1` entry is a less general entry that is more convenient when several grid points have the same components constrained to a zero displacement. The `SPCADD` entry defines a union of single point constraint sets specified with `SPC` or `SPC1` entries. The `SPCAX` entry is used only for specifying single point constraints in problems using conical shell elements.

Single point constraints can also be defined on the GRID entry. In this case, however, the points are constrained for all subcases. The default value for enforced displacement on points constrained on GRID entries is zero. The default value can be overridden at the subcase level with SPC entries.

The printed output for single point forces of constraints can be requested in Case Control. All nonzero forces are printed, whether they originate from SPCi entries, the PS field on GRID entries, or by PARAM,AUTOSPC,YES.

6.4 Rigid Body Supports

In the following discussion, a free body is defined as a structure that is capable of motion without internal stress; i.e., it has one or more rigid body degrees-of-freedom. The stiffness matrix for a free body is singular with the defect equal to the number of stress-free, or rigid body modes. A solid three-dimensional body has up to six rigid body modes. Linkages and mechanisms can have a greater number. In order to permit the analysis of mechanisms, no restriction is placed in the program on the number of stress-free modes.

Free-body supports are defined with a SUPORT or SUPORT1 entry. In the case of problems using conical shell elements, the SUPAX entry is used. Free-body supports must be defined in the global coordinate system. The SUPORT1 entry must be selected by the SUPORT1 Case Control command.

In static analysis by the displacement method, the rigid body modes must be restrained in order to remove the singularity of the stiffness matrix. The required constraints may be supplied with single point constraints, multipoint constraints, or free body supports. If free body supports are used, the rigid body characteristics will be calculated and a check will be made on the sufficiency of the supports. Such a check is obtained by calculating the rigid body error ratio and the strain energy as defined in the Rigid Body Matrix Generator operation in “[Static Solutions in SubDMAP SEKRRS](#)” on page 391. This error ratio and the strain energy are automatically printed following the execution of the Rigid Body Matrix Generator. The error ratio and the strain energy should be zero, but may be nonzero for any of the following reasons:

- Round-off error accumulation.
- Insufficient free body supports have been provided.
- Redundant free body supports have been provided.

The redundancy of the supports may be caused by improper use of the free body supports themselves or by the presence of single point or multipoint constraints that constrain the rigid body motions.

Static analysis with inertia relief is necessarily made on a model having at least one rigid body motion. Such rigid body motion must be constrained by the use of free body supports. These supported degrees-of-freedom define a reference system, and the elastic displacements are calculated relative to the motion of the support points. The element stresses and forces will be independent of any valid set of supports.

Rigid body vibration modes are calculated by a separate procedure provided that a set of free body supports is supplied by the user. This is done to improve efficiency and, in some cases, reliability. The determinant method, for example, has difficulty extracting zero frequency roots of high multiplicity, whereas the alternate procedure of extracting rigid body modes is both efficient and reliable. If the user does not specify free body supports (or he specifies an insufficient number of them), the (remaining) rigid body modes will be calculated by the method selected for the finite frequency modes, provided zero frequency is included in the range of interest. If the user does not provide free body supports, and if zero frequency is not included in the range of interest, the rigid body modes will not be calculated.

Free body supports must be specified if the mode acceleration method of solution improvement is used for dynamic problems having rigid body degrees-of-freedom (see “**Rigid-body Modes**” on page 87 of the *MSC.Nastran Basic Dynamics User’s Guide*). This solution improvement technique involves a static solution, and although the dynamic solution can be made on a free body, the static solution cannot be performed without removing the singularities in the stiffness matrix associated with the rigid body motions.

6.5 Sets for Dynamic Reduction

There are several methods for reducing the size of models in dynamic analysis. The method described here uses shapes derived as generalized functions by static analysis. The same entries may be used in static analysis with the same nomenclature being used internally as in superelement analysis.

The statically independent degrees-of-freedom remaining after constraint elimination are partitioned into two sets. The partitions are defined by listing the degrees-of-freedom for one of the partitions on the ASETi entries. These degrees-of-freedom are referred to as the “analysis set.” The remaining degrees-of-freedom are referred to as the “omitted set.” The OMITi entry can be used to place degrees-of-freedom in the omitted set with the remaining degrees-of-freedom being placed in the analysis set. This is easier if the analysis set is large. In the case of problems using conical shell elements, the OMITAX entry is used.

One of the more important applications of partitioning is Guyan reduction, described in “**Dynamic Reduction**” on page 313 of the *MSC.Nastran Basic Dynamics User’s Guide*. This technique is a means for reducing the number of degrees-of-freedom used in dynamic analysis with minimum loss of accuracy. Its basis is that many fewer grid points are needed to describe the inertia of a structure than are needed to describe its elasticity with comparable accuracy. The error in the approximation is small provided that the set of displacements used for dynamic analysis is judiciously chosen. Its members should be uniformly dispersed throughout the structure, and all large mass items should be connected to grid points that are members of the analysis set.

The user is cautioned to consider the fact that the matrix operations associated with this partitioning procedure tend to create nonzero terms and to fill what were previously very sparse matrices. The partitioning option is most effectively used if the members of the omitted set are either a very large fraction or a very small fraction of the total set. In most of the applications, the omitted set is a large fraction of the total and the matrices used for analysis, while small, are usually full. If the analysis set is not a small fraction of the total, a solution using the large, but sparser, matrices may be more efficient.

The a-set is further partitioned into the q-, r-, c-, and b-sets. The q-set is used to store the coefficients for the generalized coordinates determined by dynamic reduction and/or component mode calculations. The r-set is again used to determine rigid body modes. The c-set contains coordinates considered free to move while computing the generalized coordinate functions. The b-set contains coordinates fixed during this process.

The set used to describe the generalized functions for dynamic reduction and component mode synthesis is the v-set, the union of the o-, r-, and c-sets.

The d-set (“dynamic”) for direct formulations is formed from the union of the structural degrees-of-freedom in the a-set and the extra points of the e-set. The d-set can be used to input unsymmetric terms into the equations of motion using DMIG and TF Bulk Data entries. In the modal formulations, the h-set is composed of the union of the c_i (modal) set and the e-set. Again, the DMIG and TF Bulk Data entries are a means to introduce unsymmetric matrix coefficients.

6.6 Sets for Aerodynamics

Aerodynamic calculations are made in a Cartesian aerodynamic coordinate system. By convention, the flow is in the positive x direction. The basic coordinate system was not chosen, since it would place an undesirable restriction upon the description of the structural model. Any MSC.Nastran Cartesian system may be specified, and flow will be defined in the direction of its positive x -axis. All element and aerodynamic grid point data, computed initially in the basic coordinate system, will be converted to the aerodynamic coordinate system. The global (displacement) coordinate system of the aerodynamic grid points will have its T1 direction in the flow direction. T3 is normal to the element for boxes, and parallel to the aerodynamic T3 in the case of bodies. Coordinate system data are generated for the aerodynamic grid points.

The grid points are physically located at the centers of the boxes and body elements. Permanent constraints are generated for the unused degrees-of-freedom. A second set of grid points, used only for undeformed plotting, is located at the element corners. All six degrees-of-freedom associated with each grid point in this second set are permanently constrained. Grid point numbers are generated based upon the element identification number. For any panel, the external grid point numbers for the boxes start with the panel identification number and increase consecutively.

Aerodynamic degrees-of-freedom, along with the extra points, are added after the structural matrices and modes have been determined. This introduces the following displacement sets:

k	Aerodynamic box and body degrees-of-freedom
sa	Permanently constrained degrees-of-freedom associated with aerodynamic grid points
$ks = k + sa$	the union of k and the re-used s -set (6 dof per grid)
$js = j + sa$	the union of j and the re-used s -set (6 dof per grid)

6.7 Rules of Sets for Undefined Degrees-of-Freedom

Most of the Bulk Data entries used to list set membership specify mutually exclusive sets. This means that degrees-of-freedom listed on more than one such entry will result in a fatal error. For example, a degree-of-freedom may not be in the m-set on two RBAR entries, or on a selected SPC entry and a SUPORT entry.

The a-set, by contrast, defines a combined set. Degrees-of-freedom listed on Bulk Data entries that define its subsets, the q-, r-, c-, and b-set, may also be listed on ASETi Bulk Data entries, although this is a redundant specification and is not necessary. However, automatic restarts are more reliable if ASETi entries are present for all degrees-of-freedom listed on SUPORT entries.

Bulk Data entries used to define mutually exclusive sets automatically place them in their specified set and any combined set to which they belong. Since the ASETi entries define a combined set, the subsets of the a-set, if not defined, must be placed in a mutually exclusive set by an arbitrary convention. The conventions described below are used because they allow the user to specify these sets with the minimum input for most cases. In superelement analysis, exterior (“boundary”) degrees-of-freedom are automatically determined and placed in the a-set of the superelement. Exterior degrees-of-freedom for superelements are therefore equivalent to degrees-of-freedom listed on ASETi entries in the discussion below.

The f-set is the union of the mutually exclusive o-, q-, r-, c-, and b-sets and includes the a-set. The o-set exists under two conditions only:

1. If OMITi entries are present, the o-set consists of degrees-of-freedom listed explicitly on OMITi entries. The remaining f-set degrees-of-freedom are placed in the b-set. They are then also members of the a-set.
2. If ASETi or QSETi entries are present, the a-set consists of all degrees-of-freedom listed on ASETi entries and any entries listing its subsets, such as QSETi, SUPORTi, CSETi, and BSETi entries. The remaining f-set degrees-of-freedom are placed in the o-set. Note, however, there must be at least one explicit ASETi or QSETi entry type present for the o-set to exist, even if it is a redundant specification. If there are no ASETi or QSETi entries present, and SUPORT, BSETi, or CSETi are present, the entire f-set is placed in the a-set; that is, the o-set is not formed.

If both OMITi and ASETi entries are present, the OMITi entries are ignored, so that condition 2, above, applies. Similarly, if ASETi entries are used in conjunction with entries defining its subsets, undefined degrees-of-freedom in the a-set are placed in mutually exclusive sets as shown in the following table (0 = no entry types present; 1 = entry type present):

Set Defined by Entry Type	Case			
	1	2	3	4
c	0	0	1	1
b	0	1	0	1
Undefined a-set placed in	b	c	b	b

6.8 Output Selection Via Set Specification

The membership of grid points and their components and scalar points in the various sets is stored in the USET table as digits in a binary number. Some modules use input parameters that are translations of these binary numbers as a shorthand notation. Three formats are used:

Set name

(DEFUSET, USET and USET1 Bulk Data entries, MATGPR, UPARTN, VEC, UMERGE, UMERGE1, and PARAML (OP = USET) Modules)

The name of the set from [Table 6-1](#) (which may be redefined by the DEFUSET entry).

Decimal equivalent

(TABPRT, MATMOD (Option 17), and MATGEN (Option 11) Modules)

The decimal equivalent of the set is listed below. Several sets can be described by a single number that is the sum of the listed equivalents. For example, to print the m and s sets, the value of 3(1+2) is input to the TABPRT module. The value of -1 will cause the output of all sets.

Bit position

(VEC Module)

The location of the digit corresponding to the set in the binary number, starting with the left-most digit, can be used to identify one set, as listed in [Table 6-1](#).

The decimal equivalent and bit position described in [Table 6-1](#) are only valid for pre-V70.7 MSC.Nastran. For V70.7 and after, sets are defined by their name. For example, if the user wishes the TABPRT module to print the M, R, N, SG, K, and SA sets, then define

```
PARAM,USETSTR1, M:R:N:SG
```

```
PARAM,USETSTR2, K:SA
```

Table 6-1 Valid only for pre-V70.7 MSC.Nastran

Set Name	Decimal Equivalent	Bit Position
U1	--	1
U2	--	2
U3	--	3

Table 6-1 (continued) Valid only for pre-V70.7 MSC.Nastran

Set Name	Decimal Equivalent	Bit Position
U4	--	4
U5	--	5
U6	--	6
V	33554432	7
FR	16777216	8
T	8388608	9
O	4194304	10
B	2097152	11
C	1048576	12
PA	524288	13
K	262144	14
SA	131072	15
PS	65536	16
D	32768	17
FE	16384	18
NE	8192	19
P	4096	20
E	2048	21
SB	1024	22
SG	512	23
L	256	24
A	128	25
F	64	26
N	32	27
G	16	28
R	8	29
O	4	30

Table 6-1 (continued) Valid only for pre-V70.7 MSC.Nastran

Set Name	Decimal Equivalent	Bit Position
S	2	31
M	1	32
H,J,...	--	--

The data block naming convention in the solution sequences for matrices related to the sets is based on the set names. For example, the system stiffness matrix $[K_{gg}]$ has the name KGG. Some sets are used that do not appear in the USET table. The h-set, for example, is the set used for modal formulations. Its size is not known until after the USET table is formed, so it does not appear in the table. The j-set, a set synonymous with the g-set, is used in superelement analysis. There are several other sets used that also do not appear in the USET table.

Some names may be used for other purposes. The name H, for example, may be input to the MATGPR module when one wants to label rows or columns of matrices sequentially, rather than by their external sequence numbers. This feature may be used to number vectors in load matrices or in other instances not related to modal analysis, even though "H" is defined for modal analysis.

CHAPTER

7

Solution Sequences

- Introduction
- Output Description
- Restart Procedures
- General Operations
- Summary of Solution Sequence Operations

7.1 Introduction

MSC.Nastran is composed of a large number of building blocks called modules. A module is a collection of FORTRAN subroutines designed to perform a specific task—processing model geometry, assembling matrices, applying constraints, solving matrices, calculating output quantities, conversing with the database, printing the solution, and so on. The modules are controlled by an internal language called the Direct Matrix Abstraction Program (DMAP). Each type of analysis available in MSC.Nastran is called a solution sequence, and each solution sequence is a prepackaged collection of hundreds or thousands of DMAP commands. Once a solution sequence is chosen, its particular set of DMAP commands sends instructions to the modules that are needed to perform the requested solution. All of this happens automatically with no effort on your part beyond choosing a solution sequence.

The solution sequences contain complete DMAP sequences required to perform certain types of analysis as well as superelement generation (phase I) and data recovery (phase III). The solution sequences are listed in [Table 7-1](#). An automatic restart capability is employed in these sequences. The automatic restart is not only driven by modifications to the Case Control and Bulk Data Sections but also by modifications in upstream superelements. A full description of automatic restarts is given in “[Restart Procedures](#)” on page 372.

A solution is a feature selected by a SOL Executive Control command. It is a collection of DMAP statements that performs a specific type of analysis. For example, static analysis is requested by the command SOL 101. There are two types of solutions at present.

The major type is called a Solution Sequence. They are called with numbers ranging from 100 to 200. There were formerly Structured Solution Sequences and Unstructured Solution Sequences, but the latter class has been removed, so the name Solution Sequence is now used, although you may find references to Structured Solutions in older documentation. All new technology is introduced through Solution Sequences. They also have automatic restart capability. This is accomplished with the NASTRAN Data Definition Language feature, a file that lists the names and attributes of data types saved for restart. The evidence that a Solution Sequence has restart capability can be found by the existence of TYPE statements in its DMAP compile listings. It uses the SubDMAP feature, which allow breaking a large collection of DMAP statements into more manageable data structures similar to subroutines in other programming languages.

The other, more minor type is called a Rigid Format. They are called with SOL numbers ranging from 1 through 15. They are based on the type of solution available prior to Version 66. These solutions do not have capabilities introduced after Version 66, and use obsolete DMAP commands, such as the COND statement for conditional transfers. They do not support automatic restart or superelement analysis, nor many other features. They may be useful for writing prototype DMAP alters or learning how to write DMAP alters because they do not have all of the features of the solution sequence, or the SubDMAP organization. They may be useful for analysis that do not use the new capabilities. They have one minor initializing SubDMAP that reads the input files, with all other operations in one large SubDMAP. This gives more visibility to the data flow, and less complexity (and the convenience of SubDMAPs) than the Solution Sequences. They do not have TYPE statements built in, but these may be added with DMAP alters.

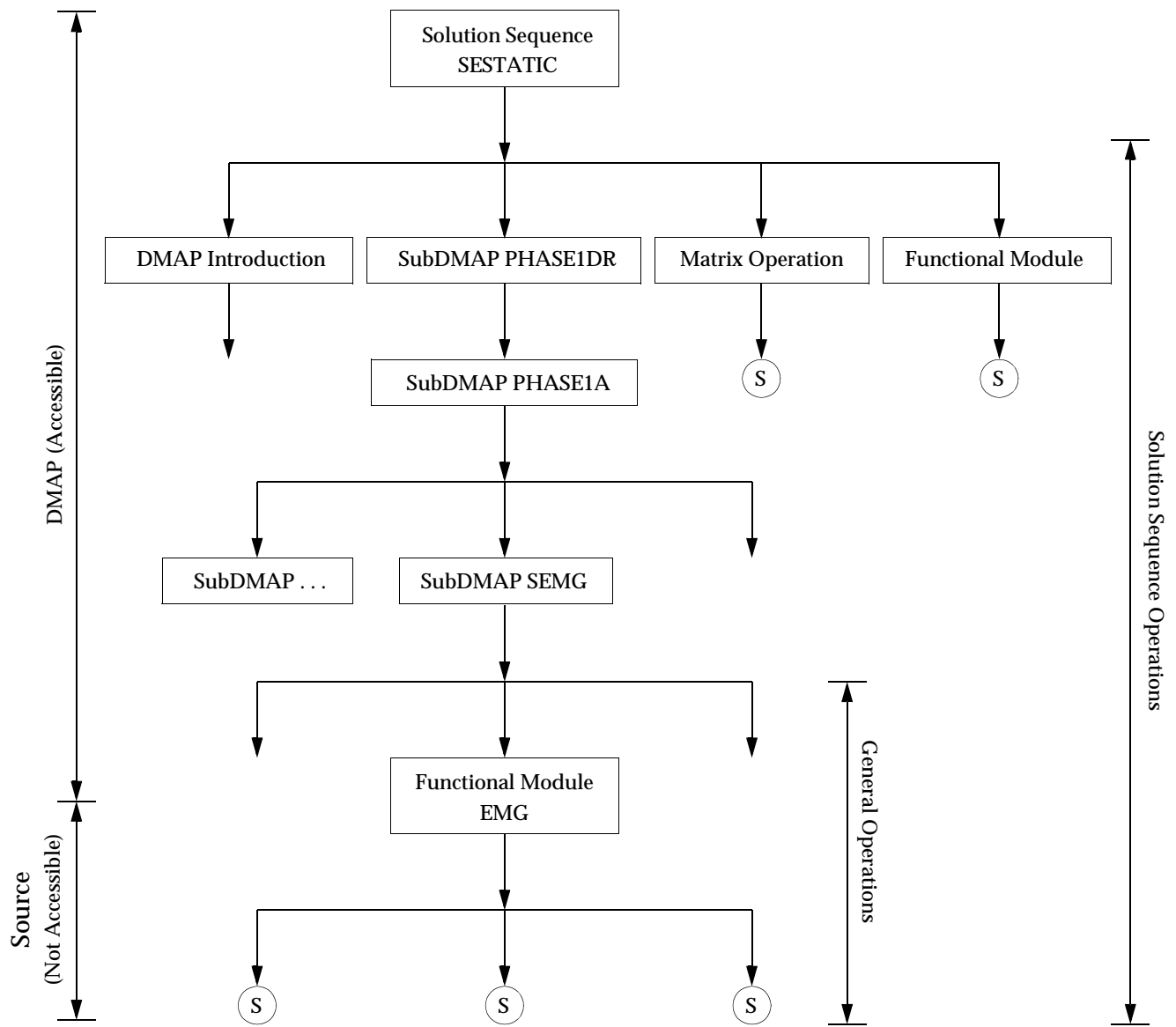


Figure 7-1 Hierarchy in the Structural Solution Sequences

Table 7-1 Structured Solution Sequences

SOL Number	SOL Name	Description	References
101	SESTATIC	Statics with options: Linear Heat Transfer Alternate Reduction Inertia Relief Design Sensitivity -- Statics	page 391 page 442
103	SEMODES	Normal Modes with option: Design Sensitivity -- Modes	page 423
105	SEBUCKL	Buckling with options: Static Analysis Design Sensitivity -- Buckling	page 442
106	NLSTATIC	Nonlinear Statics	page 568
107	SEDCEIG	Direct Complex Eigenvalues	page 440
108	SEDFREQ	Direct Frequency Response	page 441
109	SEDTRAN	Direct Transient Response	page 441
110	SEMCEIG	Modal Complex Eigenvalues	page 440
111	SEMFREQ	Modal Frequency Response	page 441
112	SEMTRAN	Modal Transient Response	page 441
114	CYCSTAT X	Cyclic Statics with option: Alternate Reduction	page 803
115	CYCMODE	Cyclic Normal Modes	page 803
118	CYCFREQ	Cyclic Direct Frequency Response	page 803
129	NLTRAN	Nonlinear Transient Response	page 568
144	AESTAT	Static Aeroelastic Response	See the <i>MSC.Nastran Aeroelastic Analysis User's Guide</i>
145	SEFLUTTR	Aerodynamic Flutter	See the <i>MSC.Nastran Aeroelastic Analysis User's Guide</i>
146	SEAERO	Aeroelastic Response	See the <i>MSC.Nastran Aeroelastic Analysis User's Guide</i>

Table 7-1 Structured Solution Sequences (continued)

SOL Number	SOL Name	Description	References
153	NLSCSH	Static Structural and/or Steady State Heat Transfer Analysis	page 460
159	NLTCSH	Transient Structural and/or Transient Heat Transfer Analysis	page 460
190	DBTRANS	Database Transfer	page 487
200	DESOPT	Design Optimization	See the <i>MSC.Nastran Design Sensitivity and Optimization User's Guide</i>
400	NONLIN	Nonlinear Static and Transient Analysis	page 721
600,ID		MSC.Nastran, MSC.Marc in Nastran	page 737

The DMAP listings for any of the solutions sequences can be obtained by specifying a COMPILER LIST statement in the Executive Control Section. A description of the data processing and matrix operations for the solution sequences is given in “[Summary of Solution Sequence Operations](#)” on page 415. The complete details for each of the functional modules are given in Chapter 4 of the *MSC.Nastran Programmer's Manual*. Those DMAP modules which are of general use (matrix operations and utility modules) are described along with examples of their use in “[General Operations](#)” on page 374.

If the user wishes to modify a DMAP sequence in some manner, he can use the alter feature described in *MSC.Nastran 2005 DMAP Programmer's Guide*. Typical uses are to schedule an EXIT prior to completion in order to check intermediate output, schedule the printing of a table or matrix for diagnostic purposes, and to delete or add a functional module to the DMAP sequence. The user should be familiar with the rules for DMAP programming, as described in the *MSC.Nastran 2005 DMAP Programmer's Guide*, to make alterations to a solution sequence. A library of DMAP alters is delivered with MSC.Nastran. These alters are used to request options not available in the solution sequences and for exceptional operations that do not fit in with the normal flow of the solution sequences.

7.2 Output Description

Although most of the solution sequence output is optional and requested in the Case Control Section, some of the printer output is automatic or under control of DIAG statements in the Executive Control Section. The printer output is designed for 132 characters per line, with the lines per page controlled by the LINE command in the Case Control Section. The default is LINE = 50 for 11-inch paper. Optional titles are printed at the top of each page from information in the Case Control Section. These titles may be defined at the subcase level. The pages are automatically dated and numbered.

The output from data recovery and plot modules is all optional and its selection is controlled by commands in the Case Control Section. The details of making selections in the Case Control Section are described in “[Case Control Section](#)” on page 11 for printer and punch output and in “[Plotting](#)” on page 501 for plotter output. The available output for each solution sequence is described in “[Summary of Solution Sequence Operations](#)” on page 415. Detailed information on the force and stress output available for each element type is given in “[Structural Elements](#)” on page 47.

A few printer output items are under the control of PARAM Bulk Data entries. The use of PARAM entries is described in “[Use of Parameters](#)” on page 35. The DIAG statement is used to control the printing of diagnostic output. The available output under the DIAG statement is given in the description of the Executive Control statements in the “[Executive Control Statements](#)” on page 99 of the *MSC.Nastran Quick Reference Guide*.

The first part of the output for an MSC.Nastran run is prepared during the execution of the preface, prior to the beginning of the solution sequence. The following output is either automatically or optionally provided during execution:

- MSC.Nastran title page -- automatic
- File Management Section echo -- automatic if statements are present
- Executive Control Section echo -- automatic
- Case Control Section echo -- automatic
- Unsorted Bulk Data Section echo -- optional, selected in the Case Control Section
- Sorted Bulk Data Section echo -- automatic, unless suppressed in the Case Control Section

MSC.Nastran Output

MSC.Nastran generates the following output files.

File	Contents
Execution Summary Table in *.f04	Computer manufacturer's model, operating system, MSC.Nastran version, system creation date, problem run date, MSC.Nastran file assignments, and a summary of the MSC.Nastran execution: time of day, elapsed time, I/O seconds, CPU seconds, module execution messages, matrix trailers (DIAG 8), value of REAL (DIAG 13), MPYAD, SDCOMP and FBS estimates (DIAG 19), and statistical information on module expected/actual CPU item (DIAG 49).
Standard Output in *.f06	MSC.Nastran results (i.e., displacements, stresses, etc.). See the <i>MSC.Nastran Linear Static Analysis User's Guide</i> and the <i>MSC.Nastran Numerical Methods User's Guide</i> for more information.

An example follows which shows the execution summary (.f04) and the standard output (.f06) for SOL 101.

Execution Summary Contents (see [Listing 7-1](#))

The information in the Execution Summary file can provide the user with CPU time, I/O time, elapsed time and disk space resource usage information on the execution of the problem. To aid in the interpretation of these data, an example is presented on the following pages. Circled integers have been added for ease of reference.

- ① The column headed DAY TIME indicates the time of day as the run progresses.
- ② The column headed ELAPSED indicates the number of elapsed "wall clock" minutes:seconds at the beginning on the job. This number may vary greatly depending on the system load in a time-sharing environment.
- ③ The column headed I/O SEC is an internally generated measure of I/O. This measure is a count of the number of BUFFSIZE blocks transferred, divided by the value of SYSTEM(84) (the estimated rate of number of blocks moved per second). This number does not include I/O for loading the executable file or I/O for FORTRAN files. The column headed DEL I/O is the I/O seconds spent in the previous DMAP statement.

- ④ The column headed CPU SEC shows the amount of CPU seconds that elapsed when this line was printed. It is recommended that the user examine these entries and look for large jumps in this number and try to understand why each one occurred. For example, the cost for “formatting” the output (modules SDR2 and OFP) may be greater than the cost of generating the numbers. In such situations, simple techniques such as reducing output requests may save more money than other more well-known techniques such as sequencing, OMITs, better matrix methods, etc. In short, the user should be aware of the expensive operations on each large run. The column headed DEL CPU is the CPU time spent in the previous DMAP statement.
- ⑤ Primary database file assignments.
- ⑥ The column headed BUFFSIZE is the buffersize of each DBset. This buffersize may be defined by the INIT command for each DBset (see the “[File Management Statements](#)” on page 31 of the *MSC.Nastran Quick Reference Guide*).
- ⑦ The column headed CLUSTER SIZE is the unit of allocation for GINO and executive blocks. This unit may be defined on the INIT command (see the “[File Management Statements](#)” on page 31 of the *MSC.Nastran Quick Reference Guide*).
- ⑧ The column headed TIME STAMP is the time the file was created. The format is YYMMDDHHMMSS.
- ⑨ The delivery database file assignments. These files contain the Solution Sequences and timing constants.
- ⑩ Indicates the FMS statement INIT MASTER (RAM) has been invoked (by default).
- ⑪ SUBDMAP name.
- ⑫ DMAP module name or subDMAP name.
- ⑬ The line REAL CORE IS XXX shows the value of the NASTRAN keyword REAL. This message is generated by using the DIAG 13 command in the Executive Control Section of the MSC.Nastran data file. It only appears after module executions which use the value. The open core (i.e., working space) for other modules is set via HICORE (see the [MSC.Nastran 2004 Installation and Operations Guide](#)).

- ⑭ The number that precedes the module name on lines that terminate with BEGN or END is the DMAP statement number. This number corresponds to the statement number in the DMAP listing for the solution sequence being used. The DMAP sequence listing will be printed as a part of the output if DIAG 14 or COMPILER LIST is inserted in the Executive Control Section.
- ⑮ BEGN indicates the start of execution of an MSC.Nastran module or subDMAP. The functions of these modules are given in [Section 7.4.1](#). Section 4 of the *MSC.Nastran Programmer's Manual* provides additional detailed information on MSC.Nastran functional modules.
- END indicates the end of execution of an MSC.Nastran module or subDMAP. The resources utilized by each module can be estimated by computing the incremental CPU SECs and I/O SECs required to execute the module. This line of information will not normally appear unless the module required more than the CPU seconds defined in SYSTEM(20). SYSTEM is defined via the NASTRAN statement and has a default of 5. M is the timing constant for the user's computer type, as described in the [MSC.Nastran 2004 Installation and Operations Guide](#). The user may force the output of this line by including the keyword SYSTEM(20) = 0 on the NASTRAN statement.
- ⑯ The line “*8**” is a typical matrix trailer. This line is output for each MSC.Nastran matrix that is generated if DIAG 8 is included in the Executive Control Section. Some parameters may be redefined using DMAP as explained in the [MSC.Nastran 2005 DMAP Programmer's Guide](#). Other terms in the DIAG 8 output are: the number of GINO blocks in the matrix (BLOCKS); the average number of terms per string (STRL); the number of strings in a matrix data block (NBRSTR); the number of bank terms in a matrix data block on the average (BNDL); the number of banks in a matrix data block (NBRBND); the average first nonzero row position (ROWL); the average of full bank width (BNDAVG); the maximum full bank width of all columns of data blocks (BNDMAX); the number of null columns in a data block (NULCOL).
- ⑰ User Information Message 4157 describes parameters for decomposition. The format of this message varies slightly if either sparse or parallel methods are selected.
- ⑱ The time estimation and method selection data are printed for DCMP, FBS, and MPYAD modules if DIAG 19 is included in the Executive Control Section.

- ①9 A detailed analysis of timing data will be printed in the .f04 file if DIAG 49 is set. The results of this analysis appear at the end of the .f04 file in three separate sections. The first section contains information about each module and submodule. The second section reports statistical information about the submodules (FBS, MPYAD, and DCMP) grouped by specific methods. Instances where the CPU estimate does not pass the following criteria are output:

$$\frac{|\text{Estimated CPU} - \text{Actual CPU}|}{\text{Actual CPU}} < 0.05$$

or

$$|\text{Estimated CPU} - \text{Actual CPU}| < 5\% \text{ of the total CPU of the job}$$

The third section contains information about the modules listed in alphabetical order. (The numbers shown are for a very small model with execution times less than the integer seconds used in these calculations.)

- ②0 This table provides the CPU usage for each process created and executed by a parallel module.
- ②1 The column headed BLOCKS USED contains the number of GINO and executive blocks used at the end of the job. One GINO Block = BUFSIZE-1 words.
- ②2 The column headed HIWATER BLOCK contains the maximum number of GINO and executive blocks used during the entire execution.

Listing 7-1 Example Execution Summary File (.f04)

MACHINE	MODEL	OPERATING SYSTEM	MSC/NASTRAN	BUILD DATE	RUN DATE		
DIGITAL EQUIPMENT	VAX 6000-410	VMS V5.4-3	VERSION 68	MAR 3, 1994	MAR 11, 1994		
1 2 3 === M S C / N A S T R A N E X E C U T I O N S U M M A R Y ===							
DAY TIME	ELAPSED	I/O SEC	DEL_I/O	CPU SEC	DEL_CPU	SUB_DMAP/DMAP_MODULE	MESSAGES
10:01:52	0:00	0.0	0.0	0.0	0.0	DBINIT	BGN
** CURRENT PROJECT ID = 'BLANK' ' ** CURRENT VERSION ID = 1							
5 SUMMARY OF FILE ASSIGNMENT FOR THE PRIMARY DATABASE (DBSNO 1, V68) 6 7 8							
ASSIGNED PHYSICAL FILE NAME (/ORIGINAL)	LOGICAL NAME	DBSET	STATUS	BUFSIZE	CLUSTER SIZE	TIME STAMP	
-----	-----	-----	-----	-----	-----	-----	
SCR_KNIGHT:[SCRATCH]N10013850.MASTER	MASTER	MASTER	NEW	1409	1	940311100153	
SCR_KNIGHT:[SCRATCH]N10013850.USROBJ	USROBJ	USROBJ	NEW	1409	1	940311100154	
SCR_KNIGHT:[SCRATCH]N10013850.USRSOU	USRSOU	USRSOU	NEW	1409	1	940311100155	
SCR_KNIGHT:[SCRATCH]N10013850.DBALL	DBALL	DBALL	NEW	1409	1	940311100156	
SCR_KNIGHT:[SCRATCH]N10013850.OBJSCR	OBJSCR	OBJSCR	NEW	1409	1	940311100157	
**** MEM FILE ****	* N/A *	SCRATCH					
SCR_KNIGHT:[SCRATCH]N10013850.SCRATCH	SCRATCH	SCRATCH	NEW	1409	1	940311100158	
SCR_KNIGHT:[SCRATCH]N10013850.SCR300	SCR300	SCRATCH	NEW	1409	1	940311100159	
10:01:57	0:05	4.5	4.5	0.6	0.7	DBINIT	END
10:01:57	0:05	4.7	0.2	0.7	0.0	XCSA	BGN
9 SUMMARY OF FILE ASSIGNMENT FOR THE DELIVERY DATABASE (DBSNO 2, V68)							
ASSIGNED PHYSICAL FILE NAME (/ORIGINAL)	LOGICAL NAME	DBSET	STATUS	BUFSIZE	CLUSTER SIZE	TIME STAMP	
-----	-----	-----	-----	-----	-----	-----	
nast:[nast68.DEL]SSS.MASTERA	MASTERA	MASTER	OLD	1409	1	940303153920	
/NAST:[NAST68.DEL.SOURCE.OUT]SSS.MASTERA							
nast:[nast68.DEL]SSS.MSCOBJ	MSCOBJ	MSCOBJ	OLD	1409	1	940303154017	
/NAST:[NAST68.DEL.SOURCE.OUT]SSS.MSCOBJ							
nast:[nast68.DEL]SSS.MSCSOU	MSCSOU	MSCSOU	OLD	1409	1	940303154032	
/NAST:[NAST68.DEL.SOURCE.OUT]SSS.MSCSOU							
10:02:57	1:05	115.3	110.7	7.1	6.4	XCSA	END
10:02:57	1:05	115.3	0.0	7.1	0.0	CGPI	BGN
10:02:57	1:05	115.3	0.0	7.1	0.1	CGPI	END
10:02:57	1:05	115.3	0.0	7.1	0.0	LINKER	BGN
10:03:09	1:17	133.6	18.3	9.1	2.0	LINKER	END
10 ** MASTER DIRECTORIES ARE LOADED IN MEMORY.							
USER OPENCORE (HICORE)	=	6074952	WORDS				
EXECUTIVE SYSTEM WORK AREA	=	76045	WORDS				
MASTER(RAM)	=	30000	WORDS				
SCRATCH(MEM) AREA	=	140900	WORDS (100 BUFFERS)				
BUFFER POOL AREA (GINO/EXEC)	=	52503	WORDS (37 BUFFERS)				
TOTAL MSC/NASTRAN MEMORY LIMIT	=	6374400	WORDS				
11 12							
DAY TIME	ELAPSED	I/O SEC	DEL_I/O	CPU SEC	DEL_CPU	SUB_DMAP/DMAP_MODULE	MESSAGES
10:03:11	1:19	135.6	2.0	9.4	0.3	SESTATIC	19 IFPL BEGN
10:03:11	1:19	135.8	0.1#	9.4	0.0#	SESTATIC	19 IFPL END
10:03:12	1:20	136.1	0.4	9.5	0.0	IFPL	22 IFPL BEGN
10:03:12	1:20	136.1	0.0#	9.5	0.0#	IFPL	22 IFPL END
10:03:12	1:20	136.1	0.0	9.5	0.0	IFPL	24 XEQUIV BEGN
10:03:12	1:20	136.1	0.0#	9.5	0.0#	IFPL	24 XEQUIV END
10:03:12	1:20	136.1	0.0	9.5	0.0	IFPL	25 XEQUIV BEGN
10:03:12	1:20	136.1	0.0#	9.5	0.0#	IFPL	25 XEQUIV END
10:03:12	1:20	136.1	0.0	9.5	0.0	IFPL	26 XEQUIV BEGN
10:03:12	1:20	136.1	0.0#	9.5	0.0#	IFPL	26 XEQUIV END
10:03:12	1:20	136.1	0.0	9.5	0.0	IFPL	27 XEQUIV BEGN
10:03:12	1:20	136.1	0.0#	9.5	0.0#	IFPL	27 XEQUIV END
10:03:12	1:20	136.1	0.0	9.6	0.0	IFPL	28 XEQUIV BEGN
10:03:12	1:20	136.1	0.0#	9.6	0.0#	IFPL	28 XEQUIV END
10:03:12	1:20	136.1	0.0	9.6	0.0	IFPL	35 XSORT BEGN
13 REAL CORE IS 6074952							
10:03:14	1:22	136.1	0.0#	10.8	1.3#	IFPL	35 XSORT END
10:03:36	1:44	137.1	1.0	14.8	4.0	IFPL	61 IFP BEGN
10:03:37	1:45	137.1	0.0	14.9	0.1	IFP	
10:03:39	1:47	137.1	0.0#	15.1	0.3#	IFPL	61 IFP END
10:03:39	1:47	137.1	0.0	15.1	0.0	IFPL	62 MODEPT BEGN
10:03:39	1:47	137.1	0.0#	15.1	0.0#	IFPL	62 MODEPT END

Listing 7-1 Example Execution Summary File (.f04) (continued)

10:03:39	1:47	137.1	0.0	15.2	0.0	IFPL	63	MODGM2	BEGN
10:03:40	1:48	137.1	0.0#	15.2	0.0#	IFPL	63	MODGM2	END
10:03:40	1:48	137.1	0.0	15.2	0.0	IFPL	64	PVT	BEGN
10:03:40	1:48	137.1	0.0#	15.3	0.0#	IFPL	64	PVT	END
10:03:40	1:48	137.6	0.5	15.3	0.1	IFPL	79	PROJVER	BEGN
10:03:41	1:49	137.6	0.0#	15.3	0.0#	IFPL	79	PROJVER	END
10:03:41	1:49	137.6	0.0	15.3	0.0	IFPL	81	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	81	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	82	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	82	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	83	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	83	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	84	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	84	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	85	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	85	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	86	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	86	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	87	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	87	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	88	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	88	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	89	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	89	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.4	0.0	IFPL	90	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.4	0.0#	IFPL	90	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.5	0.0	IFPL	92	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.5	0.0#	IFPL	92	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.5	0.0	IFPL	93	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.5	0.0#	IFPL	93	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.5	0.0	IFPL	94	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.5	0.0#	IFPL	94	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.5	0.0	IFPL	96	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.5	0.0#	IFPL	96	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.5	0.0	IFPL	97	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.5	0.0#	IFPL	97	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.5	0.0	IFPL	98	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.6	0.0#	IFPL	98	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.6	0.0	IFPL	99	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.6	0.0#	IFPL	99	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.6	0.0	IFPL	100	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.6	0.0#	IFPL	100	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.6	0.0	IFPL	101	XEQUIV	BEGN
10:03:41	1:49	137.6	0.0#	15.6	0.0#	IFPL	101	XEQUIV	END
10:03:41	1:49	137.6	0.0	15.6	0.0	IFPL	115	PVT	BEGN
10:03:41	1:49	137.6	0.0#	15.6	0.0#	IFPL	115	PVT	END
10:03:41	1:49	137.6	0.0	15.6	0.0	IFPL	128	DELETE	BEGN
10:03:41	1:49	137.6	0.0#	15.6	0.0#	IFPL	128	DELETE	END
10:03:41	1:49	137.8	0.2	15.6	0.0	IFPL	154	PLTSET	BEGN
10:03:41	1:49	137.8	0.0#	15.7	0.0#	IFPL	154	PLTSET	END
10:03:41	1:49	137.8	0.0	15.7	0.0	IFPL	157	GP0	BEGN
10:03:42	1:50	137.8	0.0#	15.7	0.0#	IFPL	157	GP0	END
10:03:42	1:50	137.8	0.0	15.7	0.0	IFPL	176	SEQP	BEGN
10:03:43	1:51	137.8	0.0#	15.9	0.1#	IFPL	176	SEQP	END
10:03:43	1:51	137.8	0.0	15.9	0.0	IFPL	182	GP1	BEGN
10:03:44	1:52	137.8	0.0#	16.1	0.2#	IFPL	182	GP1	END
10:03:44	1:52	137.9	0.1	16.1	0.0	SESTATIC	36	PHASE0	BEGN
10:03:44	1:52	138.1	0.2#	16.1	0.0#	SESTATIC	36	PHASE0	END
10:03:44	1:52	138.1	0.0	16.1	0.0	PHASE0	24	DTIIN	BEGN
10:03:44	1:52	138.1	0.0#	16.2	0.0#	PHASE0	24	DTIIN	END
10:03:44	1:52	138.1	0.0	16.2	0.0	PHASE0	25	GP1	BEGN
10:03:45	1:53	138.1	0.0#	16.4	0.2#	PHASE0	25	GP1	END
10:03:46	1:54	138.1	0.0	16.4	0.0	PHASE0	28	SEP1	BEGN
10:03:46	1:54	138.1	0.0#	16.4	0.0#	PHASE0	28	SEP1	END
10:03:46	1:54	138.1	0.0	16.4	0.0	PHASE0	37	XEQUIV	BEGN
10:03:46	1:54	138.1	0.0#	16.4	0.0#	PHASE0	37	XEQUIV	END
10:03:46	1:54	138.1	0.0	16.5	0.0	PHASE0	40	PVT	BEGN
10:03:48	1:56	138.1	0.0#	16.6	0.2#	PHASE0	40	PVT	END
10:03:48	1:56	138.1	0.0	16.6	0.0	PHASE0	46	TABPRT	BEGN
10:03:48	1:56	138.1	0.0#	16.7	0.0#	PHASE0	46	TABPRT	END
10:03:48	1:56	138.1	0.0	16.7	0.0	PHASE0	47	SEP3	BEGN
10:03:48	1:56	138.1	0.0#	16.7	0.0#	PHASE0	47	SEP3	END
10:03:48	1:56	138.2	0.1	16.7	0.1	PHASE0	88	PVT	BEGN
10:03:49	1:57	138.2	0.0#	16.8	0.1#	PHASE0	88	PVT	END
10:03:49	1:57	138.9	0.7	16.8	0.0	PHASE0	96	XEQUIV	BEGN
10:03:49	1:57	138.9	0.0#	16.8	0.0#	PHASE0	96	XEQUIV	END
10:03:49	1:57	138.9	0.0	16.9	0.0	PHASE0	97	XEQUIV	BEGN
10:03:49	1:57	138.9	0.0#	16.9	0.0#	PHASE0	97	XEQUIV	END
10:03:49	1:57	138.9	0.0	16.9	0.0	PHASE0	98	XEQUIV	BEGN
10:03:49	1:57	138.9	0.0#	16.9	0.0#	PHASE0	98	XEQUIV	END
10:03:49	1:57	138.9	0.0	16.9	0.0	PHASE0	99	XEQUIV	BEGN
10:03:49	1:57	138.9	0.0#	16.9	0.0#	PHASE0	99	XEQUIV	END
10:03:49	1:57	138.9	0.0	16.9	0.0	PHASE0	100	XEQUIV	BEGN
10:03:49	1:57	138.9	0.0#	16.9	0.0#	PHASE0	100	XEQUIV	END

Listing 7-1 Example Execution Summary File (.f04) (continued)

```

10:03:49 1:57 138.9 0.0 16.9 0.0 PHASE0 101 XEQUIV BEGN
10:03:49 1:57 138.9 0.0# 16.9 0.0# PHASE0 101 XEQUIV END
10:03:49 1:57 138.9 0.0 16.9 0.0 PHASE0 102 XEQUIV BEGN
10:03:49 1:57 138.9 0.0# 16.9 0.0# PHASE0 102 XEQUIV END
10:03:49 1:57 138.9 0.0 16.9 0.0 PHASE0 103 XEQUIV BEGN
10:03:49 1:57 138.9 0.0# 16.9 0.0# PHASE0 103 XEQUIV END
10:03:49 1:57 138.9 0.0 16.9 0.0 PHASE0 104 XEQUIV BEGN
10:03:49 1:57 138.9 0.0# 16.9 0.0# PHASE0 104 XEQUIV END
10:03:49 1:57 138.9 0.0 16.9 0.0 PHASE0 105 XEQUIV BEGN
10:03:49 1:57 138.9 0.0# 17.0 0.0# PHASE0 105 XEQUIV END
10:03:50 1:58 140.7 1.8 17.1 0.2 PHASE0 224 PROJVER BEGN
10:03:50 1:58 140.7 0.0# 17.1 0.0# PHASE0 224 PROJVER END
10:03:50 1:58 140.7 0.0 17.1 0.0 PHASE0 241 SETQ BEGN
10:03:50 1:58 140.8 0.1# 17.1 0.0# PHASE0 241 SETQ END
10:03:50 1:58 140.8 0.0 17.1 0.0 SETQ 10 XEQUIV BEGN
10:03:50 1:58 140.8 0.0# 17.1 0.0# SETQ 10 XEQUIV END
10:03:50 1:58 140.8 0.1 17.3 0.2 PHASE0 307 GP1 BEGN
10:03:50 1:58 140.8 0.1# 17.3 0.2# PHASE0 307 GP1 END
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
GP1 307 VGF5 1 12 2 2 0 0.0000 1 0 1 0 0 0 0 0 1 *8**
10:03:51 1:59 140.8 0.0# 17.5 0.2# PHASE0 307 GP1 END
10:03:51 1:59 140.9 0.1 17.5 0.0 PHASE0 311 XEQUIV BEGN
10:03:51 1:59 140.9 0.0# 17.5 0.0# PHASE0 311 XEQUIV END
10:03:51 1:59 140.9 0.0 17.5 0.0 PHASE0 313 SEPREP2 BEGN
10:03:51 1:59 141.0 0.1# 17.5 0.0# PHASE0 313 SEPREP2 END
10:03:51 1:59 141.0 0.0 17.5 0.0 SEPREP2 6 GP2 BEGN
10:03:51 1:59 141.0 0.0# 17.5 0.0# SEPREP2 6 GP2 END
10:03:51 1:59 141.1 0.1 17.6 0.0 SEPREP2 12 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.6 0.0# SEPREP2 12 XEQUIV END
10:03:51 1:59 141.1 0.0 17.6 0.0 SEPREP2 13 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.6 0.0# SEPREP2 13 XEQUIV END
10:03:51 1:59 141.1 0.0 17.6 0.0 SEPREP2 14 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.6 0.0# SEPREP2 14 XEQUIV END
10:03:51 1:59 141.1 0.0 17.6 0.0 SEPREP2 15 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.6 0.0# SEPREP2 15 XEQUIV END
10:03:51 1:59 141.1 0.0 17.6 0.0 PHASE0 332 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.6 0.0# PHASE0 332 XEQUIV END
10:03:51 1:59 141.1 0.0 17.6 0.0 PHASE0 333 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.6 0.0# PHASE0 333 XEQUIV END
10:03:51 1:59 141.1 0.0 17.7 0.0 PHASE0 335 GP3 BEGN
10:03:51 1:59 141.1 0.0# 17.7 0.0# PHASE0 335 GP3 END
10:03:51 1:59 141.1 0.0 17.7 0.0 PHASE0 336 LCGEN BEGN
10:03:51 1:59 141.1 0.0# 17.7 0.0# PHASE0 336 LCGEN END
10:03:51 1:59 141.1 0.0 17.7 0.0 PHASE0 341 XEQUIV BEGN
10:03:51 1:59 141.1 0.0# 17.7 0.0# PHASE0 341 XEQUIV END
10:03:51 1:59 141.1 0.0 17.8 0.0 PHASE0 352 BCDR BEGN
10:03:51 1:59 141.1 0.0# 17.8 0.0# PHASE0 352 BCDR END
10:03:51 1:59 141.2 0.1 17.8 0.0 PHASE0 367 GP4 BEGN
10:03:51 1:59 141.2 0.1# 17.8 0.0# PHASE0 367 GP4 END
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
GP4 367 Y5B 1 6 2 2 0 0.0000 1 0 1 0 0 0 0 0 1 *8**
10:03:52 2:00 141.2 0.0# 17.8 0.0# PHASE0 367 GP4 END
10:03:52 2:00 141.2 0.0 17.9 0.0 PHASE0 373 XPURGE BEGN
10:03:52 2:00 141.2 0.0# 17.9 0.0# PHASE0 373 XPURGE END
10:03:52 2:00 141.2 0.0 17.9 0.0 PHASE0 378 UMERG1 BEGN
10:03:52 2:00 141.2 0.0# 17.9 0.0# PHASE0 378 UMERG1 END
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
UMERG1 378 SCRATCH 1 12 2 1 6 0.5000 1 6 1 0 0 0 6 6 0 *8**
10:03:52 2:00 141.2 0.0# 17.9 0.0# PHASE0 378 UMERG1 END
10:03:52 2:00 141.3 0.1 18.0 0.1 SESTATIC 39 SETQ BEGN
10:03:52 2:00 141.3 0.0# 18.0 0.0# SESTATIC 39 SETQ END
10:03:52 2:00 141.3 0.0 18.0 0.0 SETQ 10 XEQUIV BEGN
10:03:52 2:00 141.3 0.0# 18.0 0.0# SETQ 10 XEQUIV END
10:03:52 2:00 141.3 0.0 18.2 0.2 SESTATIC 40 DBSTATUS BEGN
10:03:52 2:00 141.3 0.0# 18.2 0.0# SESTATIC 40 DBSTATUS END
10:03:52 2:00 141.3 0.0 18.2 0.0 SESTATIC 45 MATGEN BEGN
10:03:52 2:00 141.3 0.0# 18.2 0.0# SESTATIC 45 MATGEN END
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
MATGEN 45 TEMPALL 2 2 6 2 2 0.5000 1 1 2 0 0 0 1 1 0 *8**
10:03:52 2:00 141.3 0.0# 18.2 0.0# SESTATIC 45 MATGEN END
10:03:52 2:00 141.3 0.0 18.2 0.0 SESTATIC 46 RESTART BEGN
10:03:52 2:00 141.9 0.6# 18.3 0.0# SESTATIC 46 RESTART END
10:03:52 2:01 142.0 0.1 18.3 0.0 SESTATIC 47 DELETE BEGN
10:03:52 2:01 142.0 0.0# 18.3 0.0# SESTATIC 47 DELETE END
10:03:52 2:01 142.1 0.1 18.3 0.0 SESTATIC 49 PHASELDR BEGN
10:03:52 2:01 142.2 0.1# 18.3 0.0# SESTATIC 49 PHASELDR END
10:03:52 2:01 142.4 0.2 18.4 0.1 PHASELDR 70 PVT BEGN
10:03:52 2:01 142.4 0.0# 18.4 0.1# PHASELDR 70 PVT END
10:03:52 2:01 142.4 0.0 18.4 0.0 PHASELDR 71 SETQ BEGN
10:03:52 2:01 142.4 0.0# 18.5 0.0# PHASELDR 71 SETQ END
10:03:52 2:01 142.4 0.0 18.5 0.0 SETQ 10 XEQUIV BEGN
10:03:52 2:01 142.4 0.0# 18.5 0.0# SETQ 10 XEQUIV END
10:03:52 2:01 142.4 0.0 18.6 0.2 PHASELDR 74 PHASE1A BEGN
10:03:52 2:01 142.6 0.2# 18.7 0.0# PHASELDR 74 PHASE1A END
10:03:52 2:01 142.6 0.0 18.7 0.0 PHASE1A 24 TAL BEGN
10:03:52 2:01 142.6 0.0# 18.7 0.0# PHASE1A 24 TAL END
REAL CORE IS 6073541

```

Listing 7-1 Example Execution Summary File (.f04) (continued)

*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BN DL NBRBND ROW1 BNDVAVG BNDMAX NULCOL																	
EMA 79 KJJZ 12 12 6 2 8 0.2777 1 0 40 0 0 0 9 11 0 *8**																	
10:03:54	2:02	142.9	0.0#	19.0	0.0#	SEMG	79	EMA	END								
10:03:54	2:02	142.9	0.0	19.0	0.0	SEMG	95	XPURGE	BEGN								
10:03:54	2:02	142.9	0.0#	19.0	0.0#	SEMG	95	XPURGE	END								
10:03:55	2:03	143.1	0.2	19.0	0.0	SEMG	98	MTRXIN	BEGN								
REAL CORE IS 6074952																	
10:03:55	2:03	143.1	0.0#	19.0	0.0#	SEMG	98	MTRXIN	END								
10:03:55	2:03	143.1	0.0	19.0	0.0	SEMG	103	SEMGL	BEGN								
10:03:55	2:03	143.1	0.1#	19.1	0.0#	SEMG	103	SEMGL	END								
10:03:55	2:03	143.1	0.0	19.1	0.0	SEMGL	8	XEQUIV	BEGN								
10:03:55	2:03	143.1	0.0#	19.1	0.0#	SEMGL	8	XEQUIV	END								
10:03:55	2:03	143.1	0.0	19.1	0.0	SEMGL	15	XEQUIV	BEGN								
10:03:55	2:03	143.1	0.0#	19.1	0.0#	SEMGL	15	XEQUIV	END								
10:03:55	2:03	143.1	0.0	19.1	0.0	SEMGL	17	XEQUIV	BEGN								
10:03:55	2:03	143.1	0.0#	19.1	0.0#	SEMGL	17	XEQUIV	END								
10:03:55	2:03	143.1	0.0	19.1	0.0	SEMGL	19	XEQUIV	BEGN								
10:03:55	2:03	143.1	0.0#	19.1	0.0#	SEMGL	19	XEQUIV	END								
10:03:55	2:03	143.1	0.0	19.1	0.0	PHASE1A	31	SESUM	BEGN								
10:03:55	2:03	143.1	0.0#	19.1	0.0#	PHASE1A	31	SESUM	END								
10:03:55	2:03	143.1	0.0	19.2	0.0	SESUM	22	DELETE	BEGN								
10:03:55	2:03	143.1	0.0#	19.2	0.0#	SESUM	22	DELETE	END								
10:03:55	2:03	143.4	0.3	19.2	0.0	SESUM	25	XPURGE	BEGN								
10:03:55	2:03	143.4	0.0#	19.2	0.0#	SESUM	25	XPURGE	END								
10:03:55	2:03	143.6	0.2	19.2	0.0	PHASE1A	35	XEQUIV	BEGN								
10:03:55	2:03	143.6	0.0#	19.2	0.0#	PHASE1A	35	XEQUIV	END								
10:03:55	2:03	143.6	0.0	19.2	0.0	PHASE1A	60	SELG	BEGN								
10:03:55	2:03	143.7	0.1#	19.3	0.0#	PHASE1A	60	SELG	END								
10:03:55	2:03	143.7	0.0	19.3	0.0	SELG	11	SSGI	BEGN								
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BN DL NBRBND ROW1 BNDVAVG BNDMAX NULCOL																	
16 SSG1 11 SCRATCH 1 12 2 2 6 0.2500 1 3 1 0 0 0 3 3 0 *8**																	
SSG1 11 PJX 1 12 2 2 6 0.2500 1 3 1 0 0 0 3 3 0 *8**																	
10:03:56	2:04	143.7	0.0#	19.3	0.1#	SELG	11	SSGI	END								
10:03:56	2:04	143.7	0.0	19.4	0.0	SELG	22	XEQUIV	BEGN								
10:03:56	2:04	143.7	0.0#	19.4	0.0#	SELG	22	XEQUIV	END								
10:03:56	2:04	143.7	0.0	19.4	0.0	SELG	60	XEQUIV	BEGN								
10:03:56	2:04	143.7	0.0#	19.4	0.0#	SELG	60	XEQUIV	END								
10:03:56	2:04	143.7	0.0	19.4	0.0	SELG	62	VECPLOT	BEGN								
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BN DL NBRBND ROW1 BNDVAVG BNDMAX NULCOL																	
VECPLOT 62 PPJ 1 12 2 1 3 0.2500 1 3 1 0 0 0 3 3 0 *8**																	
10:03:56	2:04	143.7	0.0#	19.4	0.0#	SELG	62	VECPLOT	END								
10:03:56	2:04	143.7	0.0	19.5	0.0	PHASE1A	61	SESUM	BEGN								
10:03:56	2:04	143.7	0.0#	19.5	0.0#	PHASE1A	61	SESUM	END								
10:03:56	2:04	143.7	0.0	19.5	0.0	SESUM	22	DELETE	BEGN								
10:03:56	2:04	143.8	0.1#	19.5	0.0#	SESUM	22	DELETE	END								
10:03:56	2:04	144.4	0.6	19.6	0.0	SESUM	25	XPURGE	BEGN								
10:03:56	2:04	144.4	0.0#	19.6	0.0#	SESUM	25	XPURGE	END								
10:03:57	2:05	144.5	0.1	19.6	0.0	PHASE1A	65	SEL1	BEGN								
10:03:57	2:05	144.6	0.1#	19.6	0.0#	PHASE1A	65	SEL1	END								
10:03:57	2:05	144.6	0.0	19.6	0.0	SEL1	9	XEQUIV	BEGN								
10:03:57	2:05	144.6	0.0#	19.6	0.0#	SEL1	9	XEQUIV	END								
10:03:57	2:05	144.6	0.0	19.6	0.0	SEL1	11	XEQUIV	BEGN								
10:03:57	2:05	144.6	0.0#	19.6	0.0#	SEL1	11	XEQUIV	END								
10:03:57	2:05	144.6	0.0	19.7	0.0	PHASE1DR	78	DBSTATUSBEGN									
10:03:57	2:05	144.6	0.0#	19.7	0.0#	PHASE1DR	78	DBSTATUSEND									
10:03:57	2:05	144.6	0.0	19.7	0.0	PHASE1DR	80	XEQUIV	BEGN								
10:03:57	2:05	144.6	0.0#	19.7	0.0#	PHASE1DR	80	XEQUIV	END								
10:03:57	2:05	144.6	0.0	19.7	0.0	PHASE1DR	98	BCDR	BEGN								
10:03:57	2:05	144.6	0.0#	19.7	0.0#	PHASE1DR	98	BCDR	END								
10:03:57	2:05	144.6	0.0	19.7	0.0	PHASE1DR	100	PHASE1B	BEGN								
10:03:57	2:05	144.7	0.1#	19.8	0.0#	PHASE1DR	100	PHASE1B	END								
10:03:57	2:05	144.7	0.0	19.8	0.0	PHASE1B	31	DBSTATUSBEGN									
10:03:57	2:05	144.7	0.0#	19.8	0.0#	PHASE1B	31	DBSTATUSEND									
10:03:57	2:05	144.7	0.0	19.8	0.0	PHASE1B	32	SEKR	BEGN								
10:03:57	2:05	144.9	0.2#	19.8	0.0#	PHASE1B	32	SEKR	END								
10:03:57	2:05	144.9	0.0	19.9	0.0	SEKR	13	XEQUIV	BEGN								
10:03:57	2:05	144.9	0.0#	19.9	0.0#	SEKR	13	XEQUIV	END								
10:03:57	2:05	144.9	0.0	19.9	0.0	SEKR	14	XPURGE	BEGN								
10:03:57	2:05	144.9	0.0#	19.9	0.0#	SEKR	14	XPURGE	END								
10:03:57	2:05	144.9	0.0	19.9	0.0	SEKR	29	GPSP	BEGN								
10:03:57	2:05	144.9	0.0#	20.0	0.0#	SEKR	29	GPSP	END								
10:03:57	2:05	144.9	0.0	20.0	0.0	SEKR	30	PMLUSET	BEGN								
10:03:57	2:05	144.9	0.0#	20.0	0.0#	SEKR	30	PMLUSET	END								
10:03:57	2:05	144.9	0.0	20.0	0.0	SEKR	45	XEQUIV	BEGN								
10:03:57	2:05	144.9	0.0#	20.0	0.0#	SEKR	45	XEQUIV	END								
10:03:57	2:05	144.9	0.0	20.0	0.0	SEKR	46	XPURGE	BEGN								
10:03:57	2:05	144.9	0.0#	20.0	0.0#	SEKR	46	XPURGE	END								
10:03:57	2:05	144.9	0.0	20.0	0.0	SEKR	47	UPARTN	BEGN								

Listing 7-1 Example Execution Summary File (.f04) (continued)

*8** MODULE		DMAP	MATRIX	COLS	ROWS	F	T	NZWDS	DENS	BLOCKS	STRL	NBRSTR	BNDL	NBRBND	ROW1	BNDAVG	BNDMAX	NULCOL	
UPARTN	47	SCRATCH		1	12	2	1	6	0.5000	1	6	1	0	0	0	6	6	0	*8**
UPARTN	47	KFF		6	6	6	2	4	0.2777	1	0	10	0	0	0	3	5	0	*8**
UPARTN	47	KSF		6	6	6	2	4	0.2777	1	0	10	0	0	0	3	5	0	*8**
UPARTN	47	KFS		6	6	6	2	4	0.2777	1	0	10	0	0	0	3	5	0	*8**
UPARTN	47	KSS		6	6	6	2	4	0.2777	1	0	10	0	0	0	3	5	0	*8**
10:03:58	2:06	144.9		0.0#	20.1			0.0#	SEKR	47	UPARTN	END							
10:03:58	2:06	144.9		0.0	20.1			0.0	SEKR	48	XEQUIV	BEGN							
10:03:58	2:06	144.9		0.0#	20.1			0.0#	SEKR	48	XEQUIV	END							
10:03:58	2:06	144.9		0.0	20.1			0.0	SEKR	49	XPURGE	BEGN							
10:03:58	2:06	144.9		0.0#	20.1			0.0#	SEKR	49	XPURGE	END							
10:03:58	2:06	144.9		0.0	20.1			0.0	SEKR	54	XPURGE	BEGN							
10:03:58	2:06	144.9		0.0#	20.1			0.0#	SEKR	54	XPURGE	END							
10:03:58	2:06	144.9		0.0	20.1			0.0	SEKR	110	SESUM	BEGN							
10:03:58	2:06	144.9		0.0#	20.1			0.0#	SEKR	110	SESUM	END							
10:03:58	2:06	144.9		0.0	20.2			0.0	SESUM	22	DELETE	BEGN							
10:03:58	2:06	144.9		0.0#	20.2			0.0#	SESUM	22	DELETE	END							
10:03:58	2:06	145.3		0.4	20.2			0.0	SESUM	25	XPURGE	BEGN							
10:03:58	2:06	145.3		0.0#	20.2			0.0#	SESUM	25	XPURGE	END							
10:03:58	2:06	145.4		0.1	20.3			0.0	PHASE1B	33	SESUM	BEGN							
10:03:58	2:06	145.4		0.0#	20.3			0.0#	PHASE1B	33	SESUM	END							
10:03:59	2:07	145.4		0.0	20.3			0.1	SESUM	22	DELETE	BEGN							
10:03:59	2:07	145.4		0.0#	20.3			0.0#	SESUM	22	DELETE	END							
10:03:59	2:07	145.9		0.4	20.4			0.0	SESUM	25	XPURGE	BEGN							
10:03:59	2:07	145.9		0.0#	20.4			0.0#	SESUM	25	XPURGE	END							
10:03:59	2:07	146.0		0.1	20.4			0.0	PHASE1B	36	XEQUIV	BEGN							
10:03:59	2:07	146.0		0.0#	20.4			0.0#	PHASE1B	36	XEQUIV	END							
10:03:59	2:07	146.1		0.1	20.4			0.0	PHASE1B	41	PMLUSET	BEGN							
10:03:59	2:07	146.1		0.0#	20.4			0.0#	PHASE1B	41	PMLUSET	END							
10:03:59	2:07	146.5		0.4	20.5			0.1	PHASE1B	76	SEGOA	BEGN							
10:03:59	2:07	146.6		0.1#	20.5			0.0#	PHASE1B	76	SEGOA	END							
10:03:59	2:07	146.6		0.0	20.5			0.0	PHASE1B	77	DBSTATUSBEGN								
10:03:59	2:07	146.6		0.0#	20.5			0.0#	PHASE1B	77	DBSTATUSEND								
10:03:59	2:07	146.6		0.0	20.5			0.0	PHASE1B	78	SELR	BEGN							
10:03:59	2:07	146.6		0.1#	20.6			0.0#	PHASE1B	78	SELR	END							
10:03:59	2:07	146.6		0.0	20.6			0.0	SELR	16	XEQUIV	BEGN							
10:03:59	2:07	146.6		0.0#	20.6			0.0#	SELR	16	XEQUIV	END							
10:03:59	2:07	146.6		0.0	20.6			0.0	SELR	37	XEQUIV	BEGN							
10:03:59	2:07	146.6		0.0#	20.6			0.0#	SELR	37	XEQUIV	END							
10:03:59	2:07	146.6		0.0	20.6			0.0	SELR	38	XPURGE	BEGN							
10:03:59	2:07	146.6		0.0#	20.6			0.0#	SELR	38	XPURGE	END							
10:03:59	2:07	146.6		0.0	20.6			0.0	SELR	40	XPURGE	BEGN							
10:03:59	2:07	146.6		0.0#	20.6			0.0#	SELR	40	XPURGE	END							
10:03:59	2:07	146.6		0.0	20.6			0.0	SELR	41	UPARTN	BEGN							
*8** MODULE	DMAP	MATRIX	COLS	ROWS	F	T	NZWDS	DENS	BLOCKS	STRL	NBRSTR	BNDL	NBRBND	ROW1	BNDAVG	BNDMAX	NULCOL		
UPARTN	41	SCRATCH		1	12	2	1	6	0.5000	1	6	1	0	0	0	6	6	0	*8**
10:03:59	2:07	146.6		0.0#	20.6			0.0#	SELR	41	UPARTN	END							
10:03:59	2:07	146.6		0.0	20.6			0.0	SELR	52	XEQUIV	BEGN							
10:03:59	2:07	146.6		0.0#	20.6			0.0#	SELR	52	XEQUIV	END							
10:03:59	2:07	146.6		0.0	20.7			0.0	SELR	64	SSG2	BEGN							
*8** MODULE	DMAP	MATRIX	COLS	ROWS	F	T	NZWDS	DENS	BLOCKS	STRL	NBRSTR	BNDL	NBRBND	ROW1	BNDAVG	BNDMAX	NULCOL		
SSG2	64	SCRATCH		1	12	2	1	6	0.5000	1	6	1	0	0	0	6	6	0	*8**
SSG2	64	SCRATCH		1	6	2	2	6	0.5000	1	3	1	0	0	0	3	3	0	*8**
SSG2	64	PSS		1	6	2	2	0	0.0000	1	0	1	0	0	0	0	0	1	*8**
									REAL CORE IS										
SSG2	64	PA		1	6	2	2	6	0.5000	1	3	1	0	0	0	3	3	0	*8**
10:04:00	2:08	146.6		0.0#	20.7			0.1#	SELR	64	SSG2	END							
10:04:00	2:08	146.6		0.0	20.7			0.0	SELR	65	XPURGE	BEGN							
10:04:00	2:08	146.6		0.0#	20.7			0.0#	SELR	65	XPURGE	END							
10:04:00	2:08	146.6		0.0	20.7			0.0	SELR	75	XPURGE	BEGN							
10:04:00	2:08	146.6		0.0#	20.8			0.0#	SELR	75	XPURGE	END							
10:04:00	2:08	146.6		0.0	20.8			0.0	PHASE1B	80	XEQUIV	BEGN							
10:04:00	2:08	146.6		0.0#	20.8			0.0#	PHASE1B	80	XEQUIV	END							
10:04:00	2:08	146.6		0.0	20.8			0.0	PHASE1B	81	SESUM	BEGN							
10:04:00	2:08	146.6		0.0#	20.8			0.0#	PHASE1B	81	SESUM	END							
10:04:00	2:08	146.6		0.0	20.8			0.0	SESUM	22	DELETE	BEGN							
10:04:00	2:08	146.6		0.0#	20.8			0.0#	SESUM	22	DELETE	END							
10:04:00	2:08	147.1		0.4	20.8			0.0	SESUM	25	XPURGE	BEGN							
10:04:00	2:08	147.1		0.0#	20.9			0.0#	SESUM	25	XPURGE	END							
10:04:00	2:08	147.5		0.4	21.0			0.1	SESTATIC	63	BCDR	BEGN							
10:04:00	2:08	147.5		0.0#	21.0			0.0#	SESTATIC	63	BCDR	END							
10:04:00	2:08	147.5		0.0	21.0			0.0	SESTATIC	64	DBSTATUSBEGN								
10:04:01	2:09	147.5		0.0#	21.0			0.0#	SESTATIC	64	DBSTATUSEND								
10:04:01	2:09	147.5		0.0	21.0			0.0	SESTATIC	71	PHASE1C	BEGN							
10:04:01	2:09	147.6		0.1#	21.0			0.0#	SESTATIC	71	PHASE1C	END							
10:04:01	2:09	147.6		0.0	21.0			0.0	PHASE1C	9	DBSTATUSBEGN								
10:04:01	2:09	147.6		0.0#	21.1			0.0#	PHASE1C	9	DBSTATUSEND								
10:04:01	2:09	147.6		0.0	21.1			0.0	PHASE1C	10	SEKRRS	BEGN							
10:04:01	2:09	147.6		0.1#	21.1			0.0#	PHASE1C	10</									

Listing 7-1 Example Execution Summary File (.f04) (continued)

*** USER INFORMATION MESSAGE 4157 (DFMSA) ---PARAMETERS FOR SPARSE DECOMPOSITION OF DATA BLOCK KLL (TYPE=RDP) FOLLOW

17

```

MATRIX SIZE = 6 ROWS NUMBER OF NONZEROES = 8 TERMS
NUMBER OF ZERO COLUMNS = 0 NUMBER OF ZERO DIAGONAL TERMS = 0
CPU TIME ESTIMATE = 0 SEC I/O TIME ESTIMATE = 0 SEC
ESTIMATED MEMORY REQUIREMENT = 15 K WORDS MEMORY AVAILABLE = 6074 K WORDS
EST. INTEGER WORDS IN FACTOR = 1 K WORDS EST. NONZERO TERMS = 1 K TERMS
ESTIMATED MAXIMUM FRONT SIZE = 2 TERMS

*** USER INFORMATION MESSAGE 6439 (DFMSA) ---ACTUAL MEMORY AND DISK SPACE REQUIREMENTS FOR SPARSE SYM. DECOMPOSITION
SPARSE DECOMP MEMORY REQUIRED = 15 K WORDS MAXIMUM FRONT SIZE = 2 TERMS
INTEGER WORDS IN FACTOR = 1 K WORDS NONZERO TERMS IN FACTOR = 1 K TERMS

*8** MODULE DMAP MATRIX COLS ROWS F T IBLKS NBLKS NUMFRT FRMTRX
DCMP 18 LLL 6 6 13 2 1 1 4 2 *8**
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
DCMP 18 SCRATCH 1 6 2 2 6 0.5000 1 3 1 0 0 0 6 6 0 *8**
DCMP 18 SCRATCH 1 6 2 2 6 0.5000 1 3 1 0 0 0 6 6 0 *8**
10:04:02 2:10 147.6 0.0# 21.2 0.1# SEKRRS 18 DCMP END
10:04:02 2:10 147.6 0.0 21.3 0.0 SEKRRS 26 XPURGE BEGN
10:04:02 2:10 147.6 0.0# 21.3 0.0# SEKRRS 26 XPURGE END
10:04:02 2:10 147.6 0.0 21.3 0.0 PHASE1C 11 SESUM BEGN
10:04:02 2:10 147.6 0.0# 21.3 0.0# PHASE1C 11 SESUM END
10:04:02 2:10 147.6 0.0 21.3 0.0 SESUM 22 DELETE BEGN
10:04:02 2:10 147.6 0.0# 21.3 0.0# SESUM 22 DELETE END
10:04:02 2:10 147.9 0.3 21.3 0.0 SESUM 25 XPURGE BEGN
10:04:02 2:10 147.9 0.0# 21.3 0.0# SESUM 25 XPURGE END
10:04:02 2:10 148.1 0.2 21.4 0.0 PHASE1C 12 DBSTATUSBEGN
10:04:02 2:10 148.1 0.0# 21.4 0.0# PHASE1C 12 DBSTATUSEND
10:04:02 2:10 148.1 0.0 21.4 0.0 PHASE1C 18 DBSTATUSBEGN
10:04:02 2:10 148.1 0.0# 21.4 0.0# PHASE1C 18 DBSTATUSEND
10:04:02 2:10 148.1 0.0 21.4 0.0 PHASE1C 21 SELRRS BEGN
10:04:02 2:10 148.1 0.1# 21.4 0.0# PHASE1C 21 SELRRS END
10:04:02 2:10 148.1 0.0 21.4 0.0 SELRRS 5 XEQUIV BEGN
10:04:02 2:10 148.1 0.0# 21.4 0.0# SELRRS 5 XEQUIV END
10:04:02 2:10 148.1 0.0 21.4 0.0 SELRRS 6 XPURGE BEGN
10:04:02 2:10 148.1 0.0# 21.4 0.0# SELRRS 6 XPURGE END
10:04:02 2:10 148.1 0.0 21.5 0.0 PHASE1C 22 SESUM BEGN
10:04:02 2:10 148.1 0.0# 21.5 0.0# PHASE1C 22 SESUM END
10:04:02 2:10 148.1 0.0 21.5 0.1 SESUM 22 DELETE BEGN
10:04:02 2:10 148.1 0.0# 21.5 0.0# SESUM 22 DELETE END
10:04:03 2:11 148.4 0.3 21.5 0.0 SESUM 25 XPURGE BEGN
10:04:03 2:11 148.4 0.0# 21.5 0.0# SESUM 25 XPURGE END
10:04:03 2:11 148.6 0.2 21.6 0.0 SESTATIC 72 DBSTATUSBEGN
10:04:03 2:11 148.6 0.0# 21.6 0.0# SESTATIC 72 DBSTATUSEND
10:04:03 2:11 148.6 0.0 21.6 0.0 SESTATIC 73 STATRS BEGN
10:04:03 2:11 148.6 0.1# 21.6 0.0# SESTATIC 73 STATRS END
10:04:03 2:11 148.6 0.0 21.6 0.0 STATRS 6 XPURGE BEGN
10:04:03 2:11 148.6 0.0# 21.6 0.0# STATRS 6 XPURGE END
10:04:03 2:11 148.6 0.0 21.6 0.0 STATRS 7 XEQUIV BEGN
10:04:03 2:11 148.6 0.0# 21.6 0.0# STATRS 7 XEQUIV END
10:04:03 2:11 148.6 0.0 21.6 0.0 STATRS 27 SSG3 BEGN
REAL CORE IS 6074952
REAL CORE IS 6074952
REAL CORE IS 6074952
REAL CORE IS 6074952

```

0*** DIAG 19 --FBS-- FACTOR TRAILER=(6 6 13 2 2 2222), METHOD S2-PASSES= 1, CPU= 0.0, I/O= 0

18 R.H.S. TRAILER=(1 6 2 2 6 5000)

```

*** USER INFORMATION MESSAGE 4153. FBS METHOD S2 TIME ESTIMATE TO FORM UL - CPU= 0, I/O= 0, TOTAL= 0, PASSES=
1
10:04:03 2:11 148.6 0.0 21.7 0.0 FBS BEGN
10:04:03 2:11 148.6 0.0# 21.7 0.0# FBS END
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
SSG3 27 UL 1 6 2 2 10 0.8333 1 2 2 0 0 0 6 6 0 *8**
REAL CORE IS 6074952
* M MATRIX KLL TRAILER=( 6 6 6 2 4 2777), METHOD 1 -PASSES= 1, CPU= 0.0, I/O= 0.3, TOTAL= 0.3
* P MATRIX UL TRAILER=( 1 6 2 2 10 8333), METHOD 2 -PASSES= 1, CPU=999999.0, I/O= 0.2,
TOTAL=999999.0
* Y MATRIX PLI TRAILER=( 1 6 2 2 6 5000), METHOD 3 -PASSES= 1, CPU=999999.0, I/O= 0.3,
TOTAL=999999.0
* A WORKING MEMORY= 6074952 SYSTEM( 66)= 0 METHOD 4 -PASSES= 1, CPU=999999.0, I/O= 0.8,
TOTAL=999999.0
* D TRANSPOSE FLAG= 0 SYSTEM(126)= 57 METHOD S -PASSES= 1, CPU= 0.0, I/O= 0.2, TOTAL= 0.2
METHOD S NT, NBR PASSES= 1, EST. CPU= 0.0, I/O= 0.2, TOTAL= 0.2
10:04:03 2:11 148.6 0.0 21.7 0.0 MPYAD BGN P=1
10:04:03 2:11 148.6 0.0# 21.7 0.0# MPYAD END

```

Listing 7-1 Example Execution Summary File (.f04) (continued)

```
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL  
SSG3 27 RUL 1 6 2 2 6 0.5000 1 3 1 0 0 0 3 3 0 *8**  
10:04:03 2:11 148.6 0.0# 21.7 0.1# STATRS 27 SSG3 END  
10:04:03 2:11 148.6 0.0 21.8 0.0 SESTATIC 74 DELETE BEGN  
10:04:03 2:11 148.6 0.0# 21.8 0.0# SESTATIC 74 DELETE END  
10:04:03 2:11 148.6 0.0 21.8 0.0 SESTATIC 75 DBSTATUSBEGN  
10:04:03 2:11 148.6 0.0# 21.8 0.0# SESTATIC 75 DBSTATUSEND  
10:04:03 2:11 148.6 0.0 21.8 0.0 SESTATIC 94 SESUM BEGN  
10:04:03 2:11 148.6 0.0# 21.8 0.0# SESTATIC 94 SESUM END  
10:04:04 2:12 148.6 0.0 21.9 0.0 SESUM 22 DELETE BEGN  
10:04:04 2:12 148.6 0.0# 21.9 0.0# SESUM 22 DELETE END  
10:04:04 2:12 148.9 0.3 21.9 0.0 SESUM 25 XPURGE BEGN  
10:04:04 2:12 148.9 0.0# 21.9 0.0# SESUM 25 XPURGE END  
10:04:04 2:12 149.1 0.2 21.9 0.0 SESTATIC 98 XEQUIV BEGN  
10:04:04 2:12 149.1 0.0# 21.9 0.0# SESTATIC 98 XEQUIV END  
10:04:04 2:12 149.1 0.0 21.9 0.0 SESTATIC 103 SUPER3 BEGN  
10:04:04 2:12 149.3 0.2# 22.0 0.1# SESTATIC 103 SUPER3 END  
10:04:04 2:12 149.3 0.0 22.0 0.0 SUPER3 26 XEQUIV BEGN  
10:04:04 2:12 149.3 0.0# 22.0 0.0# SUPER3 26 XEQUIV END  
10:04:04 2:12 149.3 0.0 22.0 0.0 SUPER3 28 XEQUIV BEGN  
10:04:04 2:12 149.3 0.0# 22.0 0.0# SUPER3 28 XEQUIV END  
10:04:05 2:13 149.3 0.0 22.2 0.2 SUPER3 61 XEQUIV BEGN  
10:04:05 2:13 149.3 0.0# 22.2 0.0# SUPER3 61 XEQUIV END  
10:04:05 2:13 149.3 0.1 22.2 0.0 SUPER3 67 SEP4 BEGN  
10:04:05 2:13 149.3 0.0# 22.2 0.0# SUPER3 67 SEP4 END  
10:04:05 2:13 149.3 0.0 22.3 0.1 SUPER3 92 SEDRDR BEGN  
10:04:05 2:13 149.3 0.0# 22.3 0.0# SUPER3 92 SEDRDR END  
10:04:05 2:13 149.3 0.0 22.3 0.0 SUPER3 99 PVT BEGN  
10:04:05 2:13 149.3 0.0# 22.4 0.1# SUPER3 99 PVT END  
10:04:05 2:13 149.4 0.1 22.4 0.0 SUPER3 106 SEDR BEGN  
10:04:05 2:13 149.4 0.0# 22.5 0.1# SUPER3 106 SEDR END  
10:04:05 2:13 149.4 0.0 22.5 0.0 SUPER3 112 XEQUIV BEGN  
10:04:05 2:13 149.4 0.0# 22.5 0.0# SUPER3 112 XEQUIV END  
10:04:05 2:13 149.4 0.0 22.5 0.0 SUPER3 117 SETQ BEGN  
10:04:05 2:13 149.4 0.0# 22.5 0.0# SUPER3 117 SETQ END  
10:04:05 2:13 149.4 0.0 22.6 0.0 SETQ 10 XEQUIV BEGN  
10:04:05 2:13 149.4 0.0# 22.6 0.0# SETQ 10 XEQUIV END  
10:04:05 2:13 149.4 0.0 22.7 0.2 SUPER3 135 DBSTATUSBEGN  
10:04:05 2:13 149.4 0.0# 22.7 0.0# SUPER3 135 DBSTATUSEND  
10:04:05 2:13 149.4 0.0 22.7 0.0 SUPER3 140 LCGEN BEGN  
10:04:05 2:13 149.4 0.0# 22.7 0.0# SUPER3 140 LCGEN END  
10:04:05 2:13 149.4 0.1 22.8 0.1 SUPER3 160 XEQUIV BEGN  
10:04:05 2:13 149.4 0.0# 22.8 0.0# SUPER3 160 XEQUIV END  
10:04:05 2:13 149.4 0.0 22.8 0.0 SUPER3 168 SEDISP BEGN  
10:04:05 2:13 149.6 0.2# 22.9 0.1# SUPER3 168 SEDISP END  
10:04:05 2:13 149.6 0.0 22.9 0.0 SEDISP 46 XEQUIV BEGN  
10:04:05 2:13 149.6 0.0# 22.9 0.0# SEDISP 46 XEQUIV END  
10:04:05 2:13 149.6 0.0 22.9 0.0 SEDISP 57 XEQUIV BEGN  
10:04:05 2:13 149.6 0.0# 23.0 0.0# SEDISP 57 XEQUIV END  
10:04:06 2:14 149.6 0.0 23.0 0.0 SEDISP 65 BCDR BEGN  
10:04:06 2:14 149.6 0.0# 23.0 0.0# SEDISP 65 BCDR END  
10:04:06 2:14 149.6 0.0 23.0 0.0 SEDISP 68 XEQUIV BEGN  
10:04:06 2:14 149.6 0.0# 23.0 0.0# SEDISP 68 XEQUIV END  
10:04:06 2:14 149.7 0.1 23.0 0.0 SEDISP 79 SEGOA BEGN  
10:04:06 2:14 149.8 0.1# 23.1 0.0# SEDISP 79 SEGOA END  
10:04:06 2:14 149.8 0.0 23.1 0.0 SEDISP 105 XEQUIV BEGN  
10:04:06 2:14 149.8 0.0# 23.1 0.0# SEDISP 105 XEQUIV END  
10:04:06 2:14 149.8 0.0 23.1 0.0 SEDISP 106 XEQUIV BEGN  
10:04:06 2:14 149.8 0.0# 23.1 0.0# SEDISP 106 XEQUIV END  
10:04:06 2:14 149.8 0.0 23.1 0.0 SEDISP 107 XEQUIV BEGN  
10:04:06 2:14 149.8 0.0# 23.1 0.0# SEDISP 107 XEQUIV END  
10:04:06 2:14 149.8 0.0 23.1 0.0 SEDISP 108 XEQUIV BEGN  
10:04:06 2:14 149.8 0.0# 23.1 0.0# SEDISP 108 XEQUIV END  
10:04:06 2:14 149.8 0.0 23.1 0.0 SEDISP 110 SDR1 BEGN  
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL  
SDR1 110 SCRATCH 1 12 2 1 6 0.5000 1 6 1 0 0 0 6 6 0 *8**  
SDR1 110 SCRATCH 1 12 2 2 10 0.4166 1 2 2 0 0 0 6 6 0 *8**  
SDR1 110 UGI 1 12 2 2 10 0.4166 1 2 2 0 0 0 6 6 0 *8**  
10:04:06 2:14 149.8 0.0# 23.2 0.0# SEDISP 110 SDR1 END  
10:04:06 2:14 149.8 0.0 23.2 0.0 SEDISP 135 COPY BEGN  
10:04:06 2:14 149.8 0.0# 23.2 0.0# SEDISP 135 COPY END  
10:04:06 2:14 149.8 0.0 23.2 0.0 SEDISP 139 COPY BEGN  
10:04:06 2:14 149.8 0.0# 23.2 0.0# SEDISP 139 COPY END  
10:04:06 2:14 149.8 0.0 23.2 0.0 SEDISP 141 COPY BEGN  
10:04:06 2:14 149.8 0.0# 23.2 0.0# SEDISP 141 COPY END  
10:04:06 2:14 149.8 0.0 23.2 0.0 SEDISP 213 SESUM BEGN  
10:04:06 2:14 149.8 0.0# 23.2 0.0# SEDISP 213 SESUM END  
10:04:06 2:14 149.8 0.0 23.3 0.0 SESUM 22 DELETE BEGN  
10:04:06 2:14 149.8 0.0# 23.3 0.0# SESUM 22 DELETE END  
10:04:06 2:14 150.1 0.3 23.3 0.0 SESUM 25 XPURGE BEGN  
10:04:06 2:14 150.1 0.0# 23.3 0.0# SESUM 25 XPURGE END  
10:04:06 2:14 150.2 0.1 23.3 0.0 SUPER3 172 XEQUIV BEGN  
10:04:06 2:14 150.2 0.0# 23.3 0.0# SUPER3 172 XEQUIV END  
10:04:06 2:14 150.3 0.1 23.4 0.0 SUPER3 181 XEQUIV BEGN  
10:04:06 2:14 150.3 0.0# 23.4 0.0# SUPER3 181 XEQUIV END  
10:04:06 2:14 150.3 0.0 23.4 0.0 SUPER3 190 XEQUIV BEGN  
10:04:06 2:14 150.3 0.0# 23.4 0.0# SUPER3 190 XEQUIV END
```

Listing 7-1 Example Execution Summary File (.f04) (continued)

```

10:04:07 2:15 150.3 0.0 23.4 0.0 SUPER3 203 SEDRCVR BEGN
10:04:07 2:15 150.4 0.2# 23.5 0.1# SUPER3 203 SEDRCVR END
10:04:07 2:15 150.4 0.0 23.5 0.0 SEDRCVR 31 XEQUIV BEGN
10:04:07 2:15 150.4 0.0# 23.5 0.0# SEDRCVR 31 XEQUIV END
10:04:07 2:15 150.4 0.0 23.5 0.0 SEDRCVR 32 XEQUIV BEGN
10:04:07 2:15 150.4 0.0# 23.5 0.0# SEDRCVR 32 XEQUIV END
10:04:07 2:15 150.4 0.0 23.5 0.0 SEDRCVR 36 VECPLOT BEGN
10:04:07 2:15 150.4 0.0# 23.6 0.0# SEDRCVR 36 VECPLOT END
10:04:07 2:15 150.4 0.0 23.6 0.0 SEDRCVR 37 VECPLOT BEGN
10:04:07 2:15 150.4 0.0# 23.6 0.0# SEDRCVR 37 VECPLOT END
10:04:07 2:15 150.4 0.0 23.6 0.0 SEDRCVR 38 VECPLOT BEGN
10:04:07 2:15 150.4 0.0 23.6 0.0 SEDRCVR 38 VECPLOT BEGN
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
VECPLOT 38 UGVV 1 12 2 1 5 0.4166 1 2 2 0 0 6 6 0 *8**
10:04:07 2:15 150.4 0.0# 23.6 0.0# SEDRCVR 38 VECPLOT END
10:04:07 2:15 150.4 0.0 23.6 0.0 SEDRCVR 39 VECPLOT BEGN
*8** MODULE DMAP MATRIX COLS ROWS F T NZWDS DENS BLOCKS STRL NBRSTR BNDL NBRBND ROW1 BNDAVG BNDMAX NULCOL
VECPLOT 39 PJB 1 12 2 1 3 0.2500 1 3 1 0 0 3 3 0 *8**
10:04:07 2:15 150.4 0.0# 23.6 0.0# SEDRCVR 39 VECPLOT END
10:04:07 2:15 150.4 0.0 23.7 0.0 SEDRCVR 48 SDR2 BEGN
10:04:08 2:16 150.4 0.0# 23.8 0.1# SEDRCVR 48 SDR2 END
10:04:08 2:16 150.5 0.1 23.8 0.0 SEDRCVR 49 SDR2 BEGN
10:04:08 2:16 150.5 0.0# 23.8 0.0# SEDRCVR 49 SDR2 END
10:04:08 2:16 150.5 0.0 23.8 0.0 SEDRCVR 58 XEQUIV BEGN
10:04:08 2:16 150.5 0.0# 23.8 0.0# SEDRCVR 58 XEQUIV END
10:04:08 2:16 150.5 0.0 23.8 0.0 SEDRCVR 59 XEQUIV BEGN
10:04:08 2:16 150.5 0.0# 23.8 0.0# SEDRCVR 59 XEQUIV END
10:04:08 2:16 150.5 0.0 23.8 0.0 SEDRCVR 60 XEQUIV BEGN
10:04:08 2:16 150.5 0.0# 23.8 0.0# SEDRCVR 60 XEQUIV END
10:04:08 2:16 150.5 0.0 23.8 0.0 SEDRCVR 61 XEQUIV BEGN
10:04:08 2:16 150.5 0.0# 23.8 0.0# SEDRCVR 61 XEQUIV END
10:04:08 2:16 150.6 0.1 23.9 0.1 SEDRCVR 125 SDRX BEGN
10:04:09 2:17 150.6 0.0# 23.9 0.0# SEDRCVR 125 SDRX END
10:04:09 2:17 150.6 0.0 23.9 0.0 SEDRCVR 129 XEQUIV BEGN
10:04:09 2:17 150.6 0.0# 23.9 0.0# SEDRCVR 129 XEQUIV END
10:04:09 2:17 150.6 0.0 23.9 0.0 SEDRCVR 130 XEQUIV BEGN
10:04:09 2:17 150.6 0.0# 23.9 0.0# SEDRCVR 130 XEQUIV END
10:04:09 2:17 150.6 0.0 24.0 0.0 SEDRCVR 132 XEQUIV BEGN
10:04:09 2:17 150.6 0.0# 24.0 0.0# SEDRCVR 132 XEQUIV END
10:04:09 2:17 150.6 0.0 24.0 0.0 SEDRCVR 133 XEQUIV BEGN
10:04:09 2:17 150.6 0.0# 24.0 0.0# SEDRCVR 133 XEQUIV END
10:04:09 2:17 150.7 0.1 24.0 0.0 SEDRCVR 205 OFF BEGN
10:04:10 2:18 150.7 0.0# 24.1 0.1# SEDRCVR 205 OFF END
10:04:10 2:18 150.7 0.0 24.1 0.0 SEDRCVR 206 OFF BEGN
10:04:10 2:18 150.7 0.0# 24.1 0.0# SEDRCVR 206 OFF END
10:04:10 2:18 150.7 0.0 24.1 0.0 SEDRCVR 208 OFF BEGN
10:04:10 2:18 150.7 0.0# 24.1 0.0# SEDRCVR 208 OFF END
10:04:11 2:19 150.7 0.0 24.1 0.0 SEDRCVR 211 SDR2 BEGN
10:04:11 2:19 150.7 0.0# 24.2 0.0# SEDRCVR 211 SDR2 END
10:04:11 2:19 150.7 0.0 24.2 0.0 SEDRCVR 212 SDRCOMP BEGN
10:04:11 2:19 150.7 0.0# 24.2 0.0# SEDRCVR 212 SDRCOMP END
10:04:11 2:19 150.7 0.0 24.2 0.0 SEDRCVR 213 STRSORT BEGN
10:04:11 2:19 150.7 0.0# 24.2 0.0# SEDRCVR 213 STRSORT END
10:04:13 2:21 150.8 0.1 24.2 0.0 SEDRCVR 214 STRSORT BEGN
10:04:13 2:21 150.8 0.0# 24.2 0.0# SEDRCVR 214 STRSORT END
10:04:13 2:21 150.8 0.0 24.3 0.0 SEDRCVR 215 OFF BEGN
10:04:13 2:21 150.8 0.0# 24.3 0.0# SEDRCVR 215 OFF END
10:04:13 2:21 150.8 0.0 24.3 0.0 SEDRCVR 222 XEQUIV BEGN
10:04:14 2:22 150.8 0.0# 24.3 0.0# SEDRCVR 222 XEQUIV END
10:04:14 2:22 150.8 0.0 24.3 0.0 SEDRCVR 224 OFF BEGN
10:04:14 2:22 150.8 0.0# 24.3 0.0# SEDRCVR 224 OFF END
10:04:15 2:23 150.9 0.1 24.5 0.2 SESTATIC 132 PRTSUM BEGN
10:04:15 2:23 150.9 0.0# 24.5 0.0# SESTATIC 132 PRTSUM END
10:04:15 2:23 150.9 0.0 24.5 0.0 PRTSUM 4 PROJVER BEGN
10:04:15 2:23 150.9 0.0# 24.5 0.0# PRTSUM 4 PROJVER END
10:04:15 2:23 150.9 0.0 24.5 0.0 PRTSUM 5 DBDICT BEGN
10:04:15 2:23 150.9 0.0# 24.6 0.1# PRTSUM 5 DBDICT END
10:04:15 2:23 150.9 0.0 24.7 0.0 PRTSUM 6 PRTFARM BEGN
10:04:15 2:23 150.9 0.0# 24.7 0.0# PRTSUM 6 PRTFARM END
10:04:15 2:23 150.9 0.0 24.7 0.0 SESTATIC 133 EXIT BEGN

```

Listing 7-1 Example Execution Summary File (.f04) (continued)

1*** USER INFORMATION MESSAGE 1009 (F04SUM)
STATISTICAL SUMMARY OF ABOVE LOG FILE FOLLOWS.

19

STATISTICS FOR INDIVIDUAL EVENTS

LINE	DMAP	EVNT	METH	EST CPU	ACT CPU	DEL CPU	% DIF	<25%	<50%	>50%	% TOT
65	35	XSOR			1.						5.
67	61	IFP			0.						1.
120	154	PLTS			0.						0.
124	176	SEQP			0.						1.
126	182	GP1			0.						1.
132	25	GP1			0.						1.
134	28	SEP1			0.						0.
140	46	TABP			0.						0.
142	47	SEP3			0.						0.
172	307	GP1			0.						1.
180	6	GP2			0.						0.
194	335	GP3			0.						0.
196	336	LCGE			0.						0.
202	367	GP4			0.						0.
208	378	UMER			0.						0.
218	45	MATG			0.						0.
237	24	TAL			0.						0.
241	16	ELTP			0.						0.
243	22	EMG			0.						0.
249	79	EMA			0.						0.
278	11	SSG1			0.						0.
287	62	VECP			0.						0.
319	29	GFSP			0.						0.
327	47	UPAR			0.						0.
371	41	UPAR			0.						0.
377	64	SSG2			0.						0.
410	18	DCMP			0.						0.
459	27	FBS	S 2	0.	0.	0.	0.	1	0	0	0.
459	27	MPYA	S NT	0.	0.	0.	0.	1	0	0	0.
459	27	SSG3			0.						0.
498	67	SEP4			0.						0.
504	106	SEDR			0.						0.
514	140	LCGE			0.						0.
538	110	SDR1			0.						0.
568	36	VECP			0.						0.
570	37	VECP			0.						0.
572	38	VECP			0.						0.
576	39	VECP			0.						0.
580	48	SDR2			0.						0.
582	49	SDR2			0.						0.
592	125	SDRX			0.						0.
602	205	OFFP			0.						0.
604	206	OFFP			0.						0.
606	208	OFFP			0.						0.
608	211	SDR2			0.						0.
610	212	SDRC			0.						0.
612	213	STRS			0.						0.
614	214	STRS			0.						0.
616	215	OFFP			0.						0.
620	224	OFFP			0.						0.

STATISTICS FOR EVENTS BY METHOD

	COUNT	EST CPU	ACT CPU	DEL CPU	<25%	<50%	>50%	A.M. %	S.D. %	% TOT
MPYAD S NT	1	0.	0.	0.	1	0	0	0.	0.	0.
MPYAD N TOTALS	1	0.	0.	0.						0.
FBS S 2	1	0.	0.	0.	1	0	0	0.	0.	0.
FBS TOTALS	1	0.	0.	0.						0.

METHOD SUMMARY:

TOTAL OF EVENTS RECORDED = 0.
TOTAL CPU = 25.
PERCENT RECORDED = 0.

Standard Output Contents

The Standard Output contains the MSC.Nastran analysis results. This file also contains information about the version of MSC.Nastran being run.

Listing 7-2 MSC.Nastran Standard Output

THIS PROGRAM IS CONFIDENTIAL AND A TRADE SECRET OF THE MACNEAL-SCHWENDLER CORP. THE RECEIPT OR POSSESSION OF THIS PROGRAM DOES NOT CONVEY ANY RIGHTS TO REPRODUCE OR DISCLOSE ITS CONTENTS, OR TO MANUFACTURE, USE, OR SELL ANYTHING HEREIN, IN WHOLE OR IN PART, WITHOUT THE SPECIFIC WRITTEN CONSENT OF THE MACNEAL-SCHWENDLER CORPORATION.

```

*****
*****
*
*
* THE *
* MACNEAL-SCHWENDLER *
* CORPORATION *
*
* M S C / N A S T R A N *
*
* VERSION - 68.0 *
*
* MAR 3, 1994 *
*
* DIGITAL EQUIPMENT *
*
* VAX 6000-410 *
*
* VMS V5.4-3 *
*
*
*
*****
*****

```

N A S T R A N S Y S T E M P A R A M E T E R E C H O MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 1

```

$ Welcome to MSC/NASTRAN Version 68
$
$ MSC/NASTRAN Version 68 adds the following major new capabilities:
$
$ * P-version elements and adaptivity
$ * Shape optimization
$ * Dynamic and acoustic optimization
$ * Superelement optimization
$ * Aeroelastic optimization
$ * 3D slideline contact
$ * Additional hyperelastic elements
$ * Improved differential stiffness
$ * Substantially enhanced heat transfer
$ * Multiple boundary conditions in SOLs 101, 103, and 200
$ * Performance enhancements
$ * ... and other enhancements
$
$ See the "Version 68 Release Notes" for a complete description of the
$ Version 68 capabilities.
$
$ CALENDAR OF EVENTS:
$
$ World Users' Conference June 20-24, 1994
$ (Orlando, Florida)
$
$ Contact your MSC representative for more information about the
$ conferences and training seminars.
$
$
$ This "news" information can be turned off by changing news=yes to news=no
$ in the system runtime configuration (rc) file. Once turned off in the
$ system rc file, it can be turned on by setting news=yes in the MSC/NASTRAN
$ command line or local rc file.
$ NASTRAN SYSTEM(20)=0 $ PRINT ALL MODULES EXECUTED IN EXECUTION SUMMARY

```

Listing 7-2 MSC.Nastran Standard Output (continued)

```

MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 2
N A S T R A N   E X E C U T I V E   C O N T R O L   D E C K   E C H O

ID MSC, AM761B $ JJG 5MAR91
DIAG 8,13,19,49 $
TIME 2
SOL 101 $ STRUCTURED STATICS
CEND

EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 3

C A S E   C O N T R O L   D E C K   E C H O

CARD
COUNT
1      TITLE = EXAMPLE: ONE ELEMENT PROBLEM                AM761B
2      LOAD = 11
3      DISPLACEMENT = ALL
4      ELFORCE = ALL
5      BEGIN BULK
      INPUT BULK DATA CARD COUNT =          7

EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 4

S O R T E D   B U L K   D A T A   E C H O

CARD
COUNT      . 1 .. 2 .. 3 .. 4 .. 5 .. 6 .. 7 .. 8 .. 9 .. 10 .
1-   CBEAM  1      1      10      20      0.      1.      0.
2-   FORCE   11     20     100.    1.      .8      1.
3-   GRID   10     10     0.      0.      0.      123456
4-   GRID   20     20     10.     0.      0.
5-   MAT1   100    1.+7   .3
6-   PBEAM  1      100    1.      .08     .064    .1
      ENDDATA
      TOTAL COUNT=          7

EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 5

S E Q U E N C E   P R O C E S S O R   O U T P U T

THERE ARE          2 POINTS DIVIDED INTO    1 GROUP(S).
CONNECTION DATA
ELEMENT TYPE      NUMBER      ASSEMBLY TIME(SEC)
-----
BEAM              1              0.02
-----

TOTAL MATRIX ASSEMBLY TIME FOR          1 ELEMENTS IS    0.02 SECONDS.
ORIGINAL PERFORMANCE DATA
SUPER(GROUP) ID  NO. GRIDS  AV. CONNECTIVITY  C-AVERAGE  C-RMS  C-MAXIMUM  P-GROUPS  P-AVERAGE  DECOMP TIME(SEC)
(6.0 DOF/GRID )
0                2          2.00             1.50       1.58       2          0          0.00       0.000

RESEQUENCED PERFORMANCE DATA
SUPER(GROUP) ID  NO. GRIDS  AV. CONNECTIVITY  C-AVERAGE  C-RMS  C-MAXIMUM  P-GROUPS  P-AVERAGE  DECOMP TIME(SEC)  METHOD
(6.0 DOF/GRID )
0                2          2.00             1.50       1.58       2          0          0.00       0.000      ACTIVE

EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 6

OLOAD      RESULTANT
      T1      T2      T3      R1      R2      R3
1  1.0000000E+02  8.0000000E+01  1.0000000E+02  0.0000000E+00 -1.0000000E+03  8.0000000E+02
*** USER INFORMATION MESSAGE 5293 FOR DATA BLOCK KLL
LOAD SEQ. NO.      EPSILON      EXTERNAL WORK      EPSILONS LARGER THAN 0.001 ARE FLAGGED WITH ASTERISKS
1                -4.6914561E-17    3.9638200E+00
EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 7

MAXIMUM DISPLACEMENTS
      T1      T2      T3      R1      R2      R3
1  9.9999997E-05  3.3541333E-02  5.2343331E-02  0.0000000E+00  7.8124995E-03  4.9999999E-03
EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 8

MAXIMUM APPLIED LOADS
      T1      T2      T3      R1      R2      R3
1  1.0000000E+02  8.0000000E+01  1.0000000E+02  0.0000000E+00  0.0000000E+00  0.0000000E+00

```


Listing 7-2 MSC.Nastran Standard Output (continued)

```

EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 9

                                D I S P L A C E M E N T   V E C T O R

POINT ID.  TYPE      T1          T2          T3          R1          R2          R3
    10      G         0.0         0.0         0.0         0.0         0.0         0.0
    20      G      1.000000E-04    3.354133E-02    5.234333E-02    0.0        -7.812500E-03    5.000000E-03
EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 10

                                F O R C E S   I N   B E A M   E L E M E N T S   ( C B E A M )
ELEMENT-ID  GRID  STAT DIST/  - BENDING MOMENTS -  - WEB SHEARS -  AXIAL  TOTAL  WARPING
            LENGTH  PLANE 1  PLANE 2  PLANE 1  PLANE 2  FORCE  TORQUE  TORQUE
    1
    10      0.000    8.000000E+02    1.000000E+03    8.000000E+01    1.000000E+02    1.000000E+02    0.0        0.0
    20      1.000      0.0          0.0          8.000000E+01    1.000000E+02    1.000000E+02    0.0        0.0
EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 11

EXAMPLE: ONE ELEMENT PROBLEM                AM761B                MARCH 11, 1994 MSC/NASTRAN 3/ 3/94 PAGE 12

* * * * D B D I C T   P R I N T   * * * *          SUBDMAP = PRTSUM , DMAP STATEMENT NO.          5

                                * * * * A N A L Y S I S   S U M M A R Y   T A B L E   * * * *
SEID PEID PROJ VERS APRCH          SEMG SEMR SEKR SELG SELR MODES DYNRED SOLLIN PVALID SOLNLL LOOPID DESIGN CYCLE SENSITIVITY
-----
0      0      1      1      '      T      F      T      T      T      F      F      T      0      F      -1      0      F
0SEID = SUPERELEMENT ID.
PEID = PRIMARY SUPERELEMENT ID OF IMAGE SUPERELEMENT.
PROJ = PROJECT ID NUMBER.
VERS = VERSION ID.
APRCH = BLANK FOR STRUCTURAL ANALYSIS. HEAT FOR HEAT TRANSFER ANALYSIS.
SEMG = STIFFNESS AND MASS MATRIX GENERATION STEP.
SEMR = MASS MATRIX REDUCTION STEP (INCLUDES EIGENVALUE SOLUTION FOR MODES).
SEKR = STIFFNESS MATRIX REDUCTION STEP.
SELG = LOAD MATRIX GENERATION STEP.
SELR = LOAD MATRIX REDUCTION STEP.
MODES = T (TRUE) IF NORMAL MODES OR BUCKLING MODES CALCULATED.
DYNRED = T (TRUE) MEANS GENERALIZED DYNAMIC AND/OR COMPONENT MODE REDUCTION PERFORMED.
SOLLIN = T (TRUE) IF LINEAR SOLUTION EXISTS IN DATABASE.
PVALID = P-DISTRIBUTION ID OF P-VALUE FOR P-ELEMENTS
LOOPID = THE LAST LOOPID VALUE USED IN THE NONLINEAR ANALYSIS. USEFUL FOR RESTARTS.
SOLNLL = T (TRUE) IF NONLINEAR SOLUTION EXISTS IN DATABASE.
DESIGN CYCLE = THE LAST DESIGN CYCLE (ONLY VALID IN OPTIMIZATION).
SENSITIVITY = SENSITIVITY MATRIX GENERATION FLAG.

                                * * * * E N D   O F   J O B   * * * *

```

The Grid Point Singularity Table (GPST) is automatically output following the execution of the Grid Point Singularity Processor (GPSP) if singularities remain in the stiffness matrix at the grid point level. The table lists singular degrees-of-freedom in the global coordinate system. These singular degrees-of-freedom may be automatically constrained, at the user's option. Further information on this topic is given in [“Constraint and Mechanism Problem Identification in SubDMAP SEKR”](#) on page 383.

Any of the matrices or tables that are prepared by the functional modules can be printed by using selected utility modules described in the [MSC.Nastran 2005 DMAP Programmer's Guide](#). These utility modules can be scheduled at any point in a solution sequence by using the alter feature. In general, they should be scheduled immediately after the functional module that generates the table or matrix to be printed. However, the user is cautioned to check the calling sequence for the utility module, in order to be certain that all required inputs have been generated prior to this point.

The following descriptions relate to automatic output that is associated with particular types of solution sequences.

Real Eigenvalue Analysis

Each eigenvalue is identified with a mode number determined by sorting the eigenvalues by their magnitude. The following summary is also automatically printed for all methods:

- Mode number
- Extraction order (this is irrelevant under the Lanczos method)
- Eigenvalue
- Frequency (radians)
- Frequency (Hz)
- Generalized mass
- Generalized stiffness

For all methods except the Inverse Power Method (see below), if the orthogonality criterion is failed, then User Warning Message 3034 is issued:

The following summary is also automatically printed when using the Lanczos method:

- Block size used
- Number of decompositions
- Number of roots found
- Number of (block) solves required
- One of the following termination messages:
 - Required number of eigenvalues found
 - Not all the roots found within the range
 - All eigenvalues found within the range
 - Insufficient time to find more eigenvalues

The following summary is automatically printed when using the Inverse Power Method:

- Number of eigenvalues extracted
- Number of starting points used

- Number of starting point moves
- Number of triangular decompositions
- Number of vector iterations
- The reason for termination is indicated by one of the following numbers:

Reason Number	Description
1	Two consecutive singularities encountered while performing triangular decomposition
2	Four shift points while tracking a single root
3	All eigenvalues found in the frequency range specified
4	Three times the number of roots estimated in the frequency range have been extracted
5	All eigenvalues that exist in the problem have been found
6	The number of roots desired have been found
7	One or more eigenvalues have been found outside the frequency range specified
8	Insufficient time to find another root
9	Unable to converge

- Largest off-diagonal modal mass term identified by mode pair
- Number of off-diagonal mass terms failing orthogonality criterion

Complex Eigenvalue Analysis

Each complex eigenvalue is identified with a root number determined by sorting the complex eigenvalues by magnitude. The following summary of the complex eigenvalues extracted is automatically printed for all methods:

- Root number
- Extraction order (this is irrelevant under the Lanczos method)
- Real and imaginary parts of the eigenvalue
- The coefficients f (frequency) and g (damping coefficient) in the following representation of the eigenvalue ($p = \alpha + i\omega$):

$$p = 2\pi f(i - 1/2g) \quad \text{Eq. 7-1}$$

Therefore,

$$f = \left| \frac{\omega}{2\pi} \right| \quad \text{Eq. 7-2}$$

$$g = \frac{-2\alpha}{|\omega|} = 2 \frac{C}{C_{cr}} \quad \text{Eq. 7-3}$$

When using the complex Lanczos method, User Information Messages 5444, 5445, and 5453 are also issued. The following summary is also automatically printed when the Determinant Method is used:

- Number of eigenvalues extracted
- Number of passes through starting points.
- Number of criteria changes
- Number of starting point moves
- Number of triangular decompositions
- Number of failures to iterate to a root
- Number of predictions outside region
- The reason for termination is indicated by one of the following integers:

Reason Number	Description
1	The number of roots desired have been found
2	All predictions for eigenvalues are outside the regions specified
3	Insufficient time to find another root
4	Matrix is singular at the first three starting points

- Swept determinant functions for each starting point

The following summary is automatically printed for each region specified when the Inverse Power Method is used:

- Number of eigenvalues extracted
- Number of starting points used
- Number of starting point or shift point moves
- Number of triangular decompositions

- Number of vector iterations
- The reason for termination is indicated by one of the following integers:

Reason Number	Description
1	Two consecutive singularities encountered while performing triangular decomposition
2	Four starting point moves while tracking a single root
3	All eigenvalues found in the region specified
4	Three times the number of roots estimated in the region have been extracted
5	All eigenvalues that exist in the problem have been found
6	The number of roots desired have been found
7	One or more eigenvalues have been found outside the region specified
8	Insufficient time to find another root
9	Unable to converge

Superelement Analysis

In addition to the previously mentioned automatic output, the superelements solution sequences also print the following:

- Superelement mapping information that lists the membership of grid points and elements in the superelements, the order in which the superelements will be processed, and timing and space estimates for each superelement
- A page header that identifies the superelement for which the printed information on the page pertains
- Matrices wherein mechanisms are identified
- Resultants of external loads
- Maximum applied loads
- Maximum displacements
- Maximum SPC forces

For SOL 101, if PARAM,INREL,-1 or PARAM,INREL,-2, the following data is printed automatically:

INTERMEDIATE MATRIX...QRR

This is the rigid body mass matrix of the total structure, including superelements. It is presented as a 6×6 matrix, measured at the reference grid point, in its global coordinates. Masses on scalar points are ignored.

INTERMEDIATE MATRIX...QRL

This is the resultant of the applied loads, measured at the reference point. There is one column for each loading condition.

INTERMEDIATE MATRIX...URACCEL

This is the user-supplied rigid body acceleration, input on the DMIG,UACCEL entry. This output does not appear if the entry is not present.

INTERMEDIATE MATRIX...URA

This is the rigid body acceleration matrix, computed from the applied loads.

7.3 Restart Procedures

Scheduled exits can be requested at any point in a structured solution sequence format by means of the SEMG, SEKR, SELG, SELR, SEMR, and SEALL Case Control commands. If none of these commands are specified, then SEALL = ALL is the default.

Unscheduled exits are usually caused by errors on input entries or errors in the structural model resulting from missing or inconsistent input data. When such errors are detected, an unscheduled exit is performed accompanied with the output of the applicable user error messages. Following the correction of the input data errors, a modified restart can be performed.

Unscheduled exits may also occur because of machine failure or insufficient time allowance. In these cases, an unmodified restart is usually made. In some cases, where a portion of the problem has been completed, including the output for the completed portion, a modified restart must be made following an unscheduled exit due to insufficient time allowance. The situations are discussed under case control requirements in the sections dealing with the individual solution sequences.

The initial execution of any problem must be made with complete MSC.Nastran input data, including all of the bulk data. However, all or part of the bulk data may be assembled from alternate input sources.

For restart, the Bulk Data Section consists only of delete “/” entries (see “[Bulk Data Section](#)” on page 34) and new entries that the user wishes to add. The previous Bulk Data is read from the database. All other parts of the MSC.Nastran Input Data (including the File Management statements, the Executive Control statements, the Case Control commands, the BEGIN BULK command and the ENDDATA entry) must be resubmitted even though no changes are made in the case control and no new bulk data is added. When changing solution sequences, the solution number (SOL) must be changed to the number or name of the new solution sequence.

Any changes in the Case Control Section associated with Bulk Data entry selection or subcase definition, or changes in the Bulk Data Section.

Restarts always start at the beginning of a specified DMAP or SOL sequence and queries the database in two phases.

Phase 1: Conceptually this phase marks all appropriate existing database data blocks and parameters as existing for the current restart run.

Phase 2: This phase then conceptually checks the current input against the version from which the restart is starting from and deletes from the Phase 1 determined database data blocks and parameters any data blocks that were modified because of input. This phase is performed by the RESTART module(s) contained within the solution's DMAP.

Restart only executes DMAP Modules for which some or all of the output does not exist on the current version database. The S-type parameter is considered as an output data block for restart purposes, hence if it is not listed in the NDDL, then the module that contains it will be re-executed. All DMAP modules are executed until the first RESTART module after which output checking is performed. Forced execution of modules after this point may be manually controlled by SYSTEM (109) flag.

If no such changes are made, the executive system performs an unmodified restart. If changes have been made only in the output requests, the restart is considered unmodified. However, some modules may have to be re-executed in order to prepare the output. Output requests that were satisfied, plots made, etc., should be removed if they are not to be output again.

For modified restarts, a number of previously executed DMAP instructions may have to be re-executed, depending on the nature of the modifications made by the user. The DMAP instructions that need to be executed in a modified restart are automatically determined within the program by comparing all changes made in Case Control commands and Bulk Data entries.

7.4 General Operations

This section describes the data processing and matrix operations that are performed by the functional modules in the solution sequences (SOLs 101 through 200) unless otherwise indicated. Detailed descriptions of the functional modules are given in Chapter 4 or the *MSC.Nastran Programmers Manual*. Additional information regarding the use of DMAP modules for matrix operations is given in the *MSC.Nastran 2005 DMAP Programmer's Guide*.

The following sections will refer to the structural analysis matrices such as stiffness, damping, and mass, but the discussion also applies to heat transfer analysis for conductance and capacitance. There is no analogy for mass in heat transfer analysis.

7.4.1 Geometry Processing in SubDMAP PHASE0

The total model is given its initial processing in PHASE0. Data is taken from the input files and stored in data structures that feed more expensive processes in later SubDMAPS. Flags are set to control branching for special features. All superelement inputs are processed. Tables are indexed for faster processing in post-processors. Parameters are read from Case Control and Bulk Data files by module PVT. The shell element normals at each grid point are computed in module TASN2, and output in table GPSNT.

Tables for internal program use are generated by the geometry processors from the grid point entries, coordinate system definition entries, sequence entries, connection entries, static load entries, temperature definition entries, and constraint entries. The computer time for all geometry processing is usually only a small part of the time required for problem solution.

The basic geometry processing is performed by GP1 (Geometry Processor -- Phase 1). A list of all grid points and scalar points is assembled in internal order for use in relating internal and external identification numbers. Matrices for transformation from basic to all defined global coordinate systems are computed, and all grid point locations are transformed to the basic coordinate system.

The element connection tables (ECT) are generated by GP2 (Geometry Processor -- Phase 2) from the connection entries. The external grid point numbers are converted to the internal indices defined in GP1.

The temperature data and static loads are processed by GP3 (Geometry Processor -- Phase 3). The temperature data is collected by sets and the external grid point numbers and element numbers are converted to the internal indices defined in GP1 and GP2. In the case of static analysis, the static loads are also collected by sets and the external identification numbers are converted to internal indices.

The constraint data is processed by GP4 (Geometry Processor -- Phase 4). The displacement set definition table (USET) is generated from the constraint definition entries. USET contains one coded word for each degree-of-freedom in the model to identify the set or sets of coordinates to which the degree-of-freedom belongs (see “[Constraint and Set Notation](#)” on page 321 for the definition of degree-of-freedom sets). The multipoint constraint matrix $[R_{mg}]$ is formed from the MPC and rigid element Bulk Data entries. In static analysis, the enforced displacement matrix $[Y_s]$ is formed for use in the generation of equivalent static loads.

The BNDSPC module moves SPCs on boundary grid points to downstream superelements.

Model Checkout

A checkout procedure is provided for the development of new models. It includes the bookkeeping and control blocks, and part of the generation block of [Figure 7-3](#) in “[Superelement Analysis](#)” on page 444. It is requested by PARAM,CHECKOUT,YES.

SEP1 Module

Generates the SEMAP table, which lists superelement membership of grid points and elements. Provides estimates for computation costs and storage requirements. This module will identify any structural partitioning errors.

There are two paths through the superelement capability. If the bulk data entries have a BEGIN SUPER and/or BEGIN SUPER = [SEID] entries the data flow goes through the partitioned superelement capability modules, SEP1X and SEP2X. If only a BEGIN BULK entry is present the grid point list capability modules SEP1 and SEP2 are used. The SEP1X path also allows grid point list superelements, with the exception of fluid structure analysis, SOL 200 (Design Optimization), and the RELEASE entry. Models using these features must use the grid point list path. More details are given in “[Superelement Analysis](#)” on page 444.

SEP2, SEP2CT Modules

Partitions out the data for any superelement selected by SE-type Case Control commands. These modules identify load and rigid element partitioning errors, and errors in the Case Control Section.

Constraint Modules

The GP1 and GP4 modules process all selected superelements. They check for the presence of requested coordinate systems; for illegal duplicate specifications of mutually exclusive sets, such as SPCi and ASETi entries referencing the same point; and for correct specifications on SPCi, xSETi, and MPCi entries, and rigid elements.

The user set table may be printed by use of PARAM, USETPRT, when PARAM, CHECKOUT, YES is present. If rigid elements or selected MPC entries exist, the constraint matrix $[R_{mg}]$ (see “[Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB](#)” on page 379) is given three tests:

1. A matrix of rigid body vectors $[u_{gh}^o]$ is calculated by the VECPLOT module. This matrix includes zeroes for the rows corresponding to scalar points. The product

$$[E_{mh}] \leftarrow [R_{mg}][u_{gh}^o]$$

is calculated. Terms of $[E_{mh}]$ larger than PARAM, TINY are printed. If the constraint matrix contains internal constraints, nonzero terms will exist in $[E_{mh}]$. If an MPC equation includes scalar points, nonzero terms may exist in the corresponding row of $[E_{mh}]$ although the equation may be in equilibrium.

2. The product

$$[R_{mm}^g] \leftarrow [R_{mg}][R_{mg}]^T$$

is calculated. The product matrix is decomposed by the DCMP module. Diagonal terms of the factor matrix that are larger than PARAM, MAXRATIO are printed. These terms indicate that the printed row is linearly dependent on prior rows. Unless the constraint matrix is changed, it is likely that the model will encounter fatal errors due to singularity in the MCE1 module, or will have a poorly-conditioned constraint elimination process.

3. The product

$$[R_{mm}^m] \leftarrow [R_{mm}][R_{mm}]^T$$

is calculated, where $[R_{mm}]$ is the partition of $[R_{mg}]$ described in [Section 7.4.3](#). This product is also processed by the DCMP module. If a row appears here that also appears in the messages from the prior step, it confirms that the constraint matrix is inherently ill-conditioned. If a row appears here that did not appear in the prior step, it indicates that a change in the m-set exists that can also make the $[R_{mm}]$ matrix well conditioned.

The DBC module stores data for use in postprocessors. It runs at many points in the Solution Sequences. In this SubDMAP it stores basic structural geometric data. The AEROE SubDMAP generates the geometry of the aerodynamic model. The Automated Component Mode Synthesis (ACMS) capability controls branches at many places in the Solution Sequences. The RESTART module controls the modules that will be run on restart, based on changes in the input files. The run exits after these checks are performed.

7.4.2 Matrix Assembly Operations in SubDMAP SEMG

The stiffness, mass, damping, differential stiffness, and heat conduction matrices are formed during the matrix assembly operations in subDMAP SEMG.

Stiffness is generated from the following sources:

- Structural elements defined on connection entries (e.g., CBAR and CROD).
- General elements defined on GENEL entries.
- Scalar springs defined on CELASi entries.

Mass is generated from the following sources:

- A 6×6 matrix of mass coefficients at a grid point defined on a CONM1 entry.
- A concentrated mass element defined on a CONM2 entry in terms of its mass and moments of inertia about its center of gravity.
- Structural mass for all elements, except plate elements without membrane stiffness, using the mass density on the material definition entry.
- Nonstructural mass for all elements specifying a value on the property entry.
- Scalar masses defined on CMASSi entries.

Damping is generated from the following sources:

- Viscous rod elements defined on CVISC entries.
- Scalar viscous dampers defined on CDAMPi entries.

- Element structural damping by multiplying the stiffness matrix of an individual structural element by a damping factor obtained from the material properties (MATi) entry for the element.

These matrices are formed from the connection and geometric tables prepared by the geometry processors and the material and element property tables prepared in the Phase 0. The matrix assembly operations will usually represent a significant portion of the total solution time. The computer time for matrix assembly operations is linear with the number and type of elements. Unit times for stiffness matrix assembly are given in the *MSC.Nastran 2004 Installation and Operations Guide*. Assembly times for coupled mass matrices, differential stiffness matrices, and heat conduction matrices are about the same as those for stiffness matrices. Assembly times for lumped mass matrices and damping matrices are usually relatively small.

The matrix assembly process begins with the generation of various tables in TA1 (Table Assembler) from the geometric data, connection data and property data. These tables are used in data recovery operations as well as in the matrix assembly operations.

The matrix assembly operations are performed by EMG (Element Matrix Generator) and EMA (Element Matrix Assembler) modules. All of the element matrices (KELM, KDICT, MELM, MDICT, BELM, and BDICT) are generated by EMG. The stiffness matrix exclusive of general elements, $[K_{gg}^x]$, as well as the structural damping matrix, $[K_{gg}^4]$, the mass matrix, $[M_{gg}]$, and the viscous damping matrix, $[B_{gg}]$, are assembled in separate executions of EMA.

The parameter COUPMASS is used to select either coupled mass or lumped mass generation in EMG. If the parameter WTMASS is present, the mass matrices generated in EMG are multiplied term by term by the value in field 3 of the PARAM entry. This operation allows the user to specify mass data in units of weight.

The structural damping matrix $[K_{gg}^4]$ is assembled from elements with structural damping. Structural damping is specified in the GE field of the MATi entry.

$$[K_{gg}^4] = \sum_{e=1}^{NE} GE_e [K_e] \quad \text{Eq. 7-4}$$

where:

NE = number of elements

GE_e = element structural damping coefficient from MATi entries

$[K_e]$ = element stiffness matrix

The direct matrix inputs requested by the K2GG, M2GG, and B2GG Case Control commands defined the stiffness matrix $[K_{gg}^2]$, mass matrix $[M_{gg}^2]$, and damping matrix $[B_{gg}^2]$, and are assembled by the MTRXIN module. The outputs from EMA and MTRXIN are combined by the ADD module.

The stiffness contributions from the general elements (GENEL Bulk Data entry) are assembled in SMA3 (Structural Matrix Assembler -- Phase 3). The final stiffness matrix $[K_{gg}]$ is formed by combining $[K_{gg}^x]$ prepared by the EMA module with the contributions from general elements generated by SMA3.

Weight and balance information is calculated from the mass matrix $[M_{gg}]$ by GPWG (Grid Point Weight Generator). The execution of GPWG is controlled by the parameter GRDPNT. The details of the output are described in “**GRDPNT**” on page 648 of the *MSC.Nastran Quick Reference Guide*.

7.4.3 Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB

The multipoint constraint operations apply constraints to the stiffness matrices using the linear relationships among the displacements specified by the user on MPC, MPCAX, and rigid element Bulk Data entries.

The multipoint constraint equations are initially expressed in the form,

$$[R_g]\{u_g\} = 0, \quad \text{Eq. 7-5}$$

where the coefficients are supplied by the user. The user also specifies the degree-of-freedom that is made dependent by each equation of constraint, so that the $\{u_g\}$ matrix may immediately be partitioned into two subsets,

$$\{u_g\} = \begin{Bmatrix} u_n \\ u_m \end{Bmatrix}, \quad \text{Eq. 7-6}$$

where the set, u_m , is the set of dependent degrees-of-freedom. The matrix of constraint coefficients is similarly partitioned

$$[R_g] = [R_n \mid R_m], \quad \text{Eq. 7-7}$$

so that Eq. 7-5 becomes

$$[R_n]\{u_n\} + [R_m]\{u_m\} = \mathbf{0} . \quad \text{Eq. 7-8}$$

$[R_m]$ is a nonsingular matrix. We can, therefore, form the multipoint constraint matrix,

$$[G_m] = -[R_m]^{-1}[R_n] , \quad \text{Eq. 7-9}$$

so that Eq. 7-7 may be stated as

$$\{u_m\} = [G_m]\{u_n\} . \quad \text{Eq. 7-10}$$

Prior to the imposition of constraints, the structural problem may be written as

$$[K_{gg}]\{u_g\} = \{P_g\} , \quad \text{Eq. 7-11}$$

or, partitioning in terms of the coordinate sets, u_n and u_m

$$\begin{bmatrix} \bar{K}_{nn} & | & K_{nm} \\ \hline K_{nm}^T & | & K_{mm} \end{bmatrix} \begin{Bmatrix} u_n \\ u_m \end{Bmatrix} = \begin{Bmatrix} \bar{P}_n \\ P_m \end{Bmatrix} \quad \text{Eq. 7-12}$$

Bars over symbols are used to designate arrays that are replaced in the reduction process.

The addition of constraints to the structure requires that the forces of constraint be added to the equilibrium equations. Thus, writing the equilibrium and constraint equations together in partitioned form,

$$\begin{bmatrix} \bar{K}_{nn} & | & K_{nm} & | & G_m^T \\ \hline K_{nm}^T & | & K_{mm} & | & -I \\ \hline G_m & | & -I & | & \mathbf{0} \end{bmatrix} \begin{Bmatrix} u_n \\ u_m \\ q_m \end{Bmatrix} = \begin{Bmatrix} \bar{P}_n \\ P_m \\ \mathbf{0} \end{Bmatrix} , \quad \text{Eq. 7-13}$$

where $\{q_m\}$ is the vector of constraint forces on $\{u_m\}$. Straightforward elimination of u_m and q_m gives

$$[\bar{K}_{nn} + K_{nm}G_m + G_m^TK_{mm} + G_m^TK_{mm}G_m]\{u_n\} = \{\bar{P}_n\} + [G_m^T]\{P_m\} , \quad \text{Eq. 7-14}$$

or

$$[K_{nn}]\{u_n\} = \{P_n\} , \quad \text{Eq. 7-15}$$

where

$$K_{nn} = \bar{K}_{nn} + K_{nm}G_m + G_m^T K_{nm}^T + G_m^T K_{mm} G_m, \quad \text{Eq. 7-16}$$

and

$$P_n = \bar{P}_n + G_m^T P_m. \quad \text{Eq. 7-17}$$

The initial partition of K_{gg} and the operations indicated by [Eq. 7-9](#), [Eq. 7-16](#) and [Eq. 7-17](#) are performed by appropriate modules of the program. The multipoint constraint matrix, G_m , is used in structural matrix reduction ([Eq. 7-16](#)), load vector reduction, ([Eq. 7-17](#)) and data recovery ([Eq. 7-10](#)).

The multipoint constraint equations, R_{mg} , formed in the GP4 module, are partitioned by the MCE1 module (Multipoint Constraint Eliminator -- Phase 1) as follows:

$$[R_{mg}] = \left[\begin{array}{c|c} R_{mm} & R_{mn} \end{array} \right] \quad \text{Eq. 7-18}$$

MCE1 also solves the equation

$$[R_{mm}][G_{mn}] = -[R_{mn}] \quad \text{Eq. 7-19}$$

for the transformation matrix $[G_{mn}]$. An alternate path is available for module MCE1 in SubDMAP XMCE1, requested by PARAM,MCE1,NEW. It contains a partitioned solution for $R_{mm}^{-1} \cdot R_{mn}$, where the part of R_{mm} that is diagonal is partitioned from the coupled part. It was developed at a time when the partitioned solution was faster than the single-pass solution used in MCE1. The single pass solution was improved so that it is now faster than the partitioned solution. The partitioned solution is maintained to provide an alternate solution when the single-pass solution is implausible, or does not complete in a normal manner.

The MCE2 module (Multipoint Constraint Eliminator -- Phase 2) partitions the stiffness matrix

$$[K_{gg}] = \left[\begin{array}{c|c} \bar{K}_{nn} & K_{nm} \\ \hline K_{mn} & K_{mm} \end{array} \right] \quad \text{Eq. 7-20}$$

and performs matrix reduction

$$[K_{nn}] = [G_{mn}]^T [K_{mm} G_{mn} + K_{nn}] + [K_{mn}^T G_{mn} + \bar{K}_{nn}] \quad \text{Eq. 7-21}$$

The other structural matrices, $[K_{gg}^4]$, $[B_{gg}]$ and $[M_{gg}]$, are transformed by formulas that are identical in form to [Eq. 7-16](#), in subDMAPs SEMR2 and SEMRB.

The MCE2 module is also used in subDMAP SEMR2 to obtain $[M_{nn}]$.

The UPARTN module partitions $[K_{gg}]$ to $[K_{mg}]$ form which is equivalent to

$$[K_{mg}] = [K_{mn} \mid K_{mm}] \quad \text{Eq. 7-22}$$

and is used to obtain multipoint forces of constraint. See “[Data Recovery Operations in SubDMAP SEDISP](#)” on page 407.

7.4.4 Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB

The single point constraint operations apply constraints to the structural matrices using the single point constraints specified by the user on SPCi Bulk Data entries, the PS field on GRID entries, or constrained by the AUTOSPC feature described in “[Constraint and Mechanism Problem Identification in SubDMAP SEKR](#)” on page 383.

Single point constraints are applied to the set, u_s , in the form

$$\{u_s\} = \{Y_s\}, \quad \text{Eq. 7-23}$$

where $\{Y_s\}$ is a vector of enforced deformations, any or all of whose elements may be zero. The set, u_n , is partitioned into u_s and u_f (the free or unconstrained set)

$$\{u_n\} = \begin{Bmatrix} u_f \\ u_s \end{Bmatrix} \quad \text{Eq. 7-24}$$

The stiffness matrix, K_{nn} , is similarly partitioned

$$[K_{nn}] = \begin{bmatrix} K_{ff} & K_{fs} \\ K_{fs}^T & K_{ss} \end{bmatrix} \quad \text{Eq. 7-25}$$

The complete structural equations including the single point forces of constraint, q_s , may be written in partitioned matrix form as

$$\left[\begin{array}{c|c|c} K_{ff} & K_{fs} & \mathbf{0} \\ \hline K_{fs}^T & K_{ss} & -I \\ \hline \mathbf{0} & I & \mathbf{0} \end{array} \right] \begin{Bmatrix} u_f \\ u_s \\ q_s \end{Bmatrix} = \begin{Bmatrix} \bar{P}_f \\ P_s \\ Y_s \end{Bmatrix} \quad \text{Eq. 7-26}$$

Straightforward elimination gives

$$[K_{ff}]\{u_f\} = \{\bar{P}_f\} - [K_{fs}]\{Y_s\} = \{P_f\} \quad \text{Eq. 7-27}$$

The forces of constraint are recovered by means of the middle row of [Eq. 7-26](#), i.e.,

$$\{q_s\} = -\{P_s\} + [K_{fs}^T]\{u_f\} + [K_{ss}]\{u_s\} \quad \text{Eq. 7-28}$$

The distinct partitions K_{fs} and K_{ss} , are needed in subsequent calculations of SPC forces in static analysis. For the other structural matrices (K_{nn}^A , B_{nn} , and M_{nn}) the (sf) and (ss) partitions are saved, for dynamic SPC forces calculation.

The partially constrained stiffness matrix $[K_{nn}]$ is partitioned by UPARTN module as follows:

$$[K_{nn}] = \left[\begin{array}{c|c} K_{ff} & K_{fs} \\ \hline K_{sf} & K_{ss} \end{array} \right] \quad \text{Eq. 7-29}$$

Similar partitioning operations are performed on $[M_{nn}]$ in subDMAP SEMR2 to obtain $[M_{ff}]$, $[M_{sf}]$ and M_{ss} . In subDMAP SEMRB, the structural and viscous damping matrices

$$[K_{sf}^A], [K_{ss}^A], [B_{sf}] \text{ and } [B_{ss}]$$

are obtained from the MATREDU module. All (sf) and (ss) matrices are required for the recovery of single point constraint reaction forces described in “[Data Recovery Operations in SubDMAP SEDISP](#)” on page 407.

7.4.5 Constraint and Mechanism Problem Identification in SubDMAP SEKR

A singularity in a linear equation solution leads to a condition where a unique solution is not possible. For example, if $[K]\{u\} = \{F\}$, and $[K]$ is zero, then if $\{F\}$ is not zero, no value of $\{u\}$ will satisfy the equation. However, if $\{F\}$ is zero, any value of

$\{u\}$ will satisfy the equation. The equation can always be satisfied by prescribing $\{u\}$, rather than $\{F\}$. This is equivalent to constraining u with a single-point constraint.

There are two types of singularity:

1. The singularity can be identified by considering the stiffness terms of only one grid point. This is called a grid point singularity.
2. The singularity requires consideration of stiffness terms of more than one grid point. This is called a mechanism type of singularity.

Grid Point Singularity

Grid point singularities are automatically identified by the GPSP module. They may be automatically constrained, under user control. First, multipoint constraints are eliminated, leading to the matrix $[K_{nn}]$ (see “[Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB](#)” on page 379). Singularities are identified by grid point and component number of the failed direction. The ratio of stiffness at the degree-of-freedom to the largest stiffness at that grid point is listed. If automatic constraint is selected (see “[AUTOSPC](#)” on page 607 of the *MSC.Nastran Quick Reference Guide*), the set membership of the singular degrees-of-freedom is listed before the automatic constraint and after the automatic constraint. Both the mutually exclusive set and the highest combined set identical to the mutually exclusive set are listed for each identified singularity.

The output will include a Grid Point Singularity Table with the following format:

G R I D P O I N T S I N G U L A R I T Y T A B L E						
POINT ID	TYPE	FAILED DIRECTION	STIFFNESS RATIO	OLD USET	NEW USET	
3801	G	1	0.00E+00	L	S	*
3801	G	2	0.00E+00	L	S	*
3801	G	3	0.00E+00	L	S	*
3801	G	4	0.00E+00	L	S	*
3801	G	5	0.00E+00	L	S	*
3801	G	6	0.00E+00	L	S	*

In this table,

1. G (under the heading TYPE) stands for grid point (S for scalar point is also output).

2. FAILED DIRECTION is the degree-of-freedom component in the displacement coordinate system and closest to the weakest direction. See “[Grid Point and Coordinate System Definition](#)” on page 41 for definitions of component numbers.
3. The STIFFNESS RATIO is the ratio of the stiffness in the weakest direction to the stiffness in the strongest direction at the grid point (considering all possible directions rather than just the coordinate directions). Translational and rotational stiffnesses are considered separately. The default value for the failure criterion is 10^{-8} , which may be changed by the PARAM,EPZERO entry.
4. Note that, for the example shown, the failed degrees-of-freedom were originally in the u_l set and were placed in the u_s set. This may not always be the case, for reasons noted earlier, or because the user has not requested the AUTOSPC feature.
5. The user should carefully inspect the Grid Point Singularity Table because a singularity is very often the result of a modeling error, which should be corrected.

User Options

The identification of singularities is an automatic feature; however, the user may exercise the following options through PARAM Bulk Data entries described in “[Parameters](#)” on page 601 of the *MSC.Nastran Quick Reference Guide*.

Parameter Name	Description
AUTOSPC	Controls automatic constraint of a singularity.
EPPRT	Specifies the maximum value of a singularity to print.
EPZERO	Specifies the maximum value of a singularity.
PRGPST	Controls printout of singularity table.
SPCGEN	Writes SPC entries to the PUNCH file based on the singularity table.

The singular degrees-of-freedom are shifted to other sets by the following logic if PARAM,AUTOSPC,YES is specified:

o-set Condition	Set Before Constraint	Set After Constraint
If o-set exists	o	s
	c	b
If no o-set	c	s
	b	s

Other mutually exclusive sets not listed above will not be constrained. In addition, singular degrees-of-freedom on CYAX, CYJOIN, and SPCOFFi entries are identified but not constrained. The most usual cause of grid point singularity is degrees-of-freedom not constrained by elements because they are not of interest to the modeler. An example is out-of-plane rotations on a component made of plates lying in a plane. The automatic constraint provides the correct action for this type of singularity. Another cause is elements left out by oversight. Here, one typically observes that many or all degrees-of-freedom at a grid point are singular. The automatic constraint may mask a modeling error in this instance. For this reason, all singularity messages should be carefully inspected during the modeling checkout phase to ensure that modeling errors are not being masked.

When using the K2PP, M2PP, B2PP, or TFL Case Control command, PARAM,AUTOSPC,NO is recommended.

Mechanism-Type Singularities

A more complicated type of instability arises when two or more grid points participate in a singularity in a linear equation solution. This is known as a mechanism type of instability. Again, for the simple example shown in [Figure 7-2](#), the equations of equilibrium can not be satisfied at all, or may have nonunique solutions for special loading conditions. Note, however, that the structure of this example will pass the grid point singularity checks.

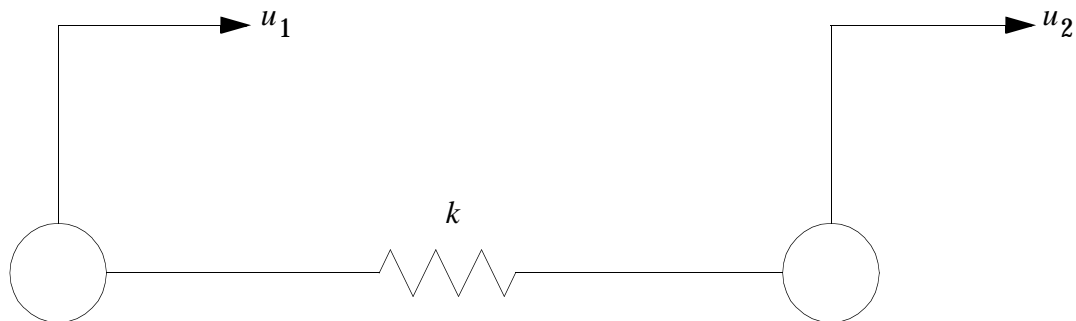


Figure 7-2 A Mechanism Type of Instability

The static equilibrium equation for the example in **Figure 7-2** is

$$\begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \begin{Bmatrix} P_1 \\ P_2 \end{Bmatrix} \quad \text{Eq. 7-30}$$

If $P_1 = -P_2$, many solutions are possible, where $(u_1 - u_2) = P_1/k$, but u_1 may have any value. If P_1 and P_2 are arbitrary, the equation cannot be satisfied.

This condition is detected during the matrix decomposition phase of the linear equation solution process by the DCMP module. The stiffness matrix is decomposed into a lower triangular matrix, $[L]$, and a diagonal matrix, $[D]$. As discussed in “**User Interface**” on page 67 of the *MSC.Nastran Numerical Methods User’s Guide* a diagonal term of $[D]$ (“ d_i ”) will approach zero for every mode of instability of the structure. (In this example there is one instability.) These instabilities are identified by dividing all d_i ’s into their corresponding stiffness term, then listing all ratios larger than a large number. See “**MAXRATIO**” on page 682 of the *MSC.Nastran Quick Reference Guide* for a description of several other parameters used to control this operation.

$$\text{Let } [K] = [L][D][L]^T$$

$$\text{Then } [D] = \begin{bmatrix} k & 0 \\ 0 & (k - k) \end{bmatrix}, [L] = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

If $(k - k) = 0.0$, replace d_i with 10^{-10}

$$\text{Ratio}_i = k_{ii} / d_i$$

Identify Grid Component for All Ratios > PARAM,MAXRATIO.

A corrective action for this example would be to constrain one of the grid points or to add springs connecting this spring to other parts of the structure (not shown). Since, in general, there may be many grid points active in a singularity, there is little reason for picking one over another for connection. The usual cause of this type of instability is elements left out by oversight. While this may be the user’s intention, it cannot be detected by the program. For these reasons, there is no automatic method to constrain mechanisms. The discussion of “**MAXRATIO**” on page 682 of the *MSC.Nastran Quick Reference Guide* describes the various actions and user options that are available in the several contexts where singularities may be detected. Diagnosis techniques and corrective actions are described in the *MSC.Nastran Numerical Methods User’s Guide*.

7.4.6 Static Condensation in SubDMAPs SEKR and SEMR2

Structural partitioning is used to refer structural matrices to the boundary points in substructuring operations and to perform a static condensation in dynamic analyses.

At user option the set of free coordinates, u_f , may be partitioned into two sets, u_o and u_a , such that the u_o set is eliminated first. Thus

$$\{u_f\} = \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} \quad \text{Eq. 7-31}$$

The equilibrium equations after the elimination of constraints (Eq. 7-27) may be written in partitioned form as

$$\begin{bmatrix} K_{aa} & | & K_{ao} \\ \hline K_{ao}^T & | & K_{oo} \end{bmatrix} \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \begin{Bmatrix} \bar{P}_a \\ P_o \end{Bmatrix} \quad \text{Eq. 7-32}$$

Rearrange the bottom half of Eq. 7-32:

$$[K_{oo}]\{u_o\} = \{P_o\} - [K_{ao}]^T\{u_a\}, \quad \text{Eq. 7-33}$$

and solve for $\{u_o\}$:

$$\{u_o\} = [K_{oo}]^{-1}\{P_o\} - [K_{oo}]^{-1}[K_{ao}]^T\{u_a\} \quad \text{Eq. 7-34}$$

Substitute for u_o into the top half of Eq. 7-32:

$$[\bar{K}_{aa} - K_{ao}K_{oo}^{-1}K_{ao}^T]\{u_a\} = \{\bar{P}_a\} - [K_{ao}][K_{oo}]^{-1}\{P_o\} \quad \text{Eq. 7-35}$$

It is convenient to define the matrix

$$[G_o] = -[K_{oo}]^{-1}[K_{ao}]^T, \quad \text{Eq. 7-36}$$

so the Eq. 7-35 becomes

$$[\bar{K}_{aa} + K_{ao}G_o]\{u_a\} = \{\bar{P}_a\} + [G_o]^T\{P_o\}, \quad \text{Eq. 7-37}$$

where advantage is taken of the symmetry of $[K_{oo}]$.

Following the practice of condensation established in preceding subsections,

$$[K_{aa}]\{u_a\} = \{P_a\}, \quad \text{Eq. 7-38}$$

where

$$[K_{aa}] = [\bar{K}_{aa}] + [K_{ao}][G_o], \quad \text{Eq. 7-39}$$

$$\{P_a\} = \{\bar{P}_a\} + [G_o]^T \{P_o\} \quad \text{Eq. 7-40}$$

The $[G_o]$ matrix defined in **Eq. 7-36** is obtained practically from the solution of

$$[K_{oo}][G_o] = -[K_{ao}]^T, \quad \text{Eq. 7-41}$$

where

$$[K_{ao}]^T$$

is treated as a set of load vectors. Each such vector produces a column of $[G_o]$.

Once $\{u_a\}$ is obtained the set of omitted coordinates, $\{u_o\}$, is obtained as follows. Define the set

$$\left\{ u_o^o \right\}$$

as the solution of

$$[K_{oo}]\left\{ u_o^o \right\} = \{P_o\} \quad \text{Eq. 7-42}$$

Then, using **Eq. 7-36** and **Eq. 7-42** in **Eq. 7-33**,

$$\{u_o\} = \left\{ u_o^o \right\} + [G_o]\{u_a\} \quad \text{Eq. 7-43}$$

In subDMAP SEKR, the constrained stiffness matrix, $[K_{ff}]$, is partitioned by UPARTN as follows:

$$[K_{ff}] = \begin{bmatrix} \bar{K}_{aa} & | & K_{ao} \\ \hline K_{oa} & | & K_{oo} \end{bmatrix} \quad \text{Eq. 7-44}$$

DCMP and FBS are used to solve the equation

$$[K_{oo}][G_{oa}^t] = -[K_{oa}] \quad \text{Eq. 7-45}$$

for the static transformation matrix, $[G_{oa}^t]$, and MPYAD performs the matrix reduction

$$[K_{aa}] = [\bar{K}_{aa}] + [K_{ao}][G_{oa}^t] \quad \text{Eq. 7-46}$$

If PARAM,ALTRED,YES is specified then the $[K_{ff}]$ matrix is partitioned and reassembled to form $[K_{ff}^x]$ which has the a-set degrees-of-freedom sequenced last. $[K_{ff}^x]$ is partially decomposed

$$[K_{ff}^x] \rightarrow [L_{fo}], [K_{aa}] \quad \text{Eq. 7-47}$$

where $[L_{fo}]$ is a trapezoidal matrix and $[K_{aa}]$ is the boundary stiffness matrix.

The $[L_{fo}]$ matrix is partitioned

$$[L_{fo}] = \begin{bmatrix} L_{oo} \\ \hline L_{ao} \end{bmatrix} \quad \text{Eq. 7-48}$$

While $[L_{ao}]^T$ is of the same dimension as the $[G_{oa}^t]$ matrix, it is on the average half as dense, thereby reducing the storage requirements.

In subDMAP SEMR2, UPARTN partitions the constrained mass matrix $[M_{ff}]$ as follows:

$$[M_{ff}] = \begin{bmatrix} \bar{M}_{aa} & | & M_{ao} \\ \hline M_{oa} & | & M_{oo} \end{bmatrix} \quad \text{Eq. 7-49}$$

and successive executions of MPYAD and SMPYAD modules perform the matrix reduction

$$[M_{aa}] = [G_{oa}^t]^T [M_{oo} \ G_{oa}^t + M_{oa}] + [M_{oa}^T \ G_{oa}^t + \bar{M}_{aa}] \quad \text{Eq. 7-50}$$

In subDMAP SEMRB, the structural and viscous damping matrices $[B_{ff}]$ and $[K_{ff}^4]$ are condensed by the MATREDU module.

The partitioned solution for statics is an exact solution. It merely changes the order of operations of the unpartitioned solution. The static reduction for dynamics is exact for o-set degrees-of-freedom that are massless, but represents an approximate relumping

of o-set masses to the a-set. The user must select enough a-set degrees-of-freedom to adequately define the eigenvectors of the system. The dynamic reduction technique discussed in “**Dynamic Reduction and Component Mode Synthesis in SubDMAP SEMR3**” on page 398 provides a way of improving the approximation by computing the normal modes for the o-set degrees-of-freedom.

7.4.7 Static Solutions in SubDMAP SEKRSS

There are two major branches through this subDMAP, standard static analysis and static analysis with inertia relief effects.

Inertia Relief Not Selected

When inertia relief is not selected and there are no SUPORTi entries are present the a-set equations are solved directly

$$K_{aa} \cdot u_a = P_a \quad \text{Eq. 7-51}$$

K_{aa} is decomposed into its factor matrices L_{aa} and D_a . These factors are passed on to SubDMAP STATRS for solution of u_a .

When inertia relief is not selected but there are SUPORTi entries are present the a-set equations are partitioned into the r-set, defined on the SUPORTi entries, and the l-set, what is “left over” after the r-set is removed.

$$\begin{bmatrix} K_{ll} & K_{lr} \\ K_{rl} & K_{rr} \end{bmatrix} \begin{bmatrix} u_l \\ u_r \end{bmatrix} \geq \begin{bmatrix} P_l \\ P_r \end{bmatrix} \quad \text{Eq. 7-52}$$

In this context, the r-set is treated as an alternate form of single-point constraints. The partitions of K involving r are discarded. u_r is set to zero. u_l is found from the first row of this equation. Loads in P_r go directly to ground, and are output with the SPC forces, along with the product $-K_{rl} \cdot u_l$.

Inertia Relief Selected

The basic assumption of inertia relief is that the structure is in a state of steady acceleration, and that all transients due to dynamic effects are negligible. The solution is obtained in a moving reference frame from which all displacements are measured. For inertia relief analysis inertial loads are computed automatically that exactly balance the applied loads. A support system is applied to make the system numerically stable. The inertial loads assume rigid body accelerations when computing the loads.

The derivations given here assume that the system mass matrix M_{gg} has been reduced to M_{aa} and is available, as described elsewhere. In the actual DMAP used in MSC.Nastran 2001 equivalent methods that never generate M_{aa} are used instead, in the interests of efficiency. M_{aa} can be a larger size data block, and costly to compute. It can be shown that the results provided by the two derivations are identical. For example, the rigid body mass matrix M_{rr} , defined below as

$$M_{rr} = D_{ar} \cdot M_{aa} \cdot D_{ar} \quad \text{Eq. 7-53}$$

can just as easily be formed from the g-set equations

$$M_{rr} = D_{gr} \cdot M_{gg} \cdot D_{gr} \quad \text{Eq. 7-54}$$

Similar comments can be made about computing the inertial loads that balance the applied loads. The derivatives are presented here in the form of a-set equations that are more readily understood, if less efficient in practice, than the g-set equations. Readers interested in the equations of the more efficient method may find them in older versions of this manual, TANs 4002 and 4854, or by reading DMAP compilations. Current research indicates that this capability may shift to a-set operations in future versions to obtain more automation.

There are at present two forms of inertia relief: automatic selection of a support system (“auto-support”); and manual selection of the support system, using SUPORTi entries. Both react to PARAM,GRDPNT, the parameter used to locate a reference point for grid point weight summaries. The more automatic method is described first.

Auto-Support (PARAM,INREL,-2)

Vectors of rigid body motion D_{gr} for six unit motions of a reference point are computed from grid point geometry in module VECPLOT. The D_{ar} component is partitioned from this matrix. D_{ar} is used to compute the rigid body mass M_{rr} about the reference point.

$$MD_{ar} = M_{aa} \cdot D_{ar} \quad \text{Eq. 7-55}$$

$$M_{rr} = D_{ar} \cdot MD_{ar} \quad \text{Eq. 7-56}$$

M_{rr} is printed out with the matrix name of QRR.

The acceleration of the reference point ar due to the applied loads is computed,

$$M_{rr} \cdot ar \geq D_{ar} \cdot P_a \quad \text{Eq. 7-57}$$

The accelerations ar are printed with the matrix name of URA.

The resulting inertial loads P_{ai} and total loads P_a are

$$P_{ai} = M_{aa} \cdot ar \quad \text{Eq. 7-58}$$

$$P_{a1} = P_a - P_{ai} \quad \text{Eq. 7-59}$$

A constraint equation is written that states that the average weighted motion of the system is zero,

$$MD_{ar} \cdot u_{ar} = 0 \quad \text{Eq. 7-60}$$

This equation is appended to the a-set to define the 1-set equations in their augmented form,

$$K_{ll} \cdot u_l = P_l \quad \text{Eq. 7-61}$$

$$\begin{bmatrix} K_{aa} & MD_{ar} \\ MD_{ar} & 0_{rr} \end{bmatrix} \begin{bmatrix} u_a \\ -q_r \end{bmatrix} = \begin{bmatrix} P_{a1} \\ 0_r \end{bmatrix} \quad \text{Eq. 7-62}$$

q_r are the unknown forces required on the system to produce equilibrium. They should be computational zeros, and are output with the SPC forces. The equation involving K_{ll} is solved directly, with u_a and q_r partitioned from u_l .

Manual Suport (PARAM, INREL, -1 and SUPORTi)

The user prescribes a set of DOFs that constrain the structure in a statically-determinate manner on SUPORTi entries. These entries define the r-set. There must be six or less DOFs listed, and they must constrain all rigid body modes that are not constrained by SPCs or other modeling devices. The rigid body modes are computed from the statics equation, where there are no loads applied, and u_{rr} is the identity matrix. Given

$$\begin{bmatrix} K_{ll} & K_{lr} \\ K_{le} & K_{rr} \end{bmatrix} \begin{bmatrix} D_{lr} \\ I_{rr} \end{bmatrix} = \begin{bmatrix} 0_{lr} \\ 0_{rr} \end{bmatrix} \quad \text{Eq. 7-63}$$

D_{lr} is found by solving the first row of this equation, then merged with I_{rr} to form D_{ar} ,

$$K_{ll} \cdot D_{lr} \geq -K_{lr} \cdot I_{rr} \quad \text{Eq. 7-64}$$

The rigid body mass, reference point accelerations, and inertial loads are formed with the same equations used for auto-support. The loads are partitioned into the l- and r-sets, and the partitioned form of K_{ll} is used for solution for displacements,

$$K_{ll} \cdot u_l = P_{ll} \quad \text{Eq. 7-65}$$

u_a is merged from u_l and a null partitioned for u_r .

A Comparison of Auto- and Manual-Support Solution

If one analyzes the same model with both methods, you should find that stresses, constraint forces, and other internal quantities are identical except for truncation errors, but the displacements may differ appreciably. This is because they use a different reference frame. For the many supports, the reference frame is set up so that the motion at the r-set DOFs are identically zero. One can think of a reference frame drawn through these points, and all displacements measured relative to it. This reference frame is moving through space at a constant acceleration rate, so the displacements are relative, rather than inertial displacements.

For the auto-support option, one can visualize the reference frame as a set of axes connected to every point with mass with a very complicated set of levers that allows the structure to “breathe”, as loads are applied to it, but keep the weighted average motion to a zero value. If the model happens to have a grid point at the reference point, you will observe very small motion there, computational zeros but not the binary zeros of the manual support method. This is because this motion is computed for the auto-support method, but assumed to be zero for the manual method.

The choice of an r-set as defined by SUPPORTi DOFs requires meeting both a geometric requirement and a stiffness requirement. The geometric requirement is that the set should form a statically determinate tie to ground. This can be difficult to do with complicated geometries. The stiffness requirement is that the DOFs should be stiff enough to provide a stable solution. A method that has been used by some modelers is to pick points stiff enough that they could be used to lift the structure without damaging it. Sometimes such points do not exist. The difficulties in picking a stable, reliable support system was one of the main motivations for development of the more modern auto-support systems.

PARAM, GRDPNT, GID is optional for both methods when superelements are not present, but required by the manual method when superelements are present. The auto-method does not support superelement analysis. The default for GRDPNT is GID = -1, which is a flag to use the basic origin as the reference point when computing rigid body modes. Use of a GID near the c.g. of the structure produces more

esthetically pleasing results. The rigid body accelerations produced by the solution are then measured near the center of the structure, rather than at a point that may be outside of the structure.

7.4.8 Static and Dynamic Load Generation

Applied Static Loads in SubDMAPs SELG, SELR, and SELRRS

The static load vectors $\{P_g\}$ are generated by SSG1 (Static Solution Generator -- Phase 1) in subDMAP SELG from the user input of applied loads, temperature fields, and enforced deformations.

The direct matrix input requested by the P2G Case Control command define the load matrix $[P_g^2]$ which is assembled by the MTRXIN module and is added to the output from SSG1.

If PARAM,INREL,-1 is specified, then

1. The rigid body transformation matrix $[D_{gr}]$ is generated with the VECPLOT module. Its columns contain the motion of all degrees-of-freedom due to unit motion of the corresponding reference point degree-of-freedom. (See PARAM,INREL in “[Use of Parameters](#)” on page 35 for a discussion of the reference point (“GIC”).)
2. Loads due to unit rigid body accelerations are computed,

$$\{P_j^i\} = -[M_{jj}][D_{gr}]$$

3. The loads above are appended to the applied loads $\{P_j\}$,

$$\{P_j\} \leftarrow \{P_j \mid P_j^i\}$$

The SELA module assembles $\{P_j\}$ into $\{P_g\}$.

The constraints are applied to the static load vectors by SSG2 (Static Solution Generator -- Phase 2) in subDMAP SELR as follows:

$$\{P_g\} = \begin{Bmatrix} \bar{P}_n \\ P_m \end{Bmatrix} \quad \text{Eq. 7-66}$$

$$\{P_n\} = \{\bar{P}_n\} + [G_{mn}]^T \{P_m\} \quad \text{Eq. 7-67}$$

$$\{P_n\} = \begin{Bmatrix} \bar{P}_f \\ P_s \end{Bmatrix} \quad \text{Eq. 7-68}$$

$$\{P_f\} = \{\bar{P}_f\} - [K_{fs}]\{Y_s\} \quad \text{Eq. 7-69}$$

$$\{P_f\} = \begin{Bmatrix} \bar{P}_a \\ P_o \end{Bmatrix} \quad \text{Eq. 7-70}$$

$$\{P_a\} = \{\bar{P}_a\} + [G_{oa}]^T \{P_o\} \quad \text{Eq. 7-71}$$

If PARAM,ALTRED,YES is specified, the diagonal extracted from $[L_{oo}] \rightarrow [D_{oo}]$ and the reduced load matrix is computed from

$$\{P_a\} = \{\bar{P}_a\} - [L_{ao}][D_{oo}]\{u_o^x\} \quad \text{Eq. 7-72}$$

where $\{u_o^x\}$ is obtained on a forward pass from the equation

$$[L_{oo} \ D_{oo}]\{u_o^x\} = \{P_o\} \quad \text{Eq. 7-73}$$

for PARAM,ALTRED,NO, the SSG3 module (Static Solution Generation -- Phase 3) solves the equation

$$[K_{oo}]\{u_o^o\} = \{P_o\} \quad \text{Eq. 7-74}$$

for $\{u_o^o\}$, the displacements of the omitted coordinates.

SSG3 also calculates the residual vector, δP_o , and the residual vector error ratio, ϵ_o , for the omitted coordinates

$$\{\delta P_o\} = [K_{oo}]\{u_o^o\} - \{P_o\} \quad \text{Eq. 7-75}$$

$$\epsilon_o = \frac{\{u_o^o\}^T \{\delta P_o\}}{\{P_o\}^T \{u_o^o\}} \quad \text{Eq. 7-76}$$

Except for round-off error, the error ratio ϵ_o should be zero. Large values of these error ratios usually indicate singularities in the stiffness matrix. The residual load vector, RUOV, may be output by use of PARAM,IRES,1.

The quantity

$$1/2\{P_o\}^T\{u_o\} \quad \text{Eq. 7-77}$$

is calculated by the SSG3 module and printed under the heading “External Work.” This component of strain energy includes effects of thermal loads, element deformations, and enforced displacements that may be subtracted later in the solution process. For example, an enforced displacement that causes zero strain will result in external work.

If a r-set is present, subDMAP SELRRS calculates determinate forces of reaction

$$[q_r] = -[D_{ar}]^T\{P_a\} \quad \text{Eq. 7-78}$$

where:

$$[D_{ar}] = \begin{bmatrix} D_{er} \\ I_{rr} \end{bmatrix}$$

Applied Dynamic Loads

Dynamic loads are produced in many subDMAPs. They are generally of the form:

$$P(x) = \sum_i PO_i \cdot F_i(x) \quad \text{Eq. 7-79}$$

The PO_i vectors are spatial functions that describe what DOFs are loaded and their scale factors. The F_i functions are temporal functions that define how the loads change with time in transient analysis $x = t$, or with excitation frequency in frequency response analysis $x = f$. All spatial loads except those on extra points are calculated in the static analysis load generator module, SSG1, to obtain the P_g vector described in the prior paragraphs.

Enforced motions are produced similarly.

$$Y(x) = \sum_i YO_i \cdot F_i(x) \quad \text{Eq. 7-80}$$

The YO_i vectors are again the spatial distribution, and the $F_i(x)$ temporal functions. The YO_i vectors are produced by the GP4 module to produce the Y_g vector. It is the same vector described for static analysis in [Eq. 7-69](#), except that it is merged to g-set size with null terms from Y_s .

The static load and enforced motion vectors are converted into dynamic loading tables in DPD module, then passed on to the dynamic load generators, FRLG and TRLG. These modules produce the applied loads and enforced motions at the p-set level, $P_p(x)$ and $Y_p(x)$, as a function of excitation frequency or time. For transient analysis velocity and acceleration vectors are also provided, $\dot{Y}_p(x)$ and $\ddot{Y}_p(x)$.

A reduction matrix R_{px} is made in the FDRMGEN module, where the subscript “x” implies the d-set, for direct analysis, and h, for modal analysis. It combines the operations shown in Eq. 7-66 through Eq. 7-71 previously. In subDMAP CFORCE2 Eq. 7-69 is modified to add velocity and acceleration effects,

$$P_f = P_f - K_{fs} \cdot Y_s - B_{fs} \cdot \dot{Y}_s - M_{fs} \cdot \ddot{Y}_s \quad \text{Eq. 7-81}$$

In frequency response the velocity and acceleration effects can be computed from Y_s by multiplying it by $i \cdot \omega$ once or twice, where ω is the excitation frequency.

P_f is reduced to P_d or P_h by the same techniques used in static analysis.

For modal analysis with enforced motion it is essential that RESVEC be used. It causes static solutions to be formed from $K_{fs} \cdot Y_s$ -type terms. When the structure is tied to ground in a statically-determinate manner, these solutions are linear combinations of the rigid body modes that would exist if the ties to ground were removed. If the ties to ground are redundant, a condition named multi-base input, the resulting shapes will have significant stresses associated with them. They represent the effect of low-frequency excitation on the structure when RESVEC is not present, for the statically determinate case the element stresses will be correct but the displacement-type data will be missing the effect of the static shapes. For the multi-base input case, both stresses and displacement-type output will be incorrect.

7.4.9 Dynamic Reduction and Component Mode Synthesis in SubDMAP SEMR3

These capabilities use several subsets of the a-set not used by other solution techniques. The v-set matrix is formed in subDMAP SEKR

$$K_{vv} = \begin{bmatrix} \overline{K_{oo}} & \overline{K_{or}} & \overline{K_{oc}} \\ \overline{K_{ro}} & \overline{K_{rr}} & \overline{K_{rc}} \\ \overline{K_{co}} & \overline{K_{cr}} & \overline{K_{cc}} \end{bmatrix} \quad \text{Eq. 7-82}$$

The v-set mass matrix is formed similarly in subDMAP SEMR3. The v-set degrees-of-freedom are free to vibrate during dynamic reduction and component mode computation. They consist of the omitted terms (o-set), the free boundary points used for rigid-body mode designation (r-set), and the other free boundary points (c-set). The $[M_v]$ matrix is formed in a similar manner.

If generalized dynamic reduction is requested by the DYNRED Case Control command, the shifted stiffness matrix is formed,

$$[A_{vv}] = [K_{vv}] + \lambda_s [M_{vv}] \quad \text{Eq. 7-83}$$

where λ_s is determined from data on the specified DYNRED Bulk Data entry. The approximate eigenvectors are found from $[A_{vv}]$ and $[M_{vv}]$ using inverse iteration based on solving the equation

$$[A_{vv}][\phi_v]_{i+1} = [M_{vv}][\phi_v]_i \quad \text{Eq. 7-84}$$

for $[\phi_v]_{i+1}$, starting with random vectors for $[\phi_v]_0$. The $[\phi_v]_i$ are stored, then made mass orthogonal with respect to each other to form the matrix $[\phi_{vy}]$, where N_y the number of generalized functions used, may be determined automatically. N_y must be equal to or less than N_g , the upper limit for the number of generalized degrees-of-freedom specified by the user.

The generalized mass and stiffness matrices are computed,

$$[M_{yy}] = [\phi_{vy}]^T [M_{vv}] [\phi_{vy}] \quad \text{Eq. 7-85}$$

$$[K_{yy}] = [\phi_{vy}]^T [K_{vv}] [\phi_{vy}] \quad \text{Eq. 7-86}$$

If the METHOD command is specified for superelements, then the component modes are computed. The auto-omit operations described in “[Real Eigenvalue Analysis in SubDMAPs SEMR3 and MODERS](#)” on page 423 are carried out resulting in the reduced eigensolution equation

$$[K_{yy} - \lambda M_{yy}][\phi_{yz}] = 0 \quad \text{Eq. 7-87}$$

where N_z is the number of eigenvectors requested by the EIGR or EIGRL Bulk Data entries. The back-transformations for the auto-omit-spc are performed, resulting in the expanded eigenvector matrix $[\phi_{yz}]$. This matrix is used to purify the approximate eigenvectors found in dynamic reduction,

$$[\phi_{vz}] = [\phi_{vy}][\phi_{yz}] \quad \text{Eq. 7-88}$$

The uncoupled eigensolution components are assembled from $[\phi_{vz}]$

$$[\phi_{vz}] = \begin{Bmatrix} \phi_{oz} \\ \phi_{rz} \\ \phi_{cz} \end{Bmatrix} \quad \text{Eq. 7-89}$$

$$[\phi_{az}] = \begin{Bmatrix} \phi_{rz} \\ \phi_{cz} \\ \phi_{bz} \end{Bmatrix} \quad \text{Eq. 7-90}$$

where $[\phi_{bz}] = [0]$

$[\phi_{oz}]$ and $[\phi_{az}]$ are stored for use with the uncoupled solution obtained using PARAM,FIXEDB,-1, as described in “[Use of Parameters](#)” on page 35 and “[Data Recovery Operations in SubDMAP SEDISP](#)” on page 407.

The residual vector capability is requested with the RESVEC Case Control command. Details on this capability are given in “[Matrix Assembly Operations in SubDMAP SEMG](#)” on page 377, where real eigensolutions are discussed. The RESVEC SubDMAP returns auxiliary functions u_{o2} . They are appended to the eigenvectors, and are given all of the operations given to eigenvectors,

$$H_{vz} = \begin{bmatrix} \phi_{oz} & u_{o2} \\ \phi_{rz} & \mathbf{0}_{r2} \\ \phi_{cz} & \mathbf{0}_{c2} \end{bmatrix} \quad \text{Eq. 7-91}$$

The inertia relief mode shapes are calculated by statically applying the rigid-body inertial loads to the structure.

$$\begin{bmatrix} K_{oo} & K_{oa} \\ K_{ao} & K_{aa} \end{bmatrix} \begin{bmatrix} H_o^i \\ H_a^i \end{bmatrix} = \begin{bmatrix} M_{oo} & M_{oa} \\ M_{ao} & M_{aa} \end{bmatrix} \begin{bmatrix} V_o \\ V_a \end{bmatrix} \quad \text{Eq. 7-92}$$

Setting $[H_a^i]$ to zero and solving for $[H_o^i]$ results in,

$$[H_o^i] = [K_{oo}]^{-1} [[M_{oo}][V_o] + [M_{oa}][V_a]] \quad \text{Eq. 7-93}$$

The $[G_{oq}]$ matrix is formed,

$$[G_{oq}] \leftarrow [G_{oz}^1 \mid H_o^i \mid 0] \quad \text{Eq. 7-94}$$

The $[G_{oq}]$ matrix is truncated or padded with null columns to provide N_q columns, where N_q is the q-set size. $[G_{oq}]$ is expanded to a-size,

$$[G_{oq} \mid 0_{ot}] \rightarrow [G_{oa}^q] \quad \text{Eq. 7-95}$$

The total transformation matrix is formed by adding dynamic and static components and is reassembled whenever needed,

$$[G_{oa}] = [G_{oa}^t] + [G_{oa}^q] \quad \text{Eq. 7-96}$$

The generalized stiffness coefficients are formed from the dynamic transformation,

$$[K_{qq}] = [G_{oq}]^T [K_{oo}] [G_{oq}] \quad \text{Eq. 7-97}$$

The physical and generalized degrees-of-freedom are not stiffness coupled. $[K_{qq}]$ is merged to N_a size,

$$[K_{aa}^l] = \begin{bmatrix} K_{qq} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \text{Eq. 7-98}$$

where superscript l refers to matrices formed from dynamic reduction. $[K_{qq}]$ is stored in the database. When the total stiffness boundary matrix is needed, it is formed from its partitions,

$$[K_{aa}^m] = \begin{bmatrix} K_{qq} & \mathbf{0} \\ \mathbf{0} & \bar{K}_{tt} \end{bmatrix} \quad \text{Eq. 7-99}$$

or

$$[K_{aa}] = [K_{aa}^m] + [K_{aa}^l] \quad \text{Eq. 7-100}$$

The physical and generalized degrees-of-freedom are mass-coupled. The terms are formed from the equations,

$$\begin{aligned}
[M_{qq}] &= [G_{oq}]^T [M_{oo}] [G_{oq}] \\
[M_{qt}] &= [G_{oq}]^T [M_{ot} + M_{oo} \ G_{oa}^t] \\
[M_{aa}^l] &= \begin{bmatrix} M_{qq} & M_{qt} \\ M_{tq} & \mathbf{0} \end{bmatrix}
\end{aligned}
\tag{Eq. 7-101}$$

Virtual mass effects, if any, are added to $[M_{aa}^l]$ to form $[M_{aa}^l]$.

The total mass matrix is formed by:

$$[M_{aa}^m] = [M_{aa}] + [M_{aa}^l] \tag{Eq. 7-102}$$

For the rigid formats $[M_{aa}]$ is formed directly from its partitions,

$$[M_{aa}] = \begin{bmatrix} M_{qq} & M_{qt} \\ M_{tq} & M_{tt} \end{bmatrix} \tag{Eq. 7-103}$$

The a-set matrices do not contain static effects in the t-set components, and dynamic effects in the q-set components. Virtual mass effects, if any, are added to $[M_{aa}]$ to form $[M_{aa}^m]$.

The damping matrices $[B_{gg}]$ and $[K_{gg}^4]$ are reduced to the a-set by the MATREDU module which performs the following operations:

1. Eliminate multipoint constraints

$$[B_{gg}] \Rightarrow \begin{bmatrix} \bar{B}_{nn} & B_{nm} \\ B_{mn} & B_{mm} \end{bmatrix}$$

$$[B_{nn}] = [G_{mn}]^T [B_{mn} G_{mn} + B_{mn}] + [B_{mn}]^T [G_{mn}] + [B_{nn}]$$

2. Eliminate single point constraints

$$[B_{nn}] \Rightarrow \begin{bmatrix} B_{ff} & B_{fs} \\ B_{sf} & B_{ss} \end{bmatrix}$$

3. Partition omitted degrees-of-freedom

$$[B_{ff}] \Rightarrow \begin{bmatrix} \bar{B}_{aa} & B_{ao} \\ B_{oa} & B_{oo} \end{bmatrix}$$

4. Perform static condensation

$$[B_{aa}] = [G_{oa}]^T [B_{oo} G_{oa} + B_{oa}] + [\bar{B}_{oa}]^T [G_{oa}] + [\bar{B}_{aa}]$$

7.4.10 Formulation of Dynamic Equations in SubDMAP GMA

The tables prepared by the geometry processors are modified by DPD (Dynamic Pool Distributor) to include the extra points (e-points) introduced for dynamic analysis.

$[K_{aa}^m]$, $[M_{aa}^m]$, $[B_{aa}]$, and $[K_{aa}^4]$ are expanded to the d-set to form $[K_{dd}^1]$, $[M_{dd}^1]$, $[B_{dd}^1]$, and $[K_{dd}^4]$. The direct input matrices $[K_{pp}^2]$, $[M_{pp}^2]$, and $[B_{pp}^2]$ requested by the K2PP, M2PP, and B2PP Case Control commands are reduced to the d-set to form $[K_{dd}^2]$, $[M_{dd}^2]$, and $[B_{dd}^2]$, respectively.

The general dynamic equation used in the direct methods is

$$[[M_{dd}]p^2 + [B_{dd}]p + [K_{dd}]]\{U_d\} = \{P_d\} \quad \text{Eq. 7-104}$$

p is the derivative operator

$$\{U_d\} = \begin{Bmatrix} U_a \\ U_e \end{Bmatrix} \quad \text{Eq. 7-105}$$

where:

$\{U_d\}$ = the extra points

For direct frequency response and complex eigenvalue analysis the stiffness, damping, and mass matrices are generated as follows:

$$[K_{dd}] = (1 + ig)[K_{dd}^1] + [K_{dd}^2] + i[K_{dd}^4] \quad \text{Eq. 7-106}$$

$$[B_{dd}] = [B_{dd}^1] + [B_{dd}^2] \quad \text{Eq. 7-107}$$

and

$$[M_{dd}] = [M_{dd}^1] + [M_{dd}^2] \quad \text{Eq. 7-108}$$

For direct transient response the stiffness, damping and mass matrices are generated as follows:

$$[K_{dd}] = [K_{dd}^1] + [K_{dd}^2] \quad \text{Eq. 7-109}$$

$$[B_{dd}] = [B_{dd}^1] + [B_{dd}^{2x}] + \frac{g}{\omega_3} [K_{dd}^1] + \frac{1}{\omega_4} [K_{dd}^4] \quad \text{Eq. 7-110}$$

and

$$[M_{dd}] = [M_{dd}^1] + [M_{dd}^2] \quad \text{Eq. 7-111}$$

where:

- $[K_{dd}^1]$ is the reduced structural stiffness matrix plus the reduced direct input K2GG (symmetric).
- $[K_{dd}^2]$ is the reduced direct input matrix K2PP plus the reduced transfer function input (symmetric or unsymmetric).
- $[K_{dd}^4]$ is the reduced structural damping matrix obtained by multiplying the stiffness matrix $[K_e]$ of an individual structural element by an element damping factor g_e and combining the results for all structural elements (symmetric).
- $[B_{dd}^1]$ is the reduced viscous damping matrix plus the reduced direct input B2GG (symmetric).
- $[B_{dd}^2]$ is the reduced direct input matrix B2PP plus the reduced transfer function input (symmetric or unsymmetric).
- $[M_{dd}^1]$ is the reduced mass matrix plus the reduced direct input M2GG (symmetric).
- $[M_{dd}^2]$ is the reduced direct input matrix M2PP plus the reduced transfer function input (symmetric or unsymmetric).
- g is the structural damping coefficient on the PARAM Bulk Data entry G. The frequencies ω_3 and ω_4 are specified on the PARAM Bulk Data entries as W3 and W4, respectively.

For direct solutions $[K_{dd}]$, $[M_{dd}]$, and $[B_{dd}]$ are examined to identify rows and columns which are null in all three matrices. If the parameters ASING = 0 (the default) the singularities caused by the null rows and columns are removed as follows:

1. For direct frequency and direct transient response, the stiffness matrix $[K_{dd}^x]$ is formed from $[K_{dd}]$ by placing unity on the diagonal for each null row and column.
2. For direct complex eigenvalue analysis null rows and columns are discarded from $[K_{dd}]$, $[M_{dd}]$, and $[B_{dd}]$ to form $[K_{xx}^d]$, $[M_{xx}^d]$, and $[B_{xx}^d]$. This is performed in subDMAP DCEIGRS. See “**Complex Eigenvalue Analysis in SubDMAP CEIGRS**” on page 440.

If parameter ASING = -1 and null rows and/or columns exist, a fatal error will result.

In modal frequency response, and complex eigenvalue analysis, structural damping and viscous damping are included in the dynamic matrices as follows:

$$[K_{dd}^2] = [K_{dd}^{2x}] + ig[K_{dd}^1] + i[K_{dd}^4] \quad \text{Eq. 7-112}$$

and

$$[B_{dd}^2] = [B_{dd}^1] + [B_{dd}^{2x}] \quad \text{Eq. 7-113}$$

In modal transient response, the structural damping and viscous damping are included as follows:

$$[K_{dd}^2] = [K_{dd}^{2x}] \quad \text{Eq. 7-114}$$

and

$$[B_{dd}^2] = [B_{dd}^1] + [B_{dd}^{2x}] + \frac{g}{\omega_3}[K_{dd}^1] + \frac{1.0}{\omega_4}[K_{dd}^4] \quad \text{Eq. 7-115}$$

The general dynamic equation used in the modal method is:

$$[[M_{hh}]p^2 + [B_{hh}]p + [K_{hh}]]\{U_h\} = \{P_h\} \quad \text{Eq. 7-116}$$

where p is the derivative operator

$$\{U_h\} = \begin{Bmatrix} \zeta_i \\ U_e \end{Bmatrix} \quad \text{Eq. 7-117}$$

ζ_i are the modal coordinates.

The transformation from $\{U_h\}$ to $\{U_d\}$ is

$$\{U_d\} = [\phi_{dh}]\{U_h\} \quad \text{Eq. 7-118}$$

where:

$$[\phi_{dh}] = \begin{bmatrix} \phi_{ai} & \mathbf{0} \\ \mathbf{0} & I_{ee} \end{bmatrix}$$

The dynamic matrices expressed in modal coordinates are assembled in GKAM (General K Assembler-Modal) as follows:

$$[K_{hh}] = [k] + [\phi_{dh}]^T [K_{dd}^2] [\phi_{dh}] \quad \text{Eq. 7-119}$$

$$[M_{hh}] = [m] + [\phi_{dh}]^T [M_{dd}^2] [\phi_{dh}] \quad \text{Eq. 7-120}$$

and

$$[B_{hh}] = [b] + [\phi_{dh}]^T [B_{dd}^2] [\phi_{dh}] \quad \text{Eq. 7-121}$$

where, if PARAM,KDAMP = 1 (default)

$$m_i = \text{modal mass}$$

$$b_i = 2\pi f_i g(f_i) m_i$$

$$k_i = 4\pi^2 f_i^2 m_i$$

If PARAM,KDAMP = -1 in complex eigenvalue analysis, modal frequency response, aerodynamic flutter, and aeroelastic response the viscous modal damping will be incorporated into the complex stiffness matrix as follows:

$$m_i = \text{modal mass}$$

$$b_i = \mathbf{0}$$

$$k_i = [1 + ig(f_i)] 4\pi^2 f_i^2 m_i$$

Eq. 7-122

Values of $g(f_i)$ are specified on the TABDMP1 Bulk Data entry, selected by the SDAMPING Case Control command. If KDAMP = 1 (default), the matrices $[b]$ and $[k]$ are formulated as in modal transient response.

In SubDMAP CMPMODE damping of CMS(o-set) modes is calculated by similar techniques. This q-set damping is expanded to g-set size, and placed in the B_{aa} matrix for summing into downtown superelements.

7.4.11 Data Recovery Operations in SubDMAP SEDISP

The dependent displacements and the single-point forces of constraint are calculated in SDR1 (Stress Data Recovery -- Phase 1). The inputs to SDR1 are columns of the solution vectors corresponding to each eigenvalue, loading condition, or output time step. In the case of transient analysis, there are three columns, corresponding to displacement, velocity and acceleration, for each output time step.

Load Modification

If PARAM,INREL,-1 or PARAM,INREL,-2 are specified, then any component of the load vector is first multiplied by $[C_{rx}]$ to produce a set of loads that are a linear combination of applied and inertial loads called $\{P_i^i\}$

$$\{P_i^i\} \leftarrow [P_l][C_{rx}] \quad \text{Eq. 7-123}$$

Static Analysis

In the case of static analysis and PARAM,ALTRED,NO, SDR1 recovers dependent displacements

$$\begin{Bmatrix} u_l \\ - \\ u_r \end{Bmatrix} \Rightarrow \{u_a\} \quad \text{Eq. 7-124}$$

$$\begin{Bmatrix} u_a \\ - \\ u_o \end{Bmatrix} \Rightarrow \{u_f\} \quad \text{Eq. 7-125}$$

where

$$\{u_o\} = [G_{oa}^t]\{u_a\} + \{u_o^o\} \quad \text{Eq. 7-126}$$

If PARAM,ALTRED,YES is specified, then the displacements $\{u_o\}$ are obtained from a backward pass from the equation

$$[L_{oo}]^T \{u_o\} = \{u_o^y\} \quad \text{Eq. 7-127}$$

where $\{u_o^y\} = \{u_o^x\} - [L_{ao}]^T \{u_a\}$

Enforced displacements are combined with u_f

$$\begin{Bmatrix} u_f \\ - \\ Y_s \end{Bmatrix} \Rightarrow \{u_n\} \quad \text{Eq. 7-128}$$

$$\begin{Bmatrix} u_n \\ - \\ u_m \end{Bmatrix} \Rightarrow \{u_g\} \quad \text{Eq. 7-129}$$

where

$$\{u_m\} = [G_{mn}]\{u_n\} \quad \text{Eq. 7-130}$$

and recovers single point forces of constraint

$$\{q_s\} = [K_{fs}]^T [u_f] + [K_{ss}]\{Y_s\} - \{P_s\} \quad \text{Eq. 7-131}$$

and multipoint forces of constraint

$$q_{mg} = \begin{bmatrix} I_{mm} \\ -G_{mn}^t \end{bmatrix} [[K_{mg}]\{u_g\} - \{P_m\}] \quad \text{Eq. 7-132}$$

In superelement static analysis, if PARAM,FIXEDB,-1 is present, then u_a is set to zero, and data recovery is performed using the modified equation

$$\{u_o\} = \{u_o^o\} \quad \text{Eq. 7-133}$$

This is equivalent to obtaining a solution with all exterior (“boundary”) degrees-of-freedom fixed. Displacements are due only to interior displacements due to interior loads, that is, the $\{u_o^o\}$ component.

Eigenvalue Analysis

In the case of eigenvalue analysis SDR1 recovers dependent components of the eigenvectors

$$\{\phi_o\} = [G_{oa}]\{\phi_a\} \quad \text{Eq. 7-134}$$

$$\begin{Bmatrix} \phi_a \\ \phi_o \end{Bmatrix} = \{\phi_f\} \quad \text{Eq. 7-135}$$

$$\begin{Bmatrix} \phi_f \\ \phi_s \end{Bmatrix} = \{\phi_n\} \quad \text{Eq. 7-136}$$

$$\{\phi_m\} = [G_{mn}]\{\phi_n\} \quad \text{Eq. 7-137}$$

$$\begin{Bmatrix} \phi_n \\ \phi_m \end{Bmatrix} = \{\phi_g\} \quad \text{Eq. 7-138}$$

and single-point forces of constraint

$$\{q_s\} = [K_{fs}]^T \{\phi_f\} \quad \text{Eq. 7-139}$$

If PARAM,DYNSPCF,NEW (default) then the single-point forces are computed as follows:

$$\{q_s\} = [[K_{sf}] - \omega^2[M_{sf}]]\{\phi_f\} \quad \text{Eq. 7-140}$$

In superelement analysis, if PARAM,FIXEDB,-1 is present, the uncoupled eigenvectors are recovered from the modified equations

$$\{\phi_a\} = \{\phi_{az}\} \quad \text{Eq. 7-141}$$

$$\{\phi_o\} = \{\phi_{oz}\} \quad \text{Eq. 7-142}$$

with $\{\phi_{az}\}$ and $\{\phi_{oz}\}$ from “[Superelement Analysis](#)” on page 444.

The multipoint forces of constraint qm_q are recovered by

$$qm_q = \begin{bmatrix} I_{mm} \\ -G_{mn}^T \end{bmatrix} (-\omega^2 M_{mg} + K_{mg}) \quad \text{Eq. 7-143}$$

Dynamic Analysis

In the case of dynamic analysis with the direct formulation the extra points are partitioned out of $\{u_d\}$.

$$\begin{Bmatrix} u_a \\ u_e \end{Bmatrix} = \{u_d\} \quad \text{Eq. 7-144}$$

SDR1 recovers the dependent components

$$\{u_o\} = [G_{oa}]\{u_a\} \quad \text{Eq. 7-145}$$

$$\begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \{u_f\} \quad \text{Eq. 7-146}$$

and recovers single point forces of constraint

$$\{q_s\} = [K_{fs}]^T \{u_f\} - \{P_s\} \quad \text{Eq. 7-147}$$

If PARAM,DYNSPCF,NEW (default) then the forces are computed as follows.

In transient response analysis, the formulation is

$$q_s = M_{sf}^x \ddot{u}_f + B_{sf}^x \dot{u}_f + K_{sf}^x u_f - P_s \quad \text{Eq. 7-148}$$

where:

$$M_{sf}^x = M_{sf} + M_{sf}^{2pp}$$

$$B_{sf}^x = B_{sf}^x + B_{sf}^{2pp} + \frac{g}{\omega_3} K_{sf} + \frac{1}{\omega_4} K_{sf}^{4gg}$$

$$K_{sf}^x = K_{sf} + K_{sf}^{2pp}$$

In frequency response and complex eigenvalue analysis, the formulation is

$$q_s = (-\omega^2 M_{sf}^x + i\omega B_{sf}^x + K_{sf}^x) u_f - P_s \quad \text{Eq. 7-149}$$

where:

$$M_{sf}^x = M_{sf} + M_{sf}^{2pp}$$

$$B_{sf}^x = B_{sf}^x + B_{sf}^{2pp}$$

$$K_{sf}^x = (1 + ig)K_{sf} + K_{sf}^{2pp} + iK_{sf}^{4gg}$$

The multipoint forces of constraint qm_g are recovered

$$qm_g = \begin{bmatrix} I_{mm} \\ -G_{mn}^T \end{bmatrix} (X_m - P_m) \quad \text{Eq. 7-150}$$

where q_{m_g} is the reactions expanded to the g-set and X_m and P_m are the internal and external loads on the m-set degrees-of-freedom, respectively. (See also Eq. 9 in Section 3.5.1 in *The NASTRAN Theoretical Manual*).

In transient response analysis,

$$X_m = M_{mg}^x \ddot{u}_g + B_{mg}^x \dot{u}_g + K_{mg}^x u_g \quad \text{Eq. 7-151}$$

where:

$$M_{mg}^x = M_{mg} + M_{mg}^{2pp}$$

$$B_{mg}^x = B_{mg} + B_{mg}^{2pp} + \frac{g}{\omega_3} K_{mg} + \frac{1}{\omega_4} K_{mg}^{4gg}$$

$$K_{mg}^x = K_{mg} + K_{mg}^{2pp}$$

In frequency response and complex eigenvalue analysis,

$$X_m = (-\omega^2 M_{mg}^x + i\omega B_{mg}^x + K_{mg}^x) u_g \quad \text{Eq. 7-152}$$

where:

$$M_{mg}^x = M_{mg} + M_{mg}^{2pp}$$

$$B_{mg}^x = B_{mg} + B_{mg}^{2pp}$$

$$K_{mg}^x = (1 + ig)K_{mg} + K_{mg}^{2pp} + iK_{mg}^{4gg}$$

For frequency response there is one complex vector for each loading condition. For complex eigenvalue analysis there is one complex eigenvector for each complex root and there is no loading term $\{P_s\}$ associated with the determination of the single point constraints. For transient analysis there is a displacement, velocity, and acceleration associated with each output time step.

In the case of dynamic analysis with the modal formulation, the user may obtain improved element stresses by requesting the mode acceleration data recovery technique.

The use of the mode acceleration technique requires the decomposition of the l -set stiffness matrix $[K_{ll}]$. This decomposition is not otherwise performed in dynamic analysis unless the model has rigid body modes. Also, the technique requires a forward/backward substitution operation with a number of righthand sides equal to the number of solution vectors. A subset of the solution vectors can be obtained can

be obtained by using the OTIME or OFREQUENCY entries in the Case Control Section. The subset of solution vectors is prepared by the MODACC functional module.

In the case of dynamic analysis with the modal formulation, two methods of data recovery are available. The default procedure is usually much more efficient for data recovery operations when the modal formulation is used for dynamic analysis. In this procedure, the dependent components of the eigenvectors used in the modal formulation are first recovered by SDR1 using the same sequence of operations described for problems in real eigenvalue analysis (after eliminating unwanted modes). The complete eigenvectors are used as the input to SDR2 in order to determine the requested forces and stresses in terms of modal coordinates.

In dynamic analysis, the user may restrict the output request to the response of selected points in the solution set. In this case, the time-consuming operations in the recovery of the dependent degrees-of-freedom can be avoided. This solution set processing is performed by VDR (Vector Data Recovery).

The modal data generated by SDR1 and SDR2 are formed into a data matrix by DDRMM (Dynamic Data Recovery Matrix Method). The output quantities are determined by multiplying the data matrix by the modal solution vectors. The computer time for this matrix multiplication is proportional to the number of output times, the number of modes used in the modal formulation, and the number of output frequencies or output time steps. If both the number of output items and the number of modes are small, the computer time required will not be significant when compared to the total problem solution time. Although the DDRMM method of data recovery is more efficient, a complete $\{u_g\}$ is not formed. For this reason the method does not permit the creation of deformed structure plots, grid point force balance or the use of the mode acceleration technique.

The alternate procedure may be invoked using PARAM,DDRMM,-1. With this method the solution vectors are first transformed from modal to physical coordinates

$$\{u_d\} = [\phi_{dh}]\{u_h\} \quad \text{Eq. 7-153}$$

This operation is performed by MPYAD in subDMAP MODACC. Having determined the vectors $\{u_d\}$ in physical coordinates, the data recovery operations can be completed using the same sequence of operations described for the direct formulation of dynamics problems.

Most of the operations in the recovery of the dependent components do not require significant amounts of computer time. However, in the case of the multiply-add operation associated with the recovery of the omitted coordinates ($[G_{oa}]\{u_a\}, [G_{oa}]\{\phi_a\}$) a significant amount of computer time will be required if there is

a large number of vectors and a large number of components in the a-set. This operation is likely to be especially costly for transient problems because there is usually a large number of output time steps and there are three vectors for each time step.

7.4.12 Data Recovery Operations in SubDMAP SEDRCVR

The element forces and stresses are calculated in SDR2 (Stress Data Recovery -- Phase 2). The matrix operations in this phase of data recovery are similar to those performed in EMG for stiffness matrix generation. The computer time will be proportional to the number of elements for which output is requested.

If static loads are applied to intermediate points on the CBAR element, the user may request the output of stresses and forces at a selected set of intermediate points. The additional calculations required for output at intermediate points on CBAR elements is performed in functional module SDRX.

The solutions and data recovery operations for transient analysis are initially sorted by the time step (SORT1). A transpose operation is performed by SDR3 (Stress Data Recovery -- Phase 3) in order to prepare the requested output sorted by external point identification number or external element identification number (SORT2). In frequency response problems, the user may request either SORT1 or SORT2 output. The SORT2 output will usually be desirable if the solution is made for a large number of frequencies. In any event, the SORT2 output must be prepared if XY-plots are requested.

Computer time for the preparation of SORT2 output will be directly related to the number of items requested for output. If a large number of items are requested for output, the transpose operation will require a significant amount of computer time.

The user may request the output of a grid point force balance table for a selected set of points. This table lists the forces acting at each selected point due to element forces, single point forces of constraints and applied loads. The grid point force balance table is prepared by GPFDR (Grid Point Force Data Recovery).

The user may also request the output of an element strain energy table and/or an element kinetic energy table for a selected set of elements. (See the Case Control commands, “**EDE**” on page 251, “**EKE**” on page 253, and “**ESE**” on page 261 of the *MSC.Nastran Quick Reference Guide*.) The tables include the energy for each element, the percentage of the total energy for each element, and the energy density for each element. The energy density is computed by dividing the element energy by the element volume. Some elements do not have enough data input to allow calculation of element volume, such as CELASi elements. The density output for these elements

is set to zero. Simplified equations for other element volumes may be used. For example, offsets are ignored for bar-like elements, and midside node locations are ignored for plate and solid elements. Element volumes are calculated by the ELPRT module, and can be output by use of PARAM,EST,1. The strain energy table is prepared by functional module GPFDR.

7.5 Summary of Solution Sequence Operations

This section describes the data processing and algorithms that are performed by special purpose functional modules, sequences of functional modules (subDMAPs) and complete solution sequences.

Detailed descriptions of the functional modules are given in the *MSC.Nastran 2005 DMAP Programmer's Guide* and Chapter 4 of the *MSC.Nastran Programmer's Manual*.

7.5.1 Static Solutions

There are several major paths for static solutions and some minor ones. For standard static analysis the stiffness matrix is decomposed in SubDMAP SEKRRS, in preparation for a linear solution in SubDMAP STATRS. There is also a form of static analysis called inertia relief analysis.

There are now several paths through inertia relief analysis, reflecting that it is still in a state of development, with more changes likely in future systems. This revision of this manual will discuss inertia relief in SOL 101 only. Each path has advantages and limitations. There is also a similar but different method used in SOL 1, using obsolete modules and with all of the limitations of SOL 1, most notably lack of superelement capability. That method has not changed in many years, and is adequately described in the first edition of this manual. It will not be documented here. There is still another method in SOL 144, Static Aeroelasticity. It is documented elsewhere.

All methods of inertia relief analysis take the standard static analysis equation

$$K_{aa} \cdot u_a = P_a \quad \text{Eq. 7-154}$$

where P_a are the applied loads, and add a set of constraint equations R_{ra} and automatically-computed inertial loads P_i . The constraints are required because K_{aa} is for a free structure and cannot be solved reliably without some form of constraint.

$$K_{aa} \cdot u_a = P_a - P_i = P_{ai} \quad \text{Eq. 7-155}$$

$$R_{ra} \cdot u_a = 0 \quad \text{Eq. 7-156}$$

In theory, if P_{ai} is a set of balanced loads (no resultant) the solution of u_a is possible without the need for constraint equations. In practice, P_{ai} may have some unbalance due to numerical truncation effects. Even a small resultant load will provide a displacement solution at infinity in theory, or with very large numbers in practice, usually. The same truncation effects that unbalance the loads may provide enough "ground springs" in the stiffness matrix to allow a solution, but it will be a poor

quality solution. Everything will first move 1.E5 units in the same direction. Stresses will be determined by subtracting numbers of this size to find element displacement perhaps on the order of 1.E-3. Ironically, if all goes well the forces in the support system are miniscule, but it must be present to control stability when truncation errors are present.

Types of Inertia Relief

Inertia Relief Analysis is a quasi-dynamic analysis. It is assumed that the structure is not tied to ground and is in a state of steady acceleration caused by the applied loads. The displacements measured relative to ground are constantly increasing and cannot be determined. Displacements are therefore determined with respect to an accelerating reference frame that is attached to the structure. The applied loads are balanced by inertial loads that are calculated automatically from the steady acceleration acting on the mass properties of the model. Inertia relief analysis is requested with `PARAM,INREL,[x]`, where [x] may have the values -1 or -2.

The reference frame may be obtained by two different methods. For the manual support method (`PARAM,INREL,-1`) the user selects 6 or fewer DOFs using `SUPPORTi` entries. The reference frame is defined from these DOFs. For the auto support method (`PARAM,INREL,-2`) 6 constraint equations are written that constrain the structure to zero motion, on average, while leaving it free to “breath” and deflect due to the applied loads and inertial loads. Both methods find it convenient to have “rigid body modes shapes” that represent a set of vectors that represent all possible strain-free motion of the structure when not loaded. The technique used to obtain rigid body mode shapes differs between the two methods.

Rigid Body Modes Using Manual Support (`PARAM,INREL,-1` and `SUPPORTi`)

The user prescribes a set of DOFs that constrain the structure in a statically-determinate manner on `SUPPORTi` entries. These entries define the *r*-set. There must be six or less DOFs listed and they must constrain all rigid body modes that are not constrained by SPCs or other modeling devices. The rigid body modes are computed from the static analysis equation. There are no loads applied and I_{rr} is the identity matrix. Given

$$\begin{bmatrix} K_{ll} & K_{lr} \\ K_{lr}^T & K_{rr} \end{bmatrix} \begin{bmatrix} D_{lr} \\ I_{rr} \end{bmatrix} = \begin{bmatrix} O_{lr} \\ O_{rr} \end{bmatrix} \quad \text{Eq. 7-157}$$

D_{lr} is found by solving the first row of this equation, then merged with I_{rr} to form D_{ar} ,

$$K_{ll} \cdot D_{lr} \leq -K_{lr} \cdot I_{rr} \quad \text{Eq. 7-158}$$

$$[D_{ar}] = \begin{bmatrix} D_{lr} \\ I_{rr} \end{bmatrix} \quad \text{Eq. 7-159}$$

Rigid Body Modes Using Auto-Support (PARAM,INREL,-2)

Vectors of rigid body motion D_{gr} for six unit motions of a reference point are computed from grid point geometry in module VECPLOT. The D_{ar} component is partitioned from this matrix.

The Basic Equation of Motion for Inertia Relief

Inertia relief analysis starts with the a-set equation

$$K_{aa} \cdot u_a \leq P_{ai} \quad \text{Eq. 7-160}$$

and adds a constraint equation

$$R_{ar}^T \cdot u_a = 0 \quad \text{Eq. 7-161}$$

The last equation implies a similar equation on a generalized constraint force variable vector, q_r

$$R_{ar} \cdot q_r = P_{ai} - K_{aa} \cdot u_a \quad \text{Eq. 7-162}$$

If P_{ai} represents balanced loads, q_r is zero. If there is a small unbalance due to numerical truncation effects, q_r is then small, but finite. There are two methods for defining R_{ar} , as discussed below. Both have the property of defining some part of u_a to have zero motion, so that the other parts of u_a are in effect measured relative to the constrained DOFs.

One of the solution methods partitions K_{aa} into its r -set and l -set components. The first equation is partitioned, and the a-set equations and the constraint equation may combine into one equation,

$$\begin{bmatrix} K_{ll} & K_{lr} & R_{rl}^T \\ K_{lr}^T & K_{rr} & R_{rr}^T \\ R_{rl} & R_{rr} & O_{rr} \end{bmatrix} \cdot \begin{bmatrix} u_l \\ u_r \\ -q_r \end{bmatrix} = \begin{bmatrix} P_{ail} \\ P_{air} \\ Y_r \end{bmatrix} \quad \text{Eq. 7-163}$$

The methods of computing P_i are described below. Y_r , the generalized displacement input, is set to zero.

Constraint Equations Using Manual Supports

In the constraint equation $R_{rr} = I_{rr}$, the identity matrix. R_{rl} is null. The last row can then be solved for $u_r = Y_r = 0$. The second equation can then be used to eliminate q_r ,

$$K_{lr}^T \cdot u_l - I_{rr} \cdot q_r = P_{air} \quad \text{Eq. 7-164}$$

$$q_r = K_{lr}^T \cdot u_l - P_{air} \quad \text{Eq. 7-165}$$

Because R_{rl} is null, the first equation then becomes

$$K_{ll} \cdot u_l = P_{ail} \quad \text{Eq. 7-166}$$

This equation can be solved for u_l reduced from u_a directly. It will be shown later that P_r terms are transferred to P_l terms in the inertial loads. u_a is formed from a merge of u_l and null terms for u_r .

Constraint Equations Using Automatic Supports

A constraint equation is written that states that the average weighted motion of the system is zero,

$$R_{ra} = D_{ar}^T \cdot M_{aa} \quad \text{Eq. 7-167}$$

$$R_{ra} \cdot u_{ar} = 0 \quad \text{Eq. 7-168}$$

R_{ra} is generally a dense matrix, with non-zero terms at most DOFs with mass. It could be used to reduce the K_{aa} matrix with equations similar to those used for multi-point constraints, but that would produce a dense K_{ll} matrix, much more expensive to decompose and solve than the K_{ll} used in the manual support method. A solution method is used which increases solution costs only slightly.

The r -set l -set partitioning is not needed. The constraint equation is appended to the a -set equations to define the l -set equations in their augmented form,

$$K_{ll} \cdot u_l = P_l \quad \text{Eq. 7-169}$$

$$\begin{bmatrix} K_{aa} & R_{ra}^T \\ R_{ra} & 0_{rr} \end{bmatrix} \begin{bmatrix} u_a \\ -q_r \end{bmatrix} = \begin{bmatrix} P_{al} \\ 0_r \end{bmatrix} \quad \text{Eq. 7-170}$$

q_r are the unknown forces required on the system to produce equilibrium. They should be computational zeros, and are output with the SPC forces. The equation involving the augmented K_{ll} is solved directly for u_l . u_a and q_r are then partitioned from u_l . In superelement analysis the constraints are applied in the residual structure only.

Matrix Decomposition in SubDMAP SEKRRS

There are two major branches through this SubDMAP, standard static analysis and static analysis with inertia relief effects.

Inertia Relief Not Selected

When inertia relief is not selected and there are no SUPORTi entries present the a -set is decomposed directly in the DCMP module,

$$K_{aa} = L_{aa} \cdot D_{aa} \cdot L_{aa}^T \quad \text{Eq. 7-171}$$

L_{aa} is lower triangular, and D_{aa} is diagonal. Both are stored in an unconventional (“packed”) format, and output in a single data block named LLL. This name is used because the l -set and the a -set are synonymous when no SUPORTi entries exist. The factor matrices L_{aa} and D_{aa} are passed on to SubDMAP STATRS for solution of u_a .

When inertia relief is not selected but there are SUPORTi entries present, the a -set equations are partitioned into the r -set, defined on the SUPORTi entries, and the l -set, what is “left over” after the r -set is removed.

$$\begin{bmatrix} K_{ll} & K_{lr} \\ K_{rl} & K_{rr} \end{bmatrix} \begin{bmatrix} u_l \\ u_r \end{bmatrix} \geq \begin{bmatrix} P_l \\ P_r \end{bmatrix} \quad \text{Eq. 7-172}$$

In this context, the r -set is treated as an alternate form of single-point constraints. The partitions of K involving r are discarded. u_r is set to zero. u_l is found from the first row of this equation. Loads in P_r go directly to ground, and are output with the SPC forces, along with the product $-K_{rl} \cdot u_l$ from the second equation. K_{ll} is decomposed with the same equations used for conventional static analysis.

Decomposition of K_{ll}

For the manual support method K_{ll} is defined above. Its factors, used to compute the rigid body modes are also used later to compute the static solution. For the auto support method the factors of the expanded form of K_{ll} are passed on.

Solution in SubDMAP STATRS

Computation of Inertial Loads

The basic assumption of inertia relief is that the structure is in a state of steady acceleration and that all transients due to dynamic effects are negligible. The solution is obtained in an accelerating reference frame from which all displacements are measured. Inertial loads are automatically computed that exactly balance the applied loads. A support system is applied to make the system numerically stable. The inertial loads assume rigid body accelerations when computing the loads.

The derivations given here assume that the system mass matrix M_{gg} has been reduced to M_{aa} and is available, as described elsewhere. In the actual DMAP used in MSC.Nastran 2001 equivalent methods that never generate M_{aa} are used instead, in the interests of efficiency. Unit inertial loads are computed from M_{gg} , then reduced by load reduction techniques, avoiding the need for M_{aa} . M_{aa} can be a large size data block, and costly to compute. It can be shown that the results provided by the two derivations are identical. For example, the rigid body mass matrix M_{rr} , defined below as

$$M_{rr} = D_{ar}^T \cdot M_{aa} \cdot D_{ar} \quad \text{Eq. 7-173}$$

can just as easily be formed from the g -set equations

$$M_{rr} = D_{gr}^T \cdot M_{gg} \cdot D_{gr} \quad \text{Eq. 7-174}$$

D_{ar} is used to compute the rigid body mass M_{rr} about the reference point.

$$M_{rr} = D_{ar}^T \cdot M_{aa} \cdot D_{ar} \quad \text{Eq. 7-175}$$

M_{rr} is printed out with the matrix name of QRR.

The acceleration of the reference point a_r due to the applied loads is computed,

$$M_{rr} \cdot a_r = D_{ar}^T \cdot P_a \quad \text{Eq. 7-176}$$

The acceleration a_r are printed with the matrix name of URA.

The resulting inertial loads P_{ai} and total loads P_{al} are

$$P_i = M_{aa} \cdot a_r \quad \text{Eq. 7-177}$$

$$P_{ai} = P_a - P_i \quad \text{Eq. 7-178}$$

For both methods, the mass matrix is reduced from M_{gg} size to M_{rr} size with load vector operations that can be shown to be identical to the M_{aa} operations described here, except for the order of operations. The load and mass transformations are based on rigid body modes produced from geometry. Readers interested in the details of this approach may review the original release of this manual. The load vector approach for mass reduction is likely to be dropped in future versions, in the interests of more automation and better maintainability.

A Comparison of Auto- and Manual-Support Solutions

When the same model is analyzed with both methods you should find that stresses, constraint forces, and other internal quantities are identical except for truncation errors, but the displacements may differ appreciably. This is because they use a different reference frame. For the manual supports, the reference frame is set up so that the motion at the r -set DOFs are identically zero. One can think of a reference frame drawn through these points, and all displacements measured relative to it. This reference frame is moving through space at a constant acceleration rate, so that displacements are relative rather than inertial displacements. There are many sets of SUPPORT i entries valid for one model. If you change SUPPORT i DOFs you may also expect to see changes in displacements, but not in internal loads or element stresses.

For the auto-support option, one can visualize the reference frame as a set of axes connected to every point with mass with a very complicated set of levers that allow the structure to “breathe” as loads are applied to it, but keep the weighted average motion to a zero value. If the model happens to have a grid point at the reference point, you will observe very small motion there, computational zeros but not the binary zeros of the manual support method. This is because this motion is computed for the auto-support method, but assumed to be zero for the manual method.

The choice of an *r*-set as defined by SUPPORTi DOFs requires meeting both a geometric requirement and a stiffness requirement. This geometric requirement is that the set should form a statically determinate tie to ground. This can be difficult to do with complicated geometries. The stiffness requirement is that the DOFs should be stiff enough to provide a stable solution. A method that has been used by some modelers is to pick points stiff points do not exist. The difficulties in picking a stable, reliable support system was one of the main motivations for development of the more modern auto-support system. SUPPORTi entries are not allowed for the auto support method, and will cause a fatal error exit when present.

PARAM,GRDPNT,GID is optional for both methods when superelements are not present. It is required when superelements are present, and must be on a boundary point attached to all superelements. This same point should be used for the SUPPORTi DOFs for the manual support method. It defines the reference point used for generating the geometry-based rigid body vectors. The default for GRDPNT is GID=-1, which is a flag to use the basic origin as the reference point when computing rigid body modes. Use of a GID near the c.g. of the structure produces more esthetically pleasing results. The rigid body accelerations produced by the solution are then measured near the center of the structure, rather than at a point that may be outside the structure.

Applications of Inertia Relief

Although this capability could be used in an environment where the reference frame is accelerating steadily, such as a free-falling elevator, the more common usage is when the modeler attempts to apply loads in equilibrium. He wishes the net acceleration of the structure to be zero. A ship floating in the sea, with pressure loads balancing gravity loads, is an example. Developing balanced load sets can be tedious on large models with complex geometry. The accelerations printed out in the URA matrix can be used as a measure of load quality. If the accelerations are negligibly small, the load balance is adequate. If the accelerations are large, some adjustment of the loads sets may be necessary. For good inertia relief analysis a good mass distribution is needed. As the structural weight is usually only a small proportion of the total weight, this means that a realistic payload mass distribution must be modeled also.

Another use of inertia relief analysis becoming more prevalent with the advent of auto support is in checkout of a new model. Although six and only six rigid body modes are provided, they are provided in the form of a “flying waterbed”. To take an extreme example, suppose that a structure is made up of two components bolted together, but the bolts were not modeled, inadvertently. This model has twelve rigid body modes before the constraints are applied. Both components sit on the same waterbed. This

means that all singularities will be constrained, and the model will run to completion on the first pass. If loads are applied to one component only, the constraints will cause the other component to move too, in a strain-free manner. A plot of the motion of the model should reveal that there is no displacement continuity between the two components. This should make the cause of the unexpected behavior apparent. As the goal of this analysis is model checkout only the structure mass distribution is usually adequate for stable solution. The payload masses may be added later.

7.5.2 Real Eigenvalue Analysis in SubDMAPs SEMR3 and MODERS

The basic equation solved in these SubDMAPs is

$$[K - \lambda \cdot M] \cdot \phi = 0 \quad \text{Eq. 7-179}$$

where K and M are real and symmetric, and M is positive semi-definite. This equation is defined for the a-set variables when called from MODERS for overall structural modes of vibration, and for o-set variables when called from SEMR3 during CMS (Component Mode Synthesis) analysis. Many steps are taken in DMAP to better automate the removal of DOFs that are ill-defined, and to augment the mode shapes with residual flexibility effects. The SubDMAPs that perform these operations are described below.

Although a knowledge of DMAP is not necessary for dynamic analysis, such a study can increase the understanding of the steps that are performed. If the Version of MSC.Nastran that you are using is later than the Version documented in this manual (MSC.Nastran 2001), you can determine the differences by comparing the DMAP statements of the two systems.

Massless Mechanism Identification in SubDMAP MMFIND

SubDMAP MMFIND identifies potential instabilities in the matrices used for real eigensolutions. The K and M matrices presented to the READ module for real eigensolutions may contain massless mechanisms (MMs). A massless mechanism has a constraint shape X such that

$$X^T \cdot K \cdot X = \epsilon_k \quad \text{Eq. 7-180}$$

$$X^T \cdot M \cdot X = \epsilon_m \quad \text{Eq. 7-181}$$

$$X^T \cdot X > 0.0 \quad \text{Eq. 7-182}$$

If ϕ contains the eigenvectors of the system, then

$$X^T \cdot \phi = 0 \quad \text{Eq. 7-183}$$

$$X^T \cdot M \cdot \phi = 0 \quad \text{Eq. 7-184}$$

The constraint shapes have non-zero terms only where the eigenvectors have zero terms. The massless mechanism shapes can be thought of as "deficient" eigenvectors describing parts of the system that are unconnected (or very weakly connected) to other parts of the structure that have both stiffness and mass that are active in the well-conditioned eigenvectors. ϵ_k and ϵ_m may be binary zeroes, or computational zeroes due to truncation effects. An "eigenvalue" λ associated with massless mechanism shapes is

$$\lambda = \epsilon_k / \epsilon_m \quad \text{Eq. 7-185}$$

λ is indeterminate in the limit as the ϵ_i approach zero. It can be a very large number, a very small number, and either negative or positive. It is likely to change a large amount when small changes are made to the model. This is the definition of an unstable eigensolution. The assumption taken here is that the cause of this condition is either stiffness and/or mass effects inadvertently left out of this part of the model, with the result that the load paths are ill-defined. This load path is constrained, with a warning message to the user. That is, this mode of motion is later removed from the K and M matrices prior to eigensolution by steps equivalent to single point constraint elimination.

The massless mechanisms are defined from the shifted stiffness matrix $A = K + s \cdot M$, where s is a shift that the user may change from its default value with user parameter PARAM, SHIFT1, (default=1.234 Hz.). The default is chosen to make detection of rigid body modes as MMs unlikely. A is decomposed into its factors,

$$A = L \cdot D \cdot L \quad \text{Eq. 7-186}$$

The ratios of the diagonal terms of A over D are stored in a vector Ry. Terms of D approach a value of zero at the ends of massless mechanisms. Terms of Ry larger in magnitude than user parameter MAXRATIO are reset to unity and are kept in vector R1. It can be shown that each of these terms is at the end of a massless mechanism constraint shape. R1 is expanded into a set of loading vectors P , with one unit term per column. P is used along with the factors to find the constraint shapes,

$$A \cdot u \leq P \quad \text{Eq. 7-187}$$

The constraint shapes are normalized, and small numbers are discarded. Each column of u is then a potential constraint mode. It could also be a rigid body mode. The rigid body modes are detected by computing the diagonal terms of the generalized mass matrix for these shapes,

$$Mg = u \cdot M \cdot u \quad \text{Eq. 7-188}$$

Terms of $Mg(i, i)$ greater than PARAM, MECHFIL, default value = 1.e-6 are an indication that the corresponding column is a rigid body mode. Columns of this type in u are discarded to produce $u1$, the matrix of constraint mode shapes. The location of the end of the constraint shapes is passed back in R . The constraint mode shapes themselves are passed back in MECHU. R is used later to remove rows and columns in K and M to constrain the massless mechanisms. The nonzero terms in MECHU are used to indicate which DOFs will discard loads applied to them, because they will have zero coefficients in the true eigenvectors of the system, leading to a null contribution to generalized force for all modes. MECHLD contains P , and is intended for a feature not yet implemented.

Massless Mechanism Control in SubDMAP XREAD

At present, the auto-omit DMAP is duplicated at several places, followed by a call to XREAD. It is therefore an implicit part of the XREAD operations now, and may be moved to XREAD in a later version to reduce the duplication of code.

SubDMAP XREAD modifies the input matrices used for eigensolution, when needed, then restores the modified solutions back to the original size of the matrices. Solutions for both structural vibration and buckling analyses are provided. This discussion describes vibration analysis first, with a concluding section on changes made for a buckling solution.

Auto-Omit Operation

All of the eigensolution methods require that each column of $[K]$ plus $[M]$ have some nonnull terms. Any degree-of-freedom that has only null terms in both matrices is regarded as undefined and causes a fatal error. The GIV and HOU methods of eigenvalue extraction also require that the mass matrix be positive-definite. The MGIV, MHOU, and the Lanczos methods allow null columns in the mass matrix but are generally faster if the massless degrees-of-freedom are removed by the static reduction method described in “[Static Solutions in SubDMAP SEKRRS](#)” on page 391. However, the static reduction tends to increase the number of active columns in the a-set matrices. The INV, SINV, and Lanczos methods, by contrast, are

generally slower if the static reduction is performed on massless degrees-of-freedom, because these methods take advantage of sparse matrices, whereas the other methods do not.

Three options are provided to deal with the various null column possibilities. If PARAM,ASING,-1 is used, any degrees-of-freedom with null columns in both the stiffness and mass matrices (“undefined degrees-of-freedom”) are identified and a fatal error exit is taken. This is an option to use to identify modeling errors before performing an expensive eigenvalue solution.

If ASING = 0 (the default) the method used depends on the eigenvalue extraction method. In the tridiagonal methods of eigenvalue extraction, degrees-of-freedom with null columns in the mass matrix are identified and placed in the w-set.

$$[M] = \begin{bmatrix} M_{xx} & | & M_{xw} \\ \hline M_{wx} & | & M_{ww} \end{bmatrix} \quad \text{Eq. 7-189}$$

where $[M_{ww}]$ and $[M_{wx}]$ are null by definition. Degrees-of-freedom in the r-set (SUPPORT Bulk Data entry) must have mass, or a fatal error will result. The stiffness matrix is similarly partitioned.

$$[K] = \begin{bmatrix} \bar{K}_{xx} & | & K_{xw} \\ \hline K_{wx} & | & K_{ww} \end{bmatrix} \quad \text{Eq. 7-190}$$

with until diagonal terms placed on null columns of $[K_{ww}]$. The static condensation matrices are formed

$$[K_{ww}][G_{wx}] = -[K_{wx}] \quad \text{Eq. 7-191}$$

and

$$[K_{xx}] = [\bar{K}_{xx}] + [K_{wx}]^T [G_{wx}] \quad \text{Eq. 7-192}$$

This method will prevent errors due to mass matrix singularly in the tridiagonal methods if the singularity is caused by null columns in $[M]$. It will not detect singularities due to mechanisms (caused, for example, by a point mass offset from a grid point). This type of singularity does not necessarily prevent obtaining eigensolutions, but will cause poor numerical stability. The MGIV or MHOu methods do not suffer from numerical instability due to singular or nearly singular mas

matrices. They are therefore more reliable than the GIV HOU methods. In well conditioned problems, however, it is unnecessary to automatically select the appropriate method. The AHOU method is the most modern, reliable method.

The static reduction will reduce the cost of solution and minimize the number of uninteresting high roots found at the value of computed infinity due to the singularity of the mass matrix. Note that the static reduction used here does not introduce any approximation, as the “omitted” that is w-set terms have null mass values. The introduction of unit diagonal terms on null columns of $[K_{ww}]$ does not introduce approximations because the terms that multiply these diagonal terms (the corresponding rows of $[G_{??}]$ are null. The same results can be achieved at higher cost by first eliminating null rows and columns of $[K_{ww}]$ by the single point constraint operations of “[Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB](#)” on page 382 followed by static reduction.

The solution for the system $[K_{xx} - \lambda \cdot M_{xx}] \cdot \phi_x = 0$ is provided. Rigid body shapes D_{xr} are provided when SUPORTi entries exist, causing an r-set. The factor of K_{xx} , named LLL, is used for computing approximate solutions for buckling analysis, to better set the range of eigenvalues to be searched. For the tridiagonal methods, K and M have DOFs with zero mass removed by an auto-omit method in a prior subDMAP. For other methods of analysis, K_{aa} and M_{aa} are used directly for residual structure modes, and K_{vv} and M_{vv} for CMS (Component Mode Synthesis) modes.

The modes of the system are computed in the READ eigensolver module. If the solution appears to be of good quality the subDMAP returns with no other operations performed. If the modes can not be found, or appear to be of low quality, the module sets a flag (Neigv parameter). This flag causes the MMFIND subDMAP to be called. Given K and M , it finds a high-ratio vector R and a matrix of constraint mode shapes MMECHA. These terms are defined in the MMFIND section above.

There are non-zero terms in R at the end of massless mechanisms. These mechanisms can be removed with SPC-like operations using R as a partitioning vector,

$$K_{xx} = \begin{bmatrix} K22 & K21 \\ K12 & K11 \end{bmatrix} \quad \text{Eq. 7-193}$$

M_{xx} is partitioned similarly, to obtain $M22$. Partitions involving the "1" subscript are discarded. The modified eigenproblem solved is then $[K22 - \lambda \cdot M22] \cdot \phi_2 = 0$. ϕ_2 is merged to the size of the original K matrix by inserting null rows for the "1-set" DOFs,

$$\phi_x = \begin{bmatrix} \phi_2 \\ 01 \end{bmatrix} \quad \text{Eq. 7-194}$$

The auto-omit reduction is backed out by the operations

$$\phi_w = G_{wx} \cdot \phi_x \quad \text{Eq. 7-195}$$

$$\phi = \begin{bmatrix} \phi_x \\ \phi_w \end{bmatrix} \quad \text{Eq. 7-196}$$

Real Eigenvalue Analysis in the READ Module

Real eigenvalues and their associated eigenvectors are determined in READ (Real Eigenvalue Analysis -- Displacement). In the case of normal mode analysis, READ extracts real eigenvalues from the equation

$$[K - \lambda M]\{u\} = 0 \quad \text{Eq. 7-197}$$

There are seven basic eigensolution methods. The Inverse (INV and SINV) Method obtains eigensolutions by iterations based on the equation

$$[K - \lambda_i M]\{\phi_{i+1}\} = [M][\phi_i] \quad \text{Eq. 7-198}$$

where λ_i is an estimate of the eigenvalue. This method is best suited to large problems with sparse matrices, where only a few eigenvectors are desired.

The tridiagonal methods, Givens (GIV and MGIV) and Householder (HOU and MHOU), first mass-scale the equation

$$[M] = [L][L]^T \quad \text{Eq. 7-199}$$

where $[L]$ is a lower-triangular matrix. The $[K]$ and $[M]$ matrices are pre- and post-multiplied by $[L]^{-1}$ and $[L^{-1}]^T$, respectively, to obtain the standard form

$$[J - \lambda I]\{\bar{\phi}\} = 0 \quad \text{Eq. 7-200}$$

where $[J] = [L^{-1}][K][L^{-1}]^T$ and $[I]$ is the identity matrix. Note that the mass matrix is in effect inverted, so that it must be nonsingular. The $[J]$ matrix is transformed to tridiagonal form for economical computation of eigenvalues and eigenvectors. The eigenvectors in the tridiagonal basis are then back-transformed to the initial basis. The Modified Givens (MGIV) and Modified Householder (MHOU) Methods use the shifted matrix

$$[K + \lambda_s M] \quad \text{Eq. 7-201}$$

in place of $[M]$ in the discussion above, thereby allowing the mass matrix to be singular if there are compensating terms in the stiffness matrix. The GIV, MGIV, and MHOu methods are best suited to small problems, or for large problems where many eigenvectors are required, after extensive static reduction or dynamic reduction is performed.

The Lanczos method uses a block shifted algorithm. Sets of vectors obtained by a recursive form are used to reduce the problem to a reduced block tridiagonal form. The eigensolutions are computed in the reduced basis by a QL algorithm, then back-transformed to the original basis. This is currently the most modern method, and should be considered for all large-size problems. Since the method takes advantage of sparsity in the input matrices, it is most economical when used without static or dynamic reduction. It is requested with either the EIGRL or EIGR Bulk Data entries.

The AHOU method combines the best features of all of the tridiagonal methods, and is the recommended method at present. It first decomposes M . If the factors are stable, the HOU method is used. If the factors may be unstable, the MHOu method is used instead.

Modal Solution Preparation in the MODERS subDMAP

MODERS is called when solving for the eigensolution of the residual structure, with a-set operations. It is used for both structural mode and buckling analysis.

In buckling, the r-set caused by SUPORTi entries are treated as equivalent SPCs. The rows and columns associated with r-set DOFs are discarded before eigensolution.

For structural analysis, the auto-omit operations take place for the tridiagonal methods. See the XREAD Section for details.

There are also operations for ACMS and indexing of output tables present in MODERS. They can also be found in many other SubDMAPs. Their description is beyond the scope of this manual, but briefly, ACMS (Automated Component Mode Synthesis) is based on breaking a large model into superelements automatically, solving it in pieces, then re-assembling the pieces so that the output appears to be from a conventional model. Some of the partitioning burden that was placed on the user in manual superelement analysis proved difficult to automate, so more automatic alternate methods have been developed for ACMS that require less user input. The automatic support system described in the RESVEC SubDMAP section is a case in point. Most of the ACMS innovations are moved to conventional analysis, over time, after they have been tested in field usage.

Table indexing is a method to allow direct access to data in large sequential files, to speed up interactive analysis. This work is being done as part of the API project.

Calculation of Residual Vectors in MSC.Nastran

Modal solutions are commonly used to reduce large-size models to relatively smaller representations that include the important dynamics of the original model. In addition to reducing the model size, modal reduction has the added advantage of producing diagonal mass and stiffness matrices for solution calculation. Although modal reduction may capture most of the dynamic response of the structure, the static response may not be complete. This 'error' is due to modal truncation of higher frequency modes that may contribute 'statically' to the total response. Residual vectors are used to improve the results of modal solutions by attempting to account for the response of these higher frequency modes.

Residual vectors can be determined from any set of base vectors. Base vectors are the raw material used to generate the residual vectors in MSC.Nastran. Although any vector can be used as a residual vector, as long as it is partially independent of the modal vectors, it may produce coupling with the modal vectors when added to the modes for matrix reduction. This coupling destroys the diagonal properties of the reduced mass and stiffness matrices.

To ensure that the residual vector used in the reduction process result in diagonal matrices, and better results, the following steps are performed.

1. Ensure that loads are linearly independent with the modal inertial forces.
2. Determine base vectors from static response due to loads.
3. Ensure that base vectors are linearly independent.
4. Orthogonalize the base vectors with respect to the modal vectors to produce residual vectors. These vectors will result in diagonal mass and stiffness matrices.

The closer the base vectors approximate the actual deformation due to the load, the better the residual vectors and the better the results. For this reason, base vectors in MSC.Nastran are determined from the static response of the structure to the following loads:

- Inertial forces due to rigid-body motion
- Applied loads
- Structural, viscous, and inertial forces due to enforced motion
- Forces at user specified discrete degrees of freedom (RVDOFi entries)
- Discrete damping forces due to viscous elements (CDAMPi and CVISC entries)

A new case control command, RESVEC, allows the user to specify the loads used for residual vector calculation. The format for the RESVEC is as follows:

$$\text{RESVEC} \left[\left(\begin{array}{l} \text{[INRLOD]} \\ \text{[NOINRL]} \end{array} \right), \left(\begin{array}{l} \text{[APPLOD]} \\ \text{[NOAPPL]} \end{array} \right), \left(\begin{array}{l} \text{[RVDOF]} \\ \text{[NORVDO]} \end{array} \right), \left(\begin{array}{l} \text{[DMPLOD]} \\ \text{[NODMP]} \end{array} \right) \right] = \left\{ \begin{array}{l} \text{SYSTEM/NOSYSTEM} \\ \text{COMPONENT/NOCOMPONENT} \\ \text{BOTH or YES} \\ \text{NO} \end{array} \right\}$$

Examples:

RESVEC=SYSTEM

RESVEC(NOINRL)=COMPONENT

RESVEC=NO

Describer	Meaning
INRLOD/ NOINRL	Controls calculation of residual vectors based on inertia relief. (Default=INRLOD)
APPLOD/ NOAPPL	Controls calculation of residual vectors based on applied loads. (Default=APPLOD)
RVDOF/ NORVDOF	Controls calculation of residual vectors based on RVDOFi entries (Default=RVDOF)
DMPLOD/ NODMP	Controls calculation of residual vectors based on viscous damping.
SYSTEM/ NOSYSTEM	Controls calculation of residual vectors for system (a-set) modes. For NOSYSTEM, describers inside the parentheses are ignored. See Remark 2 for default.
COMPONENT/ NOCOMPONENT	Controls calculation of residual vectors for component (superelement or o-set) modes. For NOCOMPONENT, describers inside the parentheses are ignored. See Remark 2 for default.
BOTH or YES	Requests calculation of residual vectors for both system modes and component modes. See Remark 2 for default.
NO	Turns off calculation of residual vectors for both system and component modes and describers inside the parentheses are ignored. See Remark 2 for default.

The inertial forces due to the normal mode shapes are removed from the loads used for residual vector calculation using the following:

$$\{\bar{P}\} = \{P\} - [M][\phi]([\phi][M][\phi])^{-1}[\phi]\{P\}$$

The modified loads are applied to the structure to determine the deformations that serve as the base vectors for residual vector calculation.

$$[K]\{V_{load}\} = \{\bar{P}\}$$

Additionally, for transient response, initial conditions are also added to the set of base vectors. Before the initial condition vectors are added, any mode shape content is removed.

$$\{\bar{V}_{initial}\} = \{V_{initial}\} - [\phi]([\phi][M][\phi])^{-1}[\phi][M]\{V_{initial}\}$$

The initial condition vectors are appended to the load vectors and the complete set of base vectors are made linearly independent. This process removes vectors that are linear combinations of other vectors in the base vector set. The remaining base vectors are then made orthogonal with respect to the mass and stiffness matrices. The resulting orthogonal vectors are the residual vectors.

The residual vectors are appended to the modes and the reduced mass and stiffness matrices are modified to include the residual vector additions.

$$[\bar{\phi}] = [\phi_{modes} | \phi_{residual}]$$

$$[\bar{M}] = [\bar{\phi}][M][\bar{\phi}]$$

$$[\bar{K}] = [\bar{\phi}][K][\bar{\phi}]$$

The augmented set of modes/residual vectors is used to reduce the damping, applied loads, and any external matrix input.

Residual vector calculation is available for all modal solutions in MSC.Nastran. They can be calculated for the residual structure and superelements to improve the analysis results.

New C-Set and R-Set Reduction with Residual Vectors

In previous version of MSC.Nastran, the o-set (interior dofs) motion associated with the b-set, c-set, and r-set (boundary dofs) was generated from the static reduction of the stiffness matrix.

$$[G_{ot}] = [G_{ob} | G_{ow}] = [-K_{oo}]^{-1}[K_{ob} | K_{ow}] \quad \text{Eq. 7-202}$$

where the w-set = c-set + r-set.

The boundary motion determines the response of the o-set.

$$\{u_0\} = [G_{ob} | G_{ow}] \begin{Bmatrix} u_b \\ u_w \end{Bmatrix} \quad \text{Eq. 7-203}$$

For dynamic analyses, the o-set motion can be augmented by the use of component modes to improve the response estimate. The motion of the component modes is represented by the q-set.

$$\{u_0\} = [G_{ob} | G_{ow} | \bar{\phi}_{oq}] \begin{Bmatrix} u_b \\ u_w \\ \xi_q \end{Bmatrix} \quad \text{Eq. 7-204}$$

where the component modes, $\bar{\phi}_{oq}$, are determined from modes of the v-set (v-set = o-set + c-set + r-set).

The modes of the v-set are calculated in the standard fashion.

$$(-\lambda[M_{vv}] + [K_{vv}])\{\phi_v\} = \{0\} \quad \text{Eq. 7-205}$$

The motion of the boundary grids are then removed from the calculated modes

$$\{\bar{\phi}_o\} = \{\phi_o\} - [G_{ow}]\{\phi_w\} \quad \text{Eq. 7-206}$$

where

$$\{\phi_v\} = \begin{Bmatrix} \phi_o \\ \phi_w \end{Bmatrix}$$

$\{\phi_v\}$ are modes of the v-set dofs

$\{\bar{\phi}_o\}$ are the v-set modes with the w-set boundary motion removed

The previous operation decouples the w-set boundary motion $[G_{ow}]$ from the v-set modes $[\phi_v]$. This decoupling allows the boundary and component mode dofs to move independently and the total motion of the structure to be written in terms of the a-set (b-set + w-set + q-set).

$$\begin{Bmatrix} u_b \\ u_w \\ u_o \end{Bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ G_{ob} & G_{ow} & \bar{\phi}_{oq} \end{bmatrix} \begin{Bmatrix} u_b \\ u_w \\ \xi_q \end{Bmatrix} \quad \text{Eq. 7-207}$$

If residual vectors are requested, they are calculated using the o-set matrices and appended to the component modes.

The new w-set processing follows a slightly different path. The procedure is based on a method suggested by Dr. Arya Majed and Ed Henkel. This method is the new default procedure for c-set reduction. The previous method for c-set reduction can be selected by setting PARAM,MHRED to 'NO'.

The new c-set procedure uses static reduction to determine the motion of the v-set due to boundary b-set motion.

$$\{u_v\} = [G_{vb}]\{u_b\} \quad \text{Eq. 7-208}$$

where

$$[G_{vb}] = [-K_{vv}]^{-1}[K_{vb}] \quad \text{Eq. 7-209}$$

As previously was done, modes of the v-set are also calculated.

$$(-\lambda[M_{vv}] + [K_{vv}])\{\phi_v\} = \{0\} \quad \text{Eq. 7-210}$$

If residual vectors are requested, they are calculated using the v-set size matrices and appended to the v-set modes.

At this point, the b-set boundary vectors and the v-set modes can describe the motion of the structure.

$$\begin{Bmatrix} u_b \\ u_v \end{Bmatrix} = \begin{bmatrix} I & 0 \\ G_{vb} & \phi_{vi} \end{bmatrix} \begin{Bmatrix} u_b \\ \xi_q \end{Bmatrix} \quad \text{Eq. 7-211}$$

or splitting the v-set into its o-set and w-set partitions,

$$\begin{Bmatrix} u_b \\ u_w \\ u_o \end{Bmatrix} = \begin{bmatrix} I & 0 \\ G_{wb} & \phi_{wq} \\ G_{ob} & \phi_{oq} \end{bmatrix} \begin{Bmatrix} u_b \\ \xi_q \end{Bmatrix} \quad \text{Eq. 7-212}$$

In the above representation, the w-set dofs are not independent dofs, but are dependent on b-set and q-set motion. In order to make the w-set independent, vectors that associate o-set motion with individual w-set dofs must be found. Once these vectors are found, the components are removed from the b-set and q-set motion resulting in w-set vectors that are independent.

The w-set vectors are determined using the flexibility of the v-set.

$$\begin{bmatrix} Y_{oo} & Y_{ow} \\ Y_{wo} & Y_{ww} \end{bmatrix} \begin{Bmatrix} P_o \\ P_w \end{Bmatrix} = \begin{Bmatrix} u_o \\ u_w \end{Bmatrix} \quad \text{Eq. 7-213}$$

If loads are applied only to the w-set, the displacements are dependent only on P_w .

$$\begin{bmatrix} Y_{ow} \\ Y_{ww} \end{bmatrix} \{P_w\} = \begin{Bmatrix} u_o \\ u_w \end{Bmatrix} \quad \text{Eq. 7-214}$$

Vectors that associate o-set motion with individual w-set dofs can be found by setting $P_w = Y_{ww}^{-1}$.

$$\begin{bmatrix} Y_{ow} \\ Y_{ww} \end{bmatrix} [Y_{ww}^{-1}] = \begin{bmatrix} Y_{ow} Y_{ww}^{-1} \\ I_{ww} \end{bmatrix} \quad \text{Eq. 7-215}$$

The above vectors are used to make the w-set an independent set.

$$\begin{Bmatrix} u_b \\ u_w \\ u_o \end{Bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ G_{ob} - Y_{ow} Y_{ww}^{-1} G_{wb} & Y_{ow} Y_{ww}^{-1} \phi_{oq} - Y_{ow} Y_{ww}^{-1} \phi_{wq} \end{bmatrix} \begin{Bmatrix} u_b \\ u_w \\ \xi_q \end{Bmatrix} \quad \text{Eq. 7-216}$$

The flexibility matrices Y are determined from displacements due to unit forces on the w-set.

$$[K_{vv}]^{-1} \begin{bmatrix} 0 \\ I_{ww} \end{bmatrix} = [\bar{Y}_{vv}] \quad \text{Eq. 7-217}$$

The modal flexibility is removed from the displacement response to determine the flexibility matrices used in [Eq. 7-216](#).

$$[Y_{vv}] = [\bar{Y}_{vv}] - [\phi_{vq}]([\phi_{vq}]^T [M_{vv}] [\phi_{vq}])^{-1} [\phi_{vq}]^T [M_{vv}] [\bar{Y}_{vv}] \quad \text{Eq. 7-218}$$

$$[Y_{vv}] = \begin{bmatrix} Y_{ow} \\ Y_{ww} \end{bmatrix} \quad \text{Eq. 7-219}$$

Enhancements to Degrees of Freedom Specification for Residual Vector Processing

Until now, the specification of degrees of freedom where unit loads are to be applied to obtain static solutions for use in residual vector computations required the use of USETi,U6 and SEUSETi,U6 Bulk Data entries. In analysis involving multiple superelements, these entries have the following disadvantages:

- Separate entries are required for each superelement.
- The unit loads on a superelement due to these entries are not passed downstream for residual vector processing by the downstream superelements.

In order to overcome the above drawbacks, two new Bulk Data entries called RVDOF and RVDOF1 have been incorporated into MSC.Nastran 2004.

- Separate entries for separate superelements are no longer required. Instead, these new entries may be used to reference points that are interior to any superelement. The program will automatically partition the data for allocation to the appropriate superelements.
- The unit loads applied to a superelement due to the RVDOF/RVDOF1 entries are passed downstream all the way down to the residual for the purpose of residual vector processing by all superelements in its downstream path. This yields more accurate results.

Static Solution for Shape Functions

If the structure contains rigid body modes it is not stable for static analysis. This problem is solved by one of two methods:

Manual SUPORT (SubDMAP SEKRRS)

If the user supplies SUPORTi entries, the r-set DOFs it produces are used as "temporary" SPCs, applied to allow solving for the static shapes, but removed before computing modes. Partitions involving "r" are discarded.

$$K_{aa} \geq \begin{bmatrix} K_{ll} & K_{lr} \\ K_{rl} & K_{rr} \end{bmatrix} \quad \text{Eq. 7-220}$$

$$P_{a2} \geq \begin{bmatrix} P_{l2} \\ P_{r2} \end{bmatrix} \quad \text{Eq. 7-221}$$

$$K_{ll} \cdot ul \leq P_{l2} \quad \text{Eq. 7-222}$$

$$u_{a2} = \begin{bmatrix} u_{l2} \\ 0_{r2} \end{bmatrix} \quad \text{Eq. 7-223}$$

Auto SUPPORT (SubDMAPs MODEFSRS, SEMR3) ACMS Only

If no SUPPORTi entries or an inadequate set are supplied for constraining all rigid body modes, and rigid body modes are found when solving $[K_{aa} - \lambda \cdot M_{aa}] \cdot \phi_a = 0$, the rigid body modes ϕ_r are separated from the flexible modes based on their natural frequency being smaller than a user parameter, PARAM, FZERO, (default = 1.E-4 Hz). ϕ_r is used to make constraint equations MP_{ar} ,

$$MP_{ar} = M_{aa} \cdot \phi_r \quad \text{Eq. 7-224}$$

These equations are appended to K_{aa} using Lagrange Multiplier Techniques, and solved directly,

$$\begin{bmatrix} K_{aa} & MP_{ar} \\ MP_{ar}^T & O_{rr} \end{bmatrix} \begin{bmatrix} u_{a2} \\ -q_{r2} \end{bmatrix} = \begin{bmatrix} P_{a2} \\ O_{r2} \end{bmatrix} \quad \text{Eq. 7-225}$$

The matrix on the left is named KLL in the subDMAP. It is similar to the KLL produced above by partitioning the larger K_{aa} matrix in that it is stable for static solution, but differs in that it is larger than K_{aa} by the addition of the constraint equations. These equations enforce the condition expressed by the bottom row,

$$MP_{ar}^T \cdot u_{a2} = 0 \quad \text{Eq. 7-226}$$

This is a statement that the c.g. of the structure is at rest on average, but the structure is free to deflect about it. This approach will produce different u_{a2} vectors than those produced by the partitioned solution. In the partitioned solution one set of points is grounded, and all loads are transmitted through the structure to them. A load on an explicit r-set point, for example, will go directly to ground, and will cause zero deflection. This implies that some skill is required in selecting DOFs for SUPPORTi entries. With the LMT approach the constraints to ground are distributed, so that loads applied anywhere will tend to load the entire structure.

Solution for Trial Vectors by Static Analysis

Inertia relief analysis is NOT performed, despite the presence of SUPORTi entries, usually used in the context of inertia relief. In this context, SUPORTi data or its equivalent for the auto-support feature is used for SPC-type operations. In CMS at present, K_{oo} may not contain rigid body modes. There is no provision to constrain them with SUPORT-type inputs. DOFs may be removed from the o-set to make it stable. For example, in superelement analysis, an interior point active in the rigid body mode, which by default is in the o-set, can be made an exterior point. This changes it to the a-set, which is held fixed when computing CMS modes. If the rigid body modes still exist downstream, they can be constrained in residual structure calculations.

The u_{a2} trial vectors, or "raw static shapes", as they are called in the SubDMAP comments, may not be linearly independent of the mode shapes, and are certainly not orthogonal to the mode shapes or each other. An orthogonalization process is now used to insure that the resulting shapes are orthogonal to the mode shapes and each other. Shapes that have little information after this process are discarded.

Orthogonalization

The trial vectors u_{a2} may not be linearly independent of each other or the mode shapes. They are processed to find a set of combined mode shapes and linear combinations of the trial shapes that are all orthogonal to each other with respect to the mass matrix. This reduces the risk of ill-conditioning when using these matrices for solution later.

The trial vectors are appended to ϕ_a to form the combined set ϕ_{ua1}

$$\phi_{ua1} = \left[\phi_a \mid u_{a1} \right] \quad \text{Eq. 7-227}$$

An optional pre-sweeping of trial vectors is done when requested by PARAM, RESVSO, YES. The derivation of the sweeping equation is similar to that for sweeping loads above, with the equation for x , the vector of linear combination coefficients, being

$$x = M_{i-1} \cdot \phi_a \cdot M_{aa} \cdot u_{a2s} \quad \text{Eq. 7-228}$$

The swept displacement shape vectors ϕ_{ua1} are then determined by eliminating x ,

$$\phi_{ua1} = u_{a2s} - \phi_{a2} \cdot M_i^{-1} \cdot \phi_{a2} \cdot M_{aa} \cdot u_{a2s} \quad \text{Eq. 7-229}$$

If the optional sweeping is not selected ϕ_{a1} is merely equal to ϕ_{a2} .

The u_{a1} vectors are mass-normalized into ϕ_{ua} such that the diagonal terms of the product on the left side are unity

$$\phi_{ua} \cdot M_{aa} \cdot \phi_{ua} = M_{zz} \quad \text{Eq. 7-230}$$

$$\text{diag}(M_{zz}) = I_{zz} \quad \text{Eq. 7-231}$$

M_{zz} is generally fully coupled at this point. It is decomposed with an LTL factorization,

$M_{zz} = L \cdot D \cdot L$, where L is lower triangular and D is diagonal.

The ratio of D over the diagonal terms of M_{zz} is calculated in the vector R ,

$$R = \left\{ \frac{\text{diag}(M_{zz})}{\text{diag}(D)} \right\} \quad \text{Eq. 7-232}$$

Rows of R whose value exceeds PARAM, RESVRAT (default=1.e8) indicate columns of ϕ_{ua} that have little new content, and are discarded to produce U_{a3} and M_{zz3} . M_{zz3} is diagonal in the region of the eigenvectors, but coupled in the region of the auxiliary shapes. It is then re-named as M_{zz} .

The generalized stiffness corresponding to M_{zz} is calculated,

$$K_{zz} = u_{a3} \cdot K_{aa} \cdot u_{a3} \quad \text{Eq. 7-233}$$

The eigensolution of the z-set is obtained,

$$[K_{zz} - \lambda_z \cdot M_{zz}] \cdot \phi_z = 0 \quad \text{Eq. 7-234}$$

The eigenvectors ϕ_z of this reduced basis are used as a transformation to orthogonalize U_{a3} ,

$$\phi_a = \phi_{ua} \cdot \phi_z \quad \text{Eq. 7-235}$$

The eigenvalue table of the combined system, λ_z , replace the prior eigenvalue table that contained modes only. λ_z includes both structural modes and the effects of orthogonalized residual flexibility vectors.

Model Partitioning in the MODEFSRS subDMAP

MODEFSRS is used in the residual structure to separate the fluid and structural components of the a-set. The modes of these components are computed independent of one another to find ϕ_{sa} and ϕ_{fa} , the structural and fluid modes, respectively. Once computed, residual flexibility effects are added with calls to the RESVEC SubDMAP.

The eigenvectors with residual flexibility effects appended to them ϕ_a are then merged together to produce one eigenvector matrix,

$$\phi_a = \begin{bmatrix} \phi_{sa} & \mathbf{0} \\ \mathbf{0} & \phi_{fa} \end{bmatrix} \quad \text{Eq. 7-236}$$

For the cases where only fluid or only structural modes are requested, the identity matrix is substituted for the eigenvector matrix that is not calculated.

7.5.3 Complex Eigenvalue Analysis in SubDMAP CEIGRS

Complex eigenvalues and the associated eigenvectors are calculated in CEAD (Complex Eigenvalue Analysis -- Displacement) using the inverse power method, the determinant method, the Hessenberg method, or the Lanczos method, as requested by the user on the EIGC Bulk Data entry. For direct complex eigenvalue analysis and if PARAM,ASING = 0 (default) then null rows and columns are discarded from $[K_{dd}]$, $[M_{dd}]$, and $[B_{dd}]$ to form $[K_{xx}^d]$, $[M_{xx}^d]$, and $[B_{xx}^d]$.

In the case of a direct formulation CEAD extracts the eigenvalues from the following equation.

$$[M_{xx}^d p^2 + B_{xx}^d p + K_{xx}^d] \{u_x^d\} = 0 \quad \text{Eq. 7-237}$$

$\{u_x^d\}$ is then augmented with null rows to form $\{u_d\}$.

In the case of a modal formulation the following equation is used:

$$[M_{hh} p^2 + B_{hh} p + K_{hh}] \{u_h\} = 0 \quad \text{Eq. 7-238}$$

CEAD also normalizes the eigenvectors according to one of the following user requests:

- Unit magnitude of a selected coordinate ("POINT").
- Unit magnitude of the largest component ("MAX").

as specified on the EIGC Bulk Data entry.

The complex Lanczos method is recommended for large problems with sparse matrices. The HESS method is best suited to small dense matrices, typical of small models or those typical in a modal formulation. The INV method is obsolete, and is retained as an alternate method when all else fails.

7.5.4 Frequency Response and Random Analysis in SubDMAP FREQRS

The frequency-dependent loads are generated in FRLG (Frequency Response Load Generation) from the RLOAD1 and RLOAD2 Bulk Data entries. Constraints are applied to reduce the loads to the d-set, and in the case of a modal formulation, the loads are transformed to the h-set.

The solution for the response is performed in FRRD1 or FRRD2 (Frequency Response Solution). In the case of a direct formulation the following equation is solved.

$$[-M_{dd}\omega^2 + iB_{dd}\omega + K_{dd}]\{u_d\} = \{P_d\} \quad \text{Eq. 7-239}$$

and in the case of a modal formulation, the following equation is used in:

$$[-M_{hh}\omega^2 + iB_{hh}\omega + K_{hh}]\{u_h\} = \{P_h\} \quad \text{Eq. 7-240}$$

The arithmetic used in the solution may be real or complex and the solution procedure may be symmetric or unsymmetric. The choice of arithmetic and solution procedure is made by the program depending on the form of the dynamic matrices.

Power spectral density functions and autocorrelation functions are computed in RANDOM (Random Analysis) according to the information on RANDPS and RANDT1 Bulk Data entries. Random analysis calculations are made for selected displacements, loads, single point constraint forces, element stresses and element forces. The selection of items for random analysis is made in the XY-output request part of the Case Control Section.

7.5.5 Transient Response in SubDMAPs SEDTRAN and SEMTRAN

The linear time-dependent loads are generated in TRLG (Transient Response Load Generation) from the TLOAD1 and TLOAD2 Bulk Data entries. The nonlinear loads are generated from the NOLINi Bulk Data entries. Constraints are applied to reduce the loads to the d-set, and in the case of a modal formulation, the loads are transformed to the h-set.

The solution for the response is performed in TRD1 (Transient Response Solution). In the case of a direct formulation the following equation is integrated over the time periods specified via the TSTEP Bulk Data entry in subDMAP DTRANRS:

$$[M_{dd}p^2 + B_{dd}p + K_{dd}^x]\{u_d\} = \{P_d\} + \{P_d^{nl}\} \quad \text{Eq. 7-241}$$

in order to determine the displacement velocity and acceleration response.

In the case of a modal formulation, the following equation is used in subDMAP MTRANRS:

$$[M_{hh}p^2 + B_{hh}p + K_{hh}]\{u_h\} = \{P_h\} + \{P_h^{nl}\} \quad \text{Eq. 7-242}$$

If there are no direct input matrices and no nonlinear loads, an analytical integration procedure is used rather than numerical integration. The numerical integration procedure may use either symmetric or unsymmetric routines. The choice is made by the program depending on the form of the dynamic matrices.

7.5.6 Buckling Analysis in SubDMAP MODERS

In linear buckling, a static load is applied to the structure and the following eigenvalue problem is solved subsequently,

$$[K_{aa} + \lambda_i K_{aa}^d]\{\phi_i\} = 0 \quad \text{Eq. 7-243}$$

where $[K_{aa}]$ is the linear stiffness, $[K_{aa}^d]$ is the differential stiffness, λ_i are the eigenvalues and $\{\phi_i\}$ are the eigenvectors (buckling modes). The critical buckling loads are

$$\{P_i^{crit}\} = \lambda_i \{P_a\} \quad \text{Eq. 7-244}$$

where $\{P_a\}$ is the vector of applied static loads including reaction forces.

The basic assumption in linear buckling is that the differential stiffness $[K_{aa}^d]$ is proportional to the applied load $\{P_a\}$. The assumption implies a linear material law and small deformations. For buckling in nonlinear problems see “[Additional Topics](#)” on page 529.

The linear stiffness $[K_{aa}]$ is derived from the variation of the stresses, the differential stiffness $[K_{aa}^d]$ is derived from the variation of the strains keeping the stresses constant.

SOL 105 is the recommended solution sequence for linear buckling in static analysis. The Case Control Section must contain at least two subcases. A static loading condition is applied in the first subcase. The load is defined with a LOAD,

TEMP(LOAD), or DEFORM Case Control command. The load may also be defined with enforced deformations on the SPC Bulk Data entries. In the second subcase, a METHOD command must appear to solve the eigenvalue problem. The Lanczos method (EIGR Bulk Data entry) is the recommended eigenvalue solver for linear buckling.

Linear buckling is available in the following solution sequences.

SOL 105 Linear buckling in statics with superelements, including original design sensitivity (DSA).

SOL 5 Linear buckling in statics.

The linear buckling capability is available for the following elements: CONROD, CROD, CTUBE, CBAR, CBEAM, CBEND, CQUAD4, CTRIA3, CTRIA6, CSHEAR, CHEXA, CPENTA, AND CTETRA.

The differential stiffness formulation for the CQUAD4 and CTRIA3 elements was changed in Version 68. Numerous spurious modes appeared in linear buckling of thin shell structures using versions prior to Version 68. The improved results of Version 68 can be produced with a version earlier than Version 68 by overlapping two CQUAD4 or CTRIA3 elements, one with bending stiffness only and one with membrane stiffness only. The pre-Version 68 method of calculating the differential stiffness may be activated in Version 68 by setting the system cell NASTRAN SYSTEM(170)=1 in the NASTRAN statement. The Version 68 method is the default, NASTRAN SYSTEM(170)=0. The pre-Version 68 method remains in the code to recover old results.

The following recommendations apply to linear buckling analysis.

- Use at least 5 elements per half sine wave to get reasonable results in the buckling load.
- For buckling of 3-D shell structures, use PARAM,K6ROT,100. to assign a stiffness to the sixth degree-of-freedom. The default is PARAM,K6ROT,0.

The following restrictions apply to linear buckling analysis.

- Offsets should not be used in beam, plate or shell elements. The buckling loads for structures with offsets are incorrect.
- Follower force effects are not included in the differential stiffness. The calculated buckling loads are incorrect. The Bulk Data entries FORCE1, FORCE2, MOMENT1, MOMENT2, PLOAD, PLOAD2, PLOAD4, RFORCE, TEMP, TEMPD, TEMPP1, TEMPP3, TEMPRB describe follower forces.

7.5.7 Superelement Analysis

The theoretical aspects of superelement analysis are discussed in the *MSC.Nastran Handbook for Superelement Analysis*. Superelements are mathematically equivalent to substructures. The major difference is in a more convenient interface for the user and in the ability to process a much broader class of problems than is practical for most other substructuring capabilities.

There are now two distinct paths through superelements in the solution sequences. There are also two major types of superelements. When a BEGIN SUPER entry is present the program uses the more modern SEP1X module to make the SEMAP table used to control partition of superelements. When there is a BEGIN BULK entry but no BEGIN SUPER entries a parallel path using the older SEP1 module is used instead. While the newer module supports most features of both part and grid list superelements, it does not yet have some of the advanced features supported in the older module. When starting a new project you should decide at modeling initiation which path is most suitable because transitioning from one to another can be painful, particularly when the model is in development. If one starts on the old path, for example, and then adds a new feature that causes the program to automatically switch to the new path, some of your modeling input may be invalidated. This may cause considerable, confusing, and unnecessary work to transition to the new path.

The short answer is that the new path should be the best path for most projects, based on the appeal of its advanced features and the likelihood that future improvements in MSC.Nastran may require use of the new path. The new path is forced by replacing the BEGIN BULK entry with a BEGIN SUPER entry in your Main Bulk Data file. It should be the first entry in your bulk data. It allows grid list superelements, but they need not be present.

The exception for the path chosen is when your project requires the use of features available in the old path but not yet enabled in the new path. Some of these features may be used in the new paths by special avoidance techniques, but these techniques can be cumbersome. It may be more convenient to stay with the older methods until the newer methods catch up.

Terminology

- **Newest Option:** The partitioned bulk data file superelements (parts) are the form introduced in Version 69. They are sometimes called substructures. A major feature of parts is ID independence. That is, element IDs and grid points IDs may be duplicated across different parts, although they must be unique within a part.

- **Intermediate:** The Main Bulk Data Superelements (superelements) introduced in Version 47 are sometimes called grid list superelements (list s.e.s). When used in the new path, they have more restrictive rules than in the old path.
- **Oldest Path:** List s.e.s only are used in the old path. There are no formal names at present to differentiate the intermediate and old flavors. Some advanced features are available only in this path, at present.

A subtle difference between the two paths is that the new path will always use a single level configuration for list s.e.s if there is no SETREE-type entry. It does this by defining new boundary points in the residual structure when some are needed but not supplied by the user. The old path automatically converts to a multilevel configuration when required, regardless of the presence of DTI, SETREE entries. It has no ability to generate new boundary points. This type of change when switching from the old path to the new, on purpose or by accident can be confusing.

Criteria for Selecting Part Versus List Superelements

Each of the two major forms has overwhelming advantages in certain contexts. Some of these advantages are of a temporary nature, as the rest of the technology in supporting FE tools catch up to the newest methods. However, other relative advantages are due to the inherent difference in concept between the two major branches that will be relevant into the foreseeable future. It is likely that both branches will be maintained and extended because of their unique capabilities. The following criteria should be considered when making a choice:

- Converting an existing non-superelement model to a part configuration is a labor-intensive process at present. No migration aides are provided. Converting it to a list superelement requires addition of only SESET entries, a trivial task by comparison.
- When debugging a new model it is much easier to comprehend and more efficient when it is in a superelement configuration. Errors in one area will not obscure errors in other areas, and the model solved is smaller, leading to faster turnaround. It may be more economical to run it in a non-superelement configuration for production runs after the model is stable and requires no more changes. If it is a list superelement this means merely removing the SESET entries, after which it is automatically an exactly equivalent non-superelement model if advanced features such as those provided by the RELEASE and CSUPEXT entries are avoided. If it is a part superelement there can be a large labor cost for conversation, as all

duplicated IDs will need to be changed, duplicate points at boundaries will need to be resolved, etc. A list s.e. model, to make virtue of necessity, enforces rules such as unique IDs when updating the model that make re-conversion to a non-s.e. configuration an almost automatic process.

- When starting a new modeling project where more than one person provides input, and most particularly when more than one organization contributes to the model, the higher degree of data control for part superelements is attractive. There is no need to legislate acceptable ID ranges for each part. It is possible to lock the description of a part into its boundary matrices, in a data base marked with a time stamp, so that there is little questions about who is responsible for the behavior of each part.
- Many GUIs can be used to modify list superelements because the rules for ID independence are the same for GUIs as they are for list s.e.s Part superelements can be modeled in GUIs only one at a time. This makes boundary matching, in particular, cumbersome.

Features Not Supported in the New Path at Present

- Fluid Structure Analysis: Starting with MSC.Nastran 2001, this capability is supported in the new path with the “old” path rules:
 - a. All fluid elements must be in a special list superelement, and/or in the residual structure
 - b. This special superelement may have fluid elements only; structural elements or grid points are not allowed
 - c. One or several fluid cavities may be in the fluid superelement
 - d. Fluid elements may be connected to structural elements in list superelements, or in the residual structure.
 - e. Elements in part superelements may not be connected to fluid elements.
- p-Elements: Not supported at all. It may be possible to segregate the portions of the model containing p-elements into external superelements, but this will be a labor-intensive process. No documentation is available on this topic at present.
- RELEASE Entries: Allowed only in the old path. There are semi-heroic modeling techniques that may be used to simulate this feature in the new path, but they are not documented.

- **SExSET Entries:** Most set entries for non-superelement models such as ASET have synonyms for use in list s.e.s such as SEASET. The synonym is not accepted in PARTs, but the simpler format of ASET, for example, is accepted.

Features Not Supported in the Old Path

In general, features associated with automatic boundary control are not supported. The QRG is not explicit on this issue. A list of entries not supported in the old path is:

SETREE, SENQSET, SEBNDRY, SEBULK, SECONCT, SEEXCLD, SELABLE, SELOC, SEMPLN

The DTI, SETREE entry is used in place of SETREE. The action of the SEELT entry is somewhat different. For the new path, the SEMAP will be changed to account for the action of SEELT entries. For the old path, it is assumed that the element being moved is one that can be moved without changing the SEMAP, such as an element attached only to boundary points. If the SEELT requires change of the SEMAP the program processes conventionally until it encounters the s.e. with the “illegal” SEELT entry, then gives a fatal error exit stating that the connectivity of the superelement is incorrect.

Check List for Transitioning an Old Model to the New Path

- Replace the BEGIN BULK entry with a BEGIN SUPER entry.
- Check the SEMAP table from an old run to determine if it is in a multilevel state. When it is, add SETREE entries that define the tree in the old run, or convert it to a single level model if multilevel is not essential. If you have many multilevel models to convert, there is a parameter that will request that SETREE entries be made from the SEMAP, and placed in the.pch (punch) file.
- Discard all RELEASE and CSUPEXT entries and p-elements. If you cannot find reasonable replacements for them you may be forced to stay in the old path.

Decide if this is the right time to convert to a PART model, or whether the list s.e. capability in the new path is more advantageous, based on the discussion above.

Solution sequences that support superelement analysis are listed in “**Output Description**” on page 347.

The Case Control commands for superelement processing selection all begin with the letters SE (SEMG, SELG, SEKR, SEMR, SELR, SEALL, and SEEX). The SEFINAL command is used to control the order of processing of the superelements. A subcase

structure is required that defines constraints, loads, and data recovery options for each superelement. Each subcase includes the Case Control command SUPER in order to relate the subcase to a particular superelement and load condition. Examples of Case Control (including plot requests) are given in the *MSC.Nastran Handbook for Superelement Analysis*. A set of plotting commands is required for each superelement to be plotted, separated by the plotter commands SEPLOT and SEUPPLOT.

Each superelement is given an identification number (SEID). The user assigns each grid point to a superelement on a GRID or SESET Bulk Data entry. Identical image, mirror image, and external superelements are defined by the CSUPER and optional SEQSEP Bulk Data entries. The SEELT Bulk Data entry may be used to change the superelement membership of boundary elements. The RELEASE Bulk Data entry may be used to disconnect selected degrees-of-freedom from exterior grid points of a superelement. “[Use of Parameters](#)” on page 35 lists all PARAMeters used in superelement analysis.

The mesh generator option, called by MESH on the NASTRAN statement, supports the superelement capability fully, in that all GRID entries generated for a field can be given the same SEID, and the SESET Bulk Data entry can change these SEIDs at boundaries. The automatic grid point resequencing features are fully implemented for superelement models.

The superelement partitioning module passes on resequencing data from SEQGP Bulk Data entries only for grid points of a superelement which are not exterior points for any superelement. This class of grid points can be determined from the SEMAP table printed by the program, but may change if the order of processing is changed. The automatic resequencing option processes each superelement separately.

Component mode synthesis uses SEBSETi, SECSETi, SESUP and SEQSETi entries to define the boundary condition when computing component modes. Uncoupled solutions for statics and modes can be obtained by use of PARAM,FIXEDB,-1.

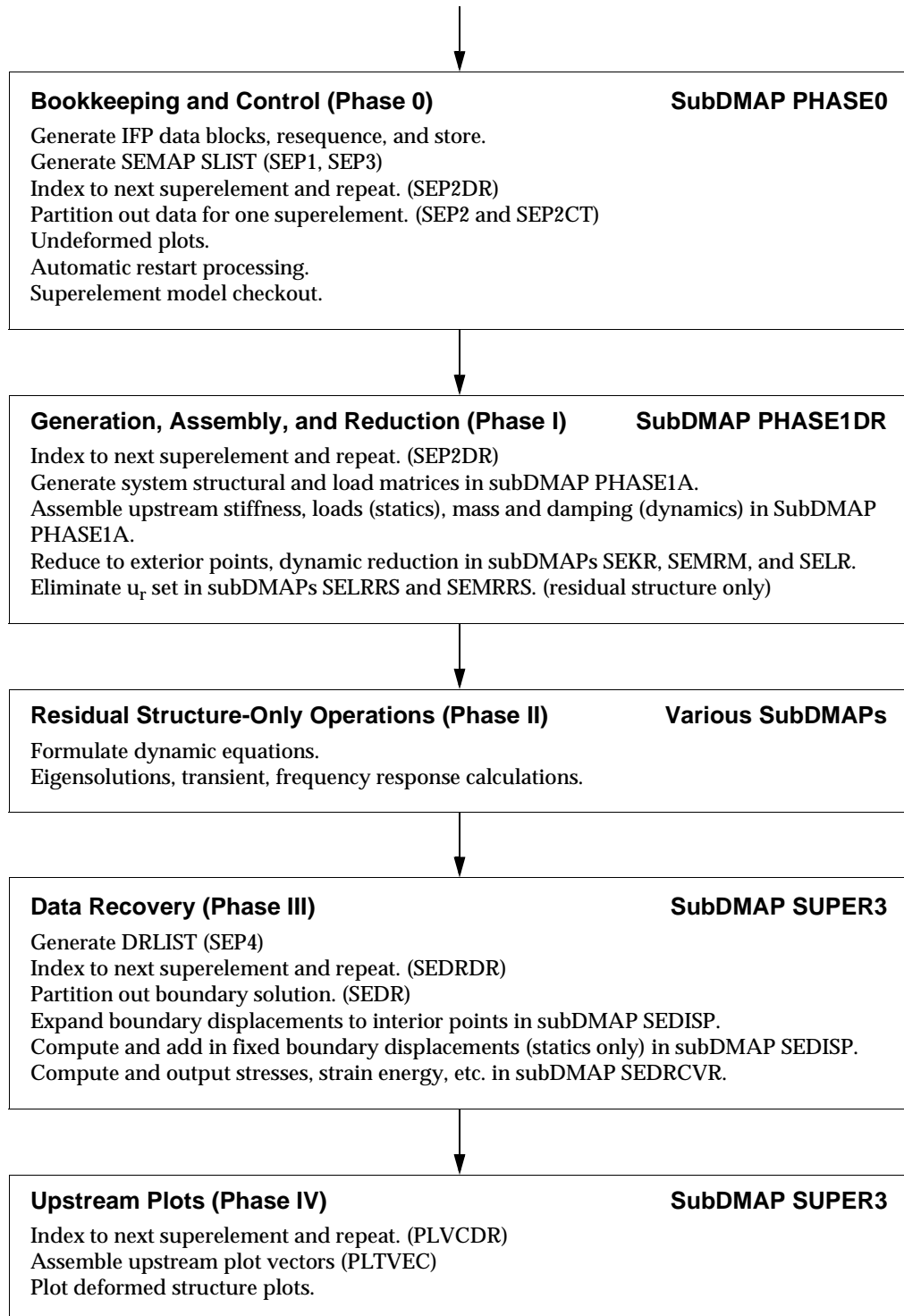
Superelement Bulk Data entries may be specified in nonsuperelement solution sequences, but those entries pertaining to the superelement capability (e.g., the CSUPER entry) are ignored, as well as some fields on other entries (e.g., Field 9 on the GRID entry). However, better automatic resequencing may be possible if the SEID data is removed.

Superelement Processing in the Structured Solution Sequences

An overview of the superelement processing in SOLutions 101-200 is shown in [Figure 7-3](#). Phase 0 performs the preparation and bookkeeping function. The input data is stored on the database and scanned to prepare the SEMAP table. This table lists

the superelement to which each Bulk Data entry is to be applied. Phase 0 also partitions out the data that defines each superelement and performs automatic restart processing. In Phase I this data is used to generate the boundary matrices to be assembled when combining the superelements into the total structure. Phase II performs calculations done only on the residual structure (the superelement processed last), such as transient response calculations or eigensolutions. Phase III expands boundary solutions to the interior grid points of the superelement, performs the data recovery operations for internal forces, stresses, and strain energy, and outputs these quantities. Phase IV combines plot vectors from upstream superelements to provide deformed structural plots for all or part of the structure.

SOLs 101 through 200
Bulk Data Entries, Case Control Commands



(Superelement module names are in parentheses)

Figure 7-3 Structured Solution Sequence Flow Chart

The input file processor operations are carried out in a conventional manner. The grid point resequencer module resequences each superelement individually.

Some of the geometry processor operations of “**Geometry Processing in SubDMAP PHASE0**” on page 374 are carried out to prepare tables of all grid points in the entire model, useful for generation of the SEMAP table. After the SEMAP is available, the SLIST table is assembled. This list controls the operations to be performed in that particular run for generating boundary matrices, as selected with Case Control commands. The GP4 module generates constraint data in the form of the USET table, as described in “**Geometry Processing in SubDMAP PHASE0**” on page 374. A separate USET table is generated for each superelement. It is similar to the USET table of the older rigid formats except that the exterior points are automatically placed in the a-set.

Assembly and Reduction (Phase I) in SubDMAP Phase1DR

The following operations may be carried out, as selected by the user with superelement-type Case Control commands. By default, all superelements and the residual structure are given these operations serially. If requests for the SEMA (or SEMR) operation are made on an upstream superelement and the residual structure, all superelements in the tree between the selected superelement and the residual structure will also be given the SEMA operation.

Superelement Matrix Generation (SEMG) in SubDMAP PhaseIA

The element structural matrices (stiffness, mass and damping) are generated and stored. The direct matrix input option (DMIG Bulk Data entry) can be used to add stiffness, mass or damping terms to the system matrices in this operation. These terms are selected by the Case Control commands K2GG, M2GG, and B2GG. The parameters CB1, CB2, CK1, CK2, CM1 and CM2 can be used to scale the structural matrices.

The SEMA module adds in stiffness terms from any upstream superelements. The SEEX Case Control command may be used to eliminate stiffness contributions from selected upstream superelements. The superelements that are upstream are defined by the current SEMAP. The number of rows and columns of the upstream superelements must be consistent with the number of exterior grid points, as defined by the SEMAP. In order to complete execution of the SEMA module, the boundary matrices of all upstream superelements must be present in the database, and consistent with the current SEMAP.

The SELA module adds loads from upstream superelements to the load vector for the current superelement. The SEEX Case Control command may be used to eliminate load contributions from upstream superelements. The requirements for conformity of load data in the database with the current state of the SEMAP are similar to those for

stiffness, except that only rows, not columns of the load vectors are mapped using data in the SEMAP table. The requirement on columns of all load matrices is that they be equal to the number of columns (number of load condition) in the residual structure. If the number of columns in all superelements and the residual structure is not the same, a user fatal error results. Static loads may be applied to any grid point, whether exterior or interior to a superelement. Dynamic loads may be applied directly to residual structure points. Dynamic load entries may also reference static load sets applied to the superelements via the LOADSET Case Control command and LSEO Bulk Data entries.

Superelement Load Generation (SELG) in SubDMAP PHASE1A

The load generation module SSG1 is executed in the static and dynamic response solutions. This operation is not executed in the normal modes and complex eigenvalue solution sequences. As loads may depend on structural matrices (i.e., thermal or inertial loads), an SELG operation must be preceded by an SEMG operation. The resultants of these loads about the grid point listed on the PARAM,GRDPNT Bulk Data entry are output.

Superelement Stiffness Reduction (SEKR) in SubDMAPs SEKR and SEKRRS

The user may change SPCs and/or MPCs between subcases (“boundary condition changes”). The GPSP module identifies grid point singularities, and at user's option provides SPCs for singular degrees-of-freedom. This is followed by the constraint eliminations described in “[Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB](#)” on page 379, “[Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB](#)” on page 382, and “[Constraint and Mechanism Problem Identification in SubDMAP SEKR](#)” on page 383. If the omitted degrees-of-freedom contain mechanism-type singularities, the grid and component number of all downstream ends of the mechanism are output in the matrix named MECHS, and a user fatal error stops execution. The approximate fixed-boundary natural frequencies of the superelement are printed in the FAPPROX matrix. The boundary stiffness matrix $[K_{aa}]$ is stored in the database. Mass and damping matrices are not processed in this phase.

SUPPORTi Bulk Data entries, which define the r-set, may reference only grid points in the residual structure. The rigid body operations for the stiffness are performed as described in “[Static Solutions in SubDMAP SEKRRS](#)” on page 391.

Superelement Load Reduction (SELR) in SubDMAP SELR

This operation processes loads in the static and dynamic response solution sequences. This operation is not executed in the normal modes and complex eigenvalue solution sequences. The static load reduction of “[Static Condensation in SubDMAPs SEKR and SEMR2](#)” on page 388 is applied, stopping after generation of the $[P_a]$ matrix for superelements and $[P_r]$ for the residual structure.

Superelement Mass Assembly and Reduction (SEMR) in SubDMAP SEMRM

In the dynamics solution sequences, the constraint operations of “[Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB](#)” on page 379, “[Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB](#)” on page 382, and “[Constraint and Mechanism Problem Identification in SubDMAP SEKR](#)” on page 383 are applied to the mass and damping matrices. The rigid body operations of “[Static Solutions in SubDMAP SEKRRS](#)” on page 391 are applied to the mass of the residual structure only. Dynamic reduction and component mode calculations are performed. This allows a mass-change-only type of restart by use of the SEMG and SEMR options and avoids the costly reduction of the stiffness matrix in the SEKR operation if no stiffness changes are made.

The optional Bulk Data entry PARAM,ERROR,0 affects operations in Phase I. If it is not used, the default action is to terminate the run after the first user fatal error, as in the older rigid formats. If the parameter is used, a user fatal error causes a branch to the end of Phase I for the superelement, and continuation to the generation, assembly, and reduction operations for any remaining superelements. The five SE-type commands may be executed on one run, or may be done separately. The following sequence must be observed if these commands are executed separately:

SEKR after SEMG

SELG after SEMG

SELR after SEKR, SELG, SEMR

SEMR after SEKR

Residual Structure-Only (Phase II)

The operations on the superelements described above are also performed on the residual structure. This is followed by operations unique to the residual structure. Eigensolutions and direct and modal forced response calculations take place at this point. Direct input matrices, when selected by the Case Control commands M2PP, K2PP, and B2PP are constrained and added as in the “[Formulation of Dynamic Equations in SubDMAP GMA](#)” on page 403.

Bulk Data entries used for static loads (i.e., FORCE, GRAV, LOAD, etc.) may be used in direct and modal dynamic analysis. They are interchangeable with DAREA Bulk Data entries and may be referenced by LSEQ Bulk Data entries.

Data Recovery Operations (Phase III) in SubDMAP SUPER3

The total solution vectors of the downstream superelement are partitioned to generate the boundary solution vectors for the superelement being processed. The boundary solution is expanded to all interior points of the superelement. The fixed boundary component is computed and added.

If SPC force output is requested, the resultants of the SPC forces about PARAM,GRDPNT or the basic coordinate system origin are also output. The data recovery operations of “[Static Solutions in SubDMAP SEKRRS](#)” on page 391 are carried out. (Note that solution sequences involving complex eigenvalue or frequency response analysis may produce complex displacement vectors. GPFDR module output (grid point force balance, element strain energy) may be requested for complex vectors, but only the real component of the vector is processed.) Deformed structure plots of only elements in the superelement are generated here.

Upstream Plots (Phase IV) in SubDMAP SUPER3

The displacement vectors of a superelement and all of its upstream members are combined and plotted at the user's request.

Superelement Processing

An overview of the superelement processing is shown in [Figure 7-4](#). Phase 0 performs the preparation and bookkeeping function. The input data is stored on the database and scanned to prepare the SEMAP table. This table lists the superelement to which each Bulk Data entry is to be applied. Phase I partitions out the data that defines one superelement and processes this data to generate the boundary matrices to be assembled when combining the superelements into the total structure. Phase II performs calculations done only on the residual structure (the superelement processed last), such as transient response calculations or eigensolutions. Phase III expands boundary solutions to the interior grid points of the superelement, performs the data recovery operations for internal forces, stresses, and strain energy, and outputs these quantities. Phase IV combines plot vectors from upstream superelements to provide deformed structural plots for all or part of the structure.

The input file processor operations are carried out in a conventional manner. The grid point resequencer module resequences each superelement individually.

Some of the geometry processor operations of “**Geometry Processing in SubDMAP PHASE0**” on page 374 are carried out to prepare tables of all grid points in the entire model, useful for generation of the SEMAP table. After the SEMAP is available, the SLIST table is assembled.

This list controls the operations to be performed in that particular run for generating boundary matrices as selected with Case Control options.

Assembly and Reduction (Phase I)

The following operations may be carried out as selected by the user on superelement-type Case Control commands. By default, all superelements and the residual structure are given these operations serially. If requests for the SEMA (or SEMR) operation are made on an upstream superelement and the residual structure, all superelements in the tree between the selected superelement and the residual structure will also be given the SEMA operation.

Superelement Matrix Generation (SEMG)

The element structural matrices (stiffness, mass and damping) are generated and stored. The direct matrix input option (DMIG Bulk Data entry) can be used to add stiffness, mass or damping terms to the system matrices in this operation. These terms are selected by the Case Control commands K2GG, M2GG, and B2GG. The parameters CB1, CB2, CK1, CK2, CM1, and CM2 can be used to scale the structural matrices.

Superelement Load Generation (SELG)

The load generation module SSGI is executed in the statics and dynamic response solutions. This operation is not executed in the normal modes and complex eigenvalue solution sequences. As loads may depend on structural matrices (i.e., thermal or inertial loads), an SELG operation must be preceded by an SEMG operation. The resultants of these loads about the grid point listed on the PARAM,GRDPNT Bulk Data entry are output.

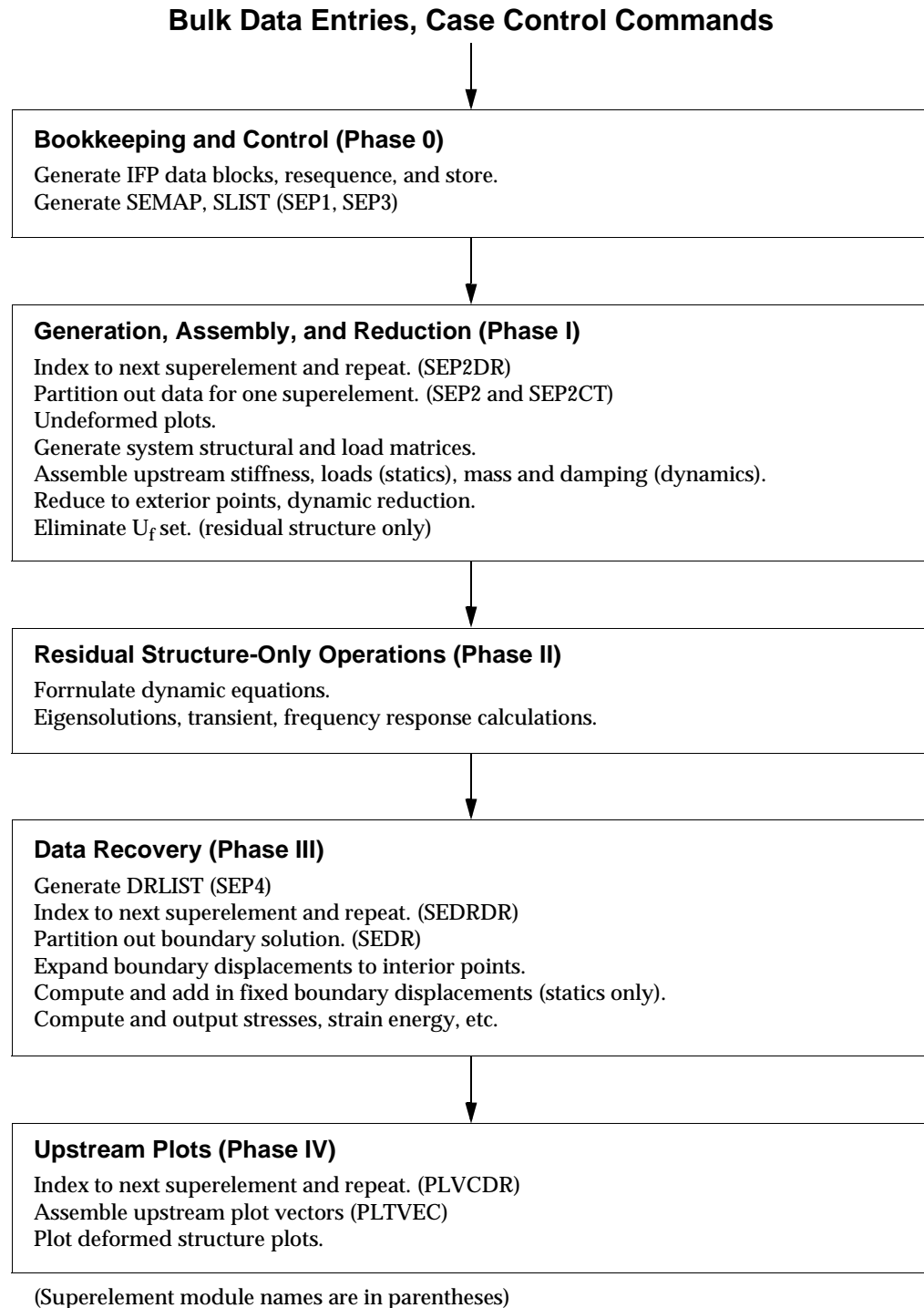


Figure 7-4 Solution Sequence Flow Chart

Superelement Stiffness Matrix Assembly and Reduction (SEKR)

The SEMA module adds in stiffness terms from any upstream superelements. The SEEX Case Control command may be used to eliminate stiffness contributions from selected upstream superelements. The superelements that are upstream are defined

by the current SEMAP. The number of rows and columns of the upstream superelements must be consistent with the number of exterior grid points, as defined by the SEMAP. In order to complete execution of the SEMA module, the boundary matrices of all upstream superelements must be present in the database, and consistent with the current SEMAP. The GP4 module generates constraint data in the form of the USET table, as described in “[Geometry Processing in SubDMAP PHASE0](#)” on page 374. A separate USET table is generated for each superelement. It is similar to the USET table of the older rigid formats except that the exterior points are automatically placed in the a-set.

The user may change SPCs and/or MPCs between subcases (“boundary condition changes”) in SOLs 101 and 103 only. The GPSP module identifies grid point singularities, and at user’s option provides SPCs for singular degrees-of-freedom. This is followed by the constraint eliminations described in “[Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB](#)” on page 379, “[Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB](#)” on page 382, and “[Constraint and Mechanism Problem Identification in SubDMAP SEKR](#)” on page 383. If the omitted degrees-of-freedom contain mechanism-type singularities, the grid and component number of all downstream ends of the mechanism are output in the matrix named MECHS, and a user fatal error stops execution. The approximate fixed-boundary natural frequencies of the superelement are printed in the FAPPROX matrix. The boundary stiffness matrix [KAA] is stored in the database. Mass and damping matrices are not processed in this phase.

SUPPORT Bulk Data entries, which define the r-set, may reference only grid points in the residual structure. The rigid body operations for the stiffness are performed as described in “[Static Solutions in SubDMAP SEKRRS](#)” on page 391.

Superelement Load Assembly and Reduction (SELR)

This operation processes loads in the static and dynamic response solution sequences. This operation is not executed in the normal modes and complex eigenvalue solution sequences. The SELA module adds loads from upstream superelements to the load vector for the current superelement. The SEEX Case Control command may be used to eliminate load contributions from upstream superelements. The requirements for conformity of load data in the database with the current state of the SEMAP are similar to those of the SEKR operation above, except that only rows, not columns of the load vectors are mapped using data in the SEMAP table. The requirement on columns of all load matrices is that they be equal to the number of columns (number of load condition) in the residual structure. If the number of columns in all superelements and the residual structure is not the same, a user fatal error results. Static loads may

be applied to any grid point, whether exterior or interior to a superelement. Dynamic loads may be applied directly to residual structure points. Dynamic load entries may also reference static load sets applied to the superelements via the LOADSET Case Control command and LSEQ Bulk Data entries. The static load reduction of “[Static Condensation in SubDMAPs SEKR and SEMR2](#)” on page 388 is applied, stopping after generation of the $[P_a]$ matrix for superelements and $[P_r]$ for the residual structure.

Superelement Mass Assembly and Reduction (SEMR)

In the dynamics solution sequences, the constraint operations of “[Multipoint Constraint Operations in SubDMAPs SEKR and SEMR2 and SEMRB](#)” on page 379, “[Single Point Constraint Operations in SubDMAPs SEKR, SEMR2 and SEMRB](#)” on page 382 and “[Constraint and Mechanism Problem Identification in SubDMAP SEKR](#)” on page 383 are applied to the mass and damping matrices. The rigid body operations of “[Static Solutions in SubDMAP SEKR](#)” on page 391 are applied to the mass of the residual structure only. Dynamic reduction and component mode calculations of “[Static and Dynamic Load Generation](#)” on page 395 are performed. This allows a mass-change-only type of restart by use of the SEMG and SEMR options and avoids the costly reduction of the stiffness matrix in the SEKR operation if no stiffness changes are made.

The optional Bulk Data entry PARAM,ERROR,0 affects operations in Phase I. If it is not used, the default action is to terminate the run after the first user fatal error, as in the older rigid formats. If the parameter is used, a user fatal error causes a branch to the end of Phase I for the superelement, and continuation to the generation, assembly, and reduction operations for any remaining superelements.

The five SE-type commands may be executed in one run, or may be done separately. The following sequence must be observed if these commands are executed separately:

SEKR after SEMG
 SELG after SEMG
 SELR after SEKR, SELG, SEMR
 SEMR after SFKR

Residual Structure-Only (Phase II)

The operations on the superelements described above are also performed on the residual structure. This is followed by operations unique to the residual structure. Eigensolutions and direct and modal forced response calculations take place at this point. Direct input matrices, when selected by the Case Control commands M2PP, K2PP, and B2PP are constrained and added as in “[Formulation of Dynamic Equations in SubDMAP GMA](#)” on page 403.

Bulk Data entries used for static loads (i.e., FORCE, GRAV, LOAD, etc.) may be used in direct and modal dynamic analysis. They are interchangeable with DAREA Bulk Data entries and may be referenced by LSEQ Bulk Data entries.

Data Recovery Operations (Phase III)

The total solution vectors of the downstream superelement are partitioned to generate the boundary solution vectors for the superelement being processed. The boundary solution is expanded to all interior points of the superelement. The fixed boundary component is computed and added.

If SPC force output is requested, the resultants of the SPC forces about PARAM,GRDPNT or the basic coordinate system origin are also output. The data recovery operations of “[Data Recovery Operations in SubDMAP SEDISP](#)” on page 407 are carried out. (Note that solution sequences involving complex eigenvalue or frequency response analysis may produce complex displacement vectors. GPFDR module output (grid point force balance, element strain energy) may be requested for complex vectors, but only the real component of the vector is processed.) Deformed structure plots of only elements in the superelement are generated here.

Upstream Plots (Phase IV)

The displacement vectors of a superelement and all of its upstream members are combined and plotted at user's request.

Superelement Alternate Statics Solution

SOL 101 provides an alternate method of performing the superelement static condensation. PARAM,ALTRED,YES must be entered in the Bulk Data Section. In some cases, the alternate solution can result in significant reduction in computer cost for static analysis.

The stiffness and load reduction and displacement recovery are described in “[Static and Dynamic Load Generation](#)” on page 395, “[Static Condensation in SubDMAPs SEKR and SEMR2](#)” on page 388 and “[Data Recovery Operations in SubDMAP SEDISP](#)” on page 407.

Static Superelement Analysis With Inertia Relief

Inertia relief analysis is available in SOL 101. PARAM,INREL,-1 or PARAM,INREL,-2 must be present for either of these options. The following method transforms static loads based on model geometry rather than the stiffness matrix as described in “[Static Solutions in SubDMAP SEKRRS](#)” on page 391.

The additional operations required to perform inertia relief are described in “[Static and Dynamic Load Generation](#)” on page 395 and “[Static Solutions in SubDMAP SEKRRS](#)” on page 391.

Assumptions and Limitations

1. The model must have six and only six rigid body modes for a continuous, three-dimensional structure.
2. Masses on scalar points are ignored in calculating inertia loads.
3. The structural model, when given enforced displacements at the reference point, is assumed to produce the same motion as a rigid body model. This will be true for any structural or rigid element. It is possible, if unconventional, to use CELASi or general elements, MPC equations, or DMIG coefficients that will not meet this assumption.
4. Identical and mirror image superelements are not processed correctly.

7.5.8 Nonlinear Heat Transfer Analysis

Steady State Heat Transfer Analysis

The steady state heat balance equation is given by

$$[K]\{u\} + [\mathfrak{R}]\{u + T_{abs}\}^4 = \{P\} + \{N\} \quad \text{Eq. 7-245}$$

where:

$[K]$ = a heat conduction matrix

$[\mathfrak{R}]$ = a radiation exchange matrix

$\{P\}$ = a vector of applied heat flows that are constant

$\{N\}$ = a vector of nonlinear heat flows that depend on temperature

$\{u\}$ = a vector of grid point temperature

T_{abs} = the absolute temperature

The components of the applied heat flow vector, $\{P\}$, are associated either with surface heat transfer or with heat generated inside the volume heat conduction elements. The vector of nonlinear heat flows, $\{N\}$, results from boundary radiation, surface convection, and temperature dependent thermal loads.

The equilibrium equation is solved by a Newton iteration scheme, where the tangential stiffness matrix is approximated by

$$[K_T]^i \approx [K]^i + 4[\mathfrak{R}]^i \{u^i + T_{abs}\}^3 - \left\{ \frac{\partial N}{\partial u} \right\}^i \quad \text{Eq. 7-246}$$

and the residual vector is

$$\{R\}^i = \{P\} + \{N\}^i - [K]^i \{u\}^i - [\mathfrak{R}]^i \{u^i + T_{abs}\}^4 \quad \text{Eq. 7-247}$$

Transient Heat Transfer Analysis

The general equation solved in transient analysis has the form

$$[B]\{\dot{u}\} + [K]\{u\} + [\mathfrak{R}]\{u + T_{abs}\}^4 = \{P\} + \{N\} \quad \text{Eq. 7-248}$$

To take phase change into consideration, the heat diffusion equation is converted into

$$\{\dot{H}\} + [K]\{u\} + [\mathfrak{R}]\{u + T_{abs}\}^4 = \{P\} + \{N\} \quad \text{Eq. 7-249}$$

where:

- $[B]$ = heat capacity matrix
- $[K]$ = heat conduction matrix
- $[\mathfrak{R}]$ = radiation matrix
- $\{P\}$ = vector of applied heat flows that are constant or functions of time
- $\{N\}$ = vector of nonlinear heat flows that depend on temperature
- $\{H\}$ = enthalpy vector
- $\{\dot{H}\}$ = $\{dH/dt\}$
- $\{u\}$ = grid point temperatures
- $\{\dot{u}\}$ = $\{du/dt\}$
- T_{abs} = absolute temperature scale factor

The equilibrium equation is solved by Newmark's method with adaptive time stepping. Base on this one-step integration scheme, the time derivative of the nodal temperatures at the $(i + 1)$ th iteration of the time step $(n + 1)$ is expressed as

$$\{\dot{u}_{n+1}\}^{i+1} = \frac{1}{\theta \Delta t} \{u_{n+1}^{i+1} - u_n\} + \left(1 - \frac{1}{\theta}\right) \{\dot{u}_n\} \quad \text{Eq. 7-250}$$

where

$$\{u_{n+1}\}^{i+1} = \{u_{n+1}\}^i + \{\Delta u_{n+1}\}^i \quad \text{Eq. 7-251}$$

and

$$\frac{1}{\theta} = 2 - 2\eta \quad \text{Eq. 7-252}$$

The parameter η is specified on the PARAM,NDAMP Bulk Data entry. When $\eta = 0$ ($\theta = 0.5$), no numerical damping is requested. In this case, Newmark's method is equivalent to the Crank-Nicolson method.

CHAPTER

8

Element Data Recovery Resolved at Grid Points

- Introduction
- Stress Recovery at Grid Points
- Mesh Stress Discontinuities at Grid Points

8.1 Introduction

Element stresses may be averaged at grid points and then used to calculate estimates of stress discontinuities in the model mesh. Resolving element data at the grid points is supported only for selected linear elastic element types (see [Table 8-1](#) for supported element types). The calculation of average stresses at grid points and the calculation of estimates of stress discontinuities use a common mesh definition method. The mesh is defined in the OUTPUT(POST) section of the Case Control Section. There are two methods available for mesh definition: SURFACE input for 2-D shell element meshes and VOLUME input for 3-D solid element meshes. The averaging of grid point stresses (and strains) and stress discontinuities are calculated independently for meshes composed of only 2-D shell elements and those composed of only 3-D solid elements. The element stress and strain data used in the calculation of grid point stresses and stress discontinuities must be requested in the Case Control Section using the STRESS command.

The request for grid point stresses for reports is made by the GPSTRESS command in Case Control. The calculation of strains may be substituted for stresses using the STRAIN command in place of the STRESS command in the Case Control Section (see “[STRAIN](#)” on page 428 of the *MSC.Nastran Quick Reference Guide* command for element strain output requests).

The calculation of stress discontinuities is supported for both elements and grid points associated with an element mesh by the ELSDCON and GPSDCON commands, respectively.

“[Stress Recovery at Grid Points](#)” on page 466 and “[Mesh Stress Discontinuities at Grid Points](#)” on page 478 explain in more detail the requesting and calculation of grid point stresses and stress discontinuities, respectively.

Table 8-1 Elements Types Supported in Element Data Resolved at Grid Point Calculations (Hyperelastic Elements Not Included)

Element Type	2-D Shell Stress	3-D Solid Stress	Stress Discontinuity
CQUAD4	x		x
CQUADR	x		x
CQUAD8	x		x
CTRIA3	x		x
CTRIAR	x		x
CTRIA6	x		x
CHEXA		x	x
CPENTA		x	x
CTETRA		x	x

8.2 Stress Recovery at Grid Points

Stresses may be computed at grid points on surfaces of two-dimensional plate elements, namely, the CQUAD4, CQUADR, CQUAD8, CTRIA3, CTRIAR, and CTRIA6, and in volumes containing CHEXA, CPENTA, and CTETRA solid elements. By default, stresses are output at the center of each of these elements and also at the vertices of CQUADR, CQUAD8, CTRIAR, and CTRIA6 shell elements and the CHEXA, CPENTA, and CTETRA solid elements. The center output is not universally satisfactory because in the case concerning the surfaces of CQUAD4 and/or CTRIA3 elements, the analyst often has a requirement for realistic stress values at the grid points. Also, in the case of surfaces of CQUADR, CQUAD8, CTRIAR, and CTRIA6 elements or of volumes containing CHEXA, CPENTA, and CTETRA elements, the vertex stresses output for the elements that connect to a common grid point are often not identical. The option described in this section provides a rational method that yields accurate and unique stresses at each vertex grid point in user-defined surfaces or volumes. Grid point stress recovery is only available in linear static analysis, real eigenvalue analysis, and transient analysis. Grid point stress recovery is not available for lamina stresses.

An option is also provided to estimate the probable error in these grid point stresses (see “[Mesh Stress Discontinuities at Grid Points](#)” on page 478). Estimates are generated for each stress component. The root mean square error of these estimates is then computed to provide a single measure of the error in the computation of the stresses at a grid point. This data provides a gross indication of the adequacy of a finite element model to represent the physical phenomena under investigation. The error estimator is available in SOLution Sequence 101.

8.2.1 Description of Method

A scheme to interpolate and/or extrapolate over a surface from a known set of stresses (the element stresses) requires that these known stress components be transformed into a consistent coordinate system prior to interpolation and extrapolation. This coordinate system is defined by the analyst and it should be generally compatible with the surface of interest in order to obtain accurate grid point stresses. Generally, such a coordinate system will be the natural coordinate system that the analyst used in the generation of the model of the surface or volume in question. Calculated grid point stress components are output in this coordinate system, called the output coordinate system of the surface or volume.

Each stress component is treated independently of other stress components in the calculation of average grid point stress components. The stress invariants at grid points (i.e., principal stresses, the vonMises stress, and the mean pressure) are

evaluated from the average values of the stress components at grid points. The grid point stress components are output in the output coordinate system of the surface or volume and the stress invariants are oriented relative to this output coordinate system.

Two methods are used to calculate grid point stress components for plate and shell elements: topological and geometric. A single, simplified, topological method is used to calculate grid point stress components in volumes containing solid elements. However, prior to a discussion of these methods, it is necessary to define the four categories into which grid points are divided.

1. Interior Grid Points

Grid points that are connected only to interior line segments; i.e., line segments that are coincident with the edges of two or more elements.

2. Corner Grid Points

Grid points that are only connected to a single element.

3. Edge Grid Points

Grid points that are connected to both interior and exterior line segments. An exterior line segment is coincident with the edge of only one element.

4. Exception Grid Points

Grid points at which discontinuities in stress can occur. Exception points will be described more completely under “[Treatment of Exception Points -- “BRANCHing”](#)” on page 472.

Stresses at interior, corner, or edge grid points are considered to be continuous between directly connected elements. Stresses at exception points may be discontinuous between connected elements and, as will be seen, a different grid point stress may be output for each of the connected elements. Engineering judgment must be exercised in assessing the validity of grid point stresses at exception points.

8.2.2 Topological Method -- Plate Elements and Solid Elements

The topological method for the evaluation of grid point stresses only recognizes the existence of interior, edge, and corner points; i.e., exception points are not considered as a distinct category of grid point (exception points arise as a result of geometrical rather than topological characteristics). The relations used to compute average grid point stresses in surfaces of plate elements by the topological method are described in [Table 8-2](#). The relations used to compute the average grid point stresses in volumes containing solid elements are described in [Table 8-4](#). Only the topological method is available for solid elements.

Table 8-2 Evaluation of Average Grid Point Stresses for Plate and Shell Elements by the Topological Method

Type of Grid Point	Average Grid Point Stress Component	Remarks
Interior	$\sigma_I = \frac{1}{N_e} \sum_{e_i=1}^{N_e} \sigma_{ei}$ <p>where:</p> <p>σ_I = average stress at interior grid point</p> <p>N_e = number of directly connected elements</p> <p>$\sigma_{ei} = \begin{cases} \text{element center stresses if only CQUAD4} \\ \text{and/or CTRIA3 elements are connected} \\ \text{element vertex stresses if CQUAD8 and/or} \\ \text{CTRIA6 are connected} \end{cases}$</p>	<p>If CQUAD4 and/or CTRIA3 elements connect to the same grid point as CQUAD8 and/or CTRIA6 elements, then only the CQUAD8 and or CTRIA6 elements are used in the calculation of average grid point stresses.</p>
Edge	$\sigma_E = \frac{2}{N_e} \sum_{e_i=1}^{N_e} \sigma_{ei} - \frac{1}{N_I} \sum_{I_i=1}^{N_I} \sigma_{Ii}$ <p>where:</p> <p>σ_E = average stress at edge grid point</p> <p>N_I = number of interior points connected by line segments</p> <p>σ_{Ii} = average stress at interior grid points</p>	<p>If an edge point is not connected to an interior point by line segments, then $N_I = 0$, the second term in the relation σ_E is set to zero, and the factor 2 in the first term is replaced by 1.</p> <p>Same as remark for interior grid points.</p>

Table 8-2 Evaluation of Average Grid Point Stresses for Plate and Shell Elements by the Topological Method (continued)

Type of Grid Point	Average Grid Point Stress Component	Remarks
Corner	<p>Corner points connected to CQUAD4 element with the grid point diagonally opposite the corner point, an interior point, or an edge point.</p> $\sigma_c = 2\sigma_1 - \sigma_D$ <p>where:</p> <p>σ_c = average stress at corner grid point</p> <p>σ_i = CQUAD4 center stress</p> <p>σ_D = average stress at grid point diagonally opposite corner point</p> <p>(If point diagonally opposite corner point is not an interior or edge point, then $\sigma_c = \sigma_1$.)</p>	Same as remark for interior points
Corner	<p>Corner points connected to CTRIA3 elements and the other two grid points, namely, E1 and E2, of the connected CTRIA3 are edge points:</p> $\sigma_c = 3\sigma_1 - (\sigma_{E1} + \sigma_{E2})$ <p>where:</p> <p>σ_c = average stress at corner grid point</p> <p>σ_1 = CTRIA3 center stress</p> <p>σ_{E1}, σ_{E2} = stresses at edge points E1 and E2</p> <p>(if either E1 or E2 is not an edge point, then $\sigma_c = \sigma_1$)</p> <p>Corner points connected to one CQUAD4 or one CTRIA6 element:</p> $\sigma_c = \sigma_r$ <p>where:</p> <p>σ_r = stress at element vertex connected to corner point</p> <p>Corner points connected to two or more CTRIA6 elements:</p> $\sigma_c = \frac{1}{N_c} \sum_{i=1}^{N_c} \sigma_{r_i}$ <p>where:</p> <p>N_c = number of CTRIA6 elements connected to corner point.</p>	Same as remark for interior points

Table 8-3 Evaluation of Average Grid Point Stress for Solid Elements by the Topological Method

Type of Grid Point	Average Grid Point Stress Component	Remarks
Interior and Edge	$\sigma_I = \frac{1}{N_e} \sum_{i=1}^{N_e} \sigma_{ei}$ <p>where: σ_I = average stress at interior or edge grid points N_e = number of directly connected elements σ_{ei} = element vertex stresses</p>	Mixtures of CHEXA, CPENTA, and CTETRA elements may connect to an interior or edge grid point.
Corner	$\sigma_c = \sigma_r$ <p>where: σ_r = stress at element vertex connected to corner point</p>	The same relation is used whether a CHEXA, CPENTA, or CTETRA element is connected to the grid point.

8.2.3 Geometric Method – Surface Elements

In the geometric method, grid point stresses for points which connect to CQUAD8 or CTRIA6 elements use the average of the element corner stresses. If any CQUAD4 or CTRIA3 elements connect to these points, their influence is not considered.

The geometric method for the determination of average grid point stresses which connect only to CQUAD4 or CTRIA3 elements utilizes the numerical technique of least squares. The “best fit” value of the stress Σ at a point (x, y) is functionally denoted as:

$$\Sigma(x,y) = A_0 + A_1x + A_2y \quad \text{Eq. 8-1}$$

where the coefficients A_0 , A_1x , and A_2 must be determined. The coordinates x and y are the locations of points at which known stresses exist. In accordance with the least squares technique, it is required to minimize the function:

$$f = \frac{1}{2} \left\{ \sigma - \Sigma \right\}^T \left\{ \sigma - \Sigma \right\} \quad \text{Eq. 8-2}$$

with respect to $\{A\}$. Here σ represents the known element stresses.

To perform the minimization, **Eq. 8-1** is written in matrix form as:

$$\left\{ \Sigma \right\} = [T]\{A\} \quad \text{Eq. 8-3}$$

where:

$$[T] = \begin{bmatrix} 1 & X_1 & X_2 \\ \vdots & \vdots & \vdots \\ i & X_i & X_i \end{bmatrix}$$

and **Eq. 8-2** is differentiated with respect to the coefficients A_i .

$$\frac{\partial f}{\partial A_i} = [T]^T [T]\{A\} - [T]^T \{\sigma\} = 0 \quad \text{Eq. 8-4}$$

The coefficients $\{A\}$ are then determined from **Eq. 8-4** to be:

$$\{A\} = ([T]^T [T])^{-1} [T]^T \{\sigma\} \quad \text{Eq. 8-5}$$

The average grid point stress at a given grid point can then be calculated by substituting **Eq. 8-5** into the matrix representation of **Eq. 8-1** as follows:

$$\Sigma = [1 \ X_o \ Y_o]([T]^T [T])^{-1} [T]^T \{\sigma\} = [Q]\{\sigma\} \quad \text{Eq. 8-6}$$

where:

Q = matrix of interpolation factors

$X_o \ Y_o$ = locations of grid points at which stress is required

Note that if $([T]^T [T])^{-1}$ is singular, the matrix of interpolation factors is taken to be $1/N_I$ so that **Eq. 8-6** becomes:

$$\Sigma = \frac{1}{N_I} \sum_{i=1}^{N_I} \sigma_{ei} \quad \text{Eq. 8-7}$$

where N_I = number of known stress points associated with the grid point. In most cases, **Eq. 8-7** is equivalent to the topological interpolation method.

The strategy used in the evaluation of Eq. 8-6 for interior, edge, and corner grid points is presented in Table 8-4.

As previously mentioned, the coordinates x and y of Eq. 8-1 are the locations of points at which known stresses exist. These locations are actually taken as the projected coordinates of the known stress points on the average surface evaluated at the grid point at which the stress is to be computed. This average surface is determined to be the surface whose normal is the average of the unit normals to the elements connected to the grid point of interest. The local x -axis used in the evaluation of the geometric coefficients of the matrix $[T]$ is taken as the projection of the x -axis of the output coordinate system on the average surface evaluated at the grid point of interest.

Table 8-4 Evaluation of Average Grid Point Stresses for Plate and Shell Elements by the Geometric Method

Type of Grid Point	Number of Elements Involved in Stress Calculation
Interior	All directly connected elements
Edge	<p><u>Two directly connected elements:</u> Use the two known element stresses plus the stresses at all interior points connected by line segments of the directly connected elements.</p> <p><u>Three or more directly connected elements:</u> Use the known element stresses of the directly connected elements.</p>
Corner	Same as topological method (see Table 8-1).

8.2.4 Treatment of Exception Points -- “BRANCHing”

Exception points have been defined as grid points at which stress discontinuities may occur. Such discontinuities may occur because a substantial difference in slope exists between adjacent elements directly connected to the same line segment or if more than two elements are directly connected to the same line segment. Examples of these two cases are illustrated below.

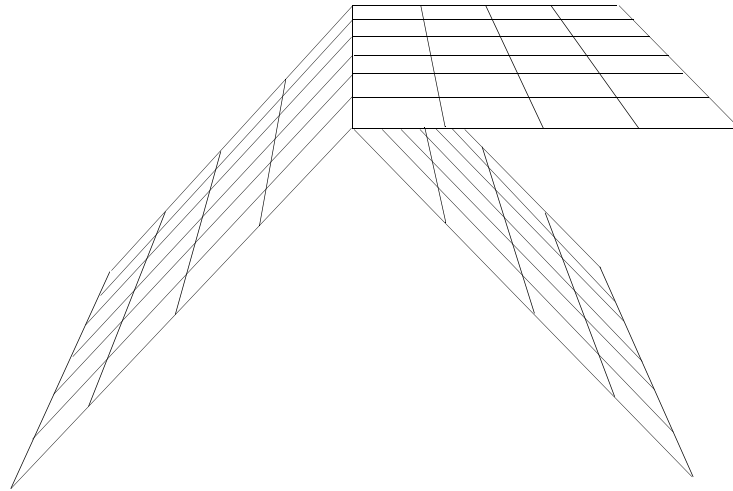


Figure 8-1 Three or More Elements Connected to the Same Line Segment

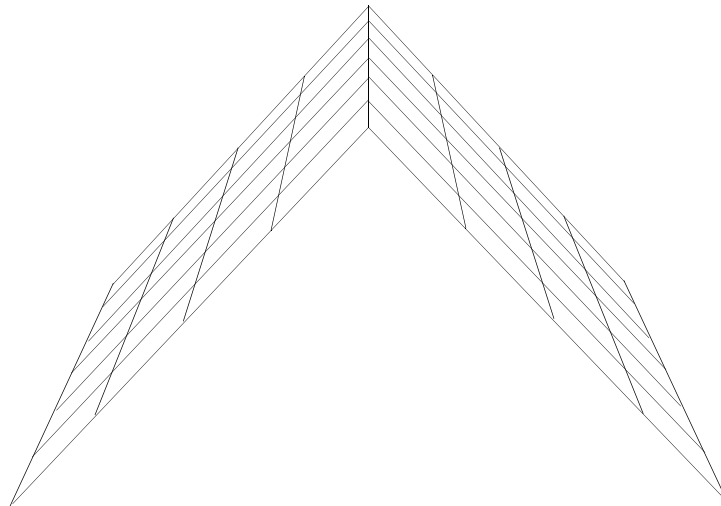


Figure 8-2 Slope Difference between Adjacent Elements Connected to the Same Line

Note: That these two situations are geometric rather than topological in nature and, therefore, are only considered under the geometric interpolation option.

Since exception points are defined as grid points at which stress discontinuities may exist, it is conceivable that a computed average grid point stress at an exception point may be different for each of the directly connected elements. However, it is more common to find that a subset of the elements that are directly connected to an exception point forms a subsurface on which stresses may be considered to be continuous. A unique average grid point stress may then be computed for the elements of the subsurface.

The calculation of average grid point stresses for a subsurface that contains two or more elements is performed with the geometric interpolation method used for edge points. If a subsurface consists of a single element, the calculation of an average grid point stress is performed in the same manner as utilized in the calculation of average grid point stresses at corner points. It must be noted, however, that σ_D , σ_{E1} , and σ_{E2} may be interior, exterior, or corner points (see [Table 8-4](#)).

8.2.5 User Information

The use of the grid point stress recovery option is demonstrated in this section through the consideration of several example problems. These problems are discussed separately after the presentation of the requirements that must be met in the Executive Control Section, the Case Control Section, and the Bulk Data Section.

Executive Control Section

A typical Executive Control Section would be of the following form:

```
SOL 101
CEND
```

Case Control Section

The option to obtain grid point stress output is exercised through commands in the Case Control Section. The general request for the recovery of grid point stresses is accomplished using commands with the form:

```
GPSTRESS = ALL
```

or

```
SET j = k, l, m, . . .
GPSTRESS = j
```

The k, l, m, \dots reference the identification numbers of the surfaces or volumes of elements that are defined in the OUTPUT(POST) section of the Case Control Section. Grid point stress data will be output only for those surfaces or volumes referenced through the GPSTRESS Case Control command.

Note: If grid point stress data is desired, an ELSTRESS (or STRESS) request for all elements in the surfaces or volumes of interest must be included in the Case Control Section. Such requests will cause the output of element stress data as well as grid point stress data. If the output of element stress data is not desired, the requests ELSTRESS(PLOT) = or STRESS(PLOT) = may be used.

In static analysis, one may place the GPSTRESS command above the subcase level or in individual subcases. If the GPSTRESS command is above the subcase level, it will be utilized by all subcases and subcoms unless overridden by a GPSTRESS command in a subcase or subcom.

The OUTPUT(POST) section of the Case Control Section contains specific requests for grid point stress data for the surfaces or volumes of elements. Any number of element surfaces or volumes may be defined, but only those surfaces or volumes that are referenced through the GPSTRESS command will have grid point stress data generated and output.

Element surfaces and volumes are defined through a SURFACE and VOLUME commands in the OUTPUT(POST) section of the Case Control Section.

8.2.6 Grid Point Stress Output Description

Listing 8-1 through **Listing 8-3** are examples of grid point stress output produced for two-dimensional plate elements requested on the SURFACE Case Control command. **Listing 8-1** is the most common form of output and is always the result when the TOPOLOGICAL method is used, and when the GEOMETRIC method is used and no surface “BRANCHing” is detected. Note that when the “ELEMENT ID” value is zero, it always indicates that all elements in the SURFACE connected to the grid point contribute to the average grid point stress and lie in the “best” subsurface. **Listing 8-2** and **Listing 8-3** show examples of the output when the GEOMETRIC method is used and when “BRANCHing” occurs. The negative value for the “ELEMENT ID” indicates the “best” average grid point stress and the element identification number of an element associated with “best” subsurface. Similarly a positive element identification numbers indicate another average grid point stress for a “less best” grid point stress and the element identification number associated with it’s subsurface. When TOLERANCE is set positive as in **Listing 8-2** only the “best” grid point stress is calculated when “BRANCHing” is detected and when TOLERANCE is set negative all subsurface average grid point stress are calculated. See “**SURFACE**” on page 443 of the *MSC.Nastran Quick Reference Guide*.

Listing 8-1 Grid Point Stress Output – Typical

SQUARE PLATE WITH CIRCULAR HOLE										JUNE 24, 1991 MSC/NASTRAN 6/14/91 PAGE 27		
GRID POINT STRESS TEST CASE												
UNIFORM LOAD ALONG X=5.0.										SUBCASE 1		
SUBCASE = 1												
STRESSES AT GRID POINTS - - SURFACE 4										1		
SURFACE X-AXIS X NORMAL(Z-AXIS) Z										REFERENCE COORDINATE SYSTEM FOR SURFACE DEFINITION CID		
STRESSES IN SURFACE SYSTEM										PRINCIPAL STRESSES		
MAX										VON MISES		
GRID ID	ELEMENT ID	FIBER	NORMAL-X	NORMAL-Y	SHEAR-XY	ANGLE	MAJOR	MINOR	SHEAR	VON MISES		
10304	0	MID	4.965E+03	4.041E+03	-6.431E+03	-42.9450	1.095E+04	-1.944E+03	6.447E+03	1.204E+04		
10305	0	MID	6.132E+03	3.839E+03	-5.518E+03	-39.1310	1.062E+04	-6.503E+02	5.636E+03	1.096E+04		
10306	0	MID	7.033E+03	3.408E+03	-4.246E+03	-33.4429	9.837E+03	6.037E+02	4.617E+03	9.549E+03		
10400	0	MID	6.101E+02	1.631E+04	-1.038E+03	-86.2353	1.638E+04	5.418E+02	7.919E+03	1.612E+04		
10401	0	MID	1.463E+03	1.417E+04	-3.402E+03	-75.9185	1.503E+04	6.098E+02	7.209E+03	1.473E+04		
10402	0	MID	2.228E+03	1.211E+04	-5.354E+03	-66.3558	1.446E+04	-1.156E+02	7.286E+03	1.452E+04		
10403	0	MID	2.809E+03	1.019E+04	-6.381E+03	-60.0211	1.387E+04	-8.721E+02	7.371E+03	1.433E+04		
10404	0	MID	3.159E+03	8.514E+03	-6.308E+03	-56.5007	1.269E+04	-1.016E+03	6.852E+03	1.323E+04		
10405	0	MID	3.307E+03	7.318E+03	-5.482E+03	-55.0445	1.115E+04	-5.249E+02	5.837E+03	1.142E+04		

Listing 8-2 Grid Point Stress Output with “BRANCHing” and TOLERANCE > 0

INTERSECTING PLATES										JUNE 24, 1991 MSC/NASTRAN 6/14/91 PAGE 34		
GRID POINT STRESS TEST CASE												
SUBCASE = 1												
STRESSES AT GRID POINTS - - SURFACE 4										0		
SURFACE X-AXIS X NORMAL(Z-AXIS) Z										REFERENCE COORDINATE SYSTEM FOR SURFACE DEFINITION CID		
STRESSES IN SURFACE SYSTEM										PRINCIPAL STRESSES		
MAX										VON MISES		
GRID ID	ELEMENT ID	FIBER	NORMAL-X	NORMAL-Y	SHEAR-XY	ANGLE	MAJOR	MINOR	SHEAR	VON MISES		
10304	0	Z1	9.107E+02	1.209E+03	2.799E+03	46.5224	3.863E+03	-1.743E+03	2.803E+03	4.969E+03		
		Z2	1.052E+03	1.571E+03	2.913E+03	47.5464	4.236E+03	-1.613E+03	2.925E+03	5.233E+03		
10305	0	MID	9.813E+02	1.390E+03	2.856E+03	47.0453	4.049E+03	-1.678E+03	2.863E+03	5.099E+03		
		Z1	1.906E+03	5.699E+02	1.789E+03	34.7594	3.147E+03	-6.714E+02	1.909E+03	3.531E+03		
10306	0	Z2	2.032E+03	5.648E+02	1.934E+03	34.6114	3.367E+03	-7.698E+02	2.068E+03	3.811E+03		
		MID	1.969E+03	5.674E+02	1.861E+03	34.6824	3.257E+03	-7.206E+02	1.989E+03	3.671E+03		
10400	-119	Z1	2.771E+03	-1.983E+02	2.678E+02	5.1132	2.795E+03	-2.223E+02	1.508E+03	2.912E+03		
		Z2	3.024E+03	-1.118E+02	4.889E+02	8.6589	3.098E+03	-1.862E+02	1.642E+03	3.196E+03		
10401	-119	MID	2.897E+03	-1.550E+02	3.783E+02	6.9615	2.944E+03	-2.012E+02	1.572E+03	3.049E+03		
		Z1	-5.775E+04	1.181E+03	-2.060E+04	-72.5215	7.667E+03	-6.423E+04	3.595E+04	6.839E+04		
10402	-119	Z2	-5.821E+04	1.207E+04	-1.647E+04	-77.4454	1.573E+04	-6.188E+04	3.880E+04	7.106E+04		
		MID	-5.798E+04	6.623E+03	-1.853E+04	-75.0777	1.156E+04	-6.292E+04	3.724E+04	6.942E+04		
10403	-119	Z1	-1.856E+04	7.060E+03	-4.939E+03	-79.4558	7.979E+03	-1.948E+04	1.373E+04	2.446E+04		
		Z2	-1.714E+04	1.656E+04	-3.881E+03	-83.5131	1.700E+04	-1.758E+04	1.729E+04	2.995E+04		
10404	-119	MID	-1.785E+04	1.181E+04	-4.410E+03	-81.7177	1.245E+04	-1.849E+04	1.547E+04	2.696E+04		

Listing 8-3 Grid Point Stress Output with “BRANCHing” and TOLERANCE < 0

INTERSECTING PLATES		JUNE 24, 1991 MSC/NASTRAN 6/14/91 PAGE 40								
GRID POINT STRESS TEST CASE										
SUBCASE = 1										
STRESSES AT GRID POINTS -- SURFACE 5										
SURFACE X-AXIS X NORMAL(Z-AXIS) Z REFERENCE COORDINATE SYSTEM FOR SURFACE DEFINITION CID 0										
GRID ID	ELEMENT ID	FIBER	STRESSES IN SURFACE SYSTEM			PRINCIPAL STRESSES				MAX
			NORMAL-X	NORMAL-Y	SHEAR-XY	ANGLE	MAJOR	MINOR	SHEAR	
10304	0	Z1	9.107E+02	1.209E+03	2.799E+03	46.5224	3.863E+03	-1.743E+03	2.803E+03	4.969E+03
		Z2	1.052E+03	1.571E+03	2.913E+03	47.5464	4.236E+03	-1.613E+03	2.925E+03	5.233E+03
		MID	9.813E+02	1.390E+03	2.856E+03	47.0453	4.049E+03	-1.678E+03	2.863E+03	5.099E+03
10305	0	Z1	1.906E+03	5.699E+02	1.789E+03	34.7594	3.147E+03	-6.714E+02	1.909E+03	3.531E+03
		Z2	2.032E+03	5.648E+02	1.934E+03	34.6114	3.367E+03	-7.698E+02	2.068E+03	3.811E+03
		MID	1.969E+03	5.674E+02	1.861E+03	34.6824	3.257E+03	-7.206E+02	1.989E+03	3.671E+03
10306	0	Z1	2.771E+03	-1.983E+02	2.678E+02	5.1132	2.795E+03	-2.223E+02	1.508E+03	2.912E+03
		Z2	3.024E+03	-1.118E+02	4.889E+02	8.6589	3.098E+03	-1.862E+02	1.642E+03	3.196E+03
		MID	2.897E+03	-1.550E+02	3.783E+02	6.9615	2.944E+03	-2.012E+02	1.572E+03	3.049E+03
10400	-119	Z1	-5.775E+04	1.181E+03	-2.060E+04	-72.5215	7.667E+03	-6.423E+04	3.595E+04	6.839E+04
		Z2	-5.821E+04	1.207E+04	-1.647E+04	-77.4454	1.573E+04	-6.188E+04	3.880E+04	7.106E+04
	219	MID	-5.798E+04	6.623E+03	-1.853E+04	-75.0777	1.156E+04	-6.292E+04	3.724E+04	6.942E+04
		Z1	-5.772E+04	1.152E+03	2.064E+04	72.4819	7.667E+03	-6.423E+04	3.595E+04	6.839E+04
	220	Z2	-5.819E+04	1.204E+04	1.652E+04	77.4058	1.573E+04	-6.188E+04	3.880E+04	7.106E+04
		MID	-5.795E+04	6.597E+03	1.858E+04	75.0381	1.156E+04	-6.292E+04	3.724E+04	6.942E+04
10401	-119	Z1	-1.856E+04	7.060E+03	-4.939E+03	-79.4558	7.979E+03	-1.948E+04	1.373E+04	2.446E+04
		Z2	-1.714E+04	1.656E+04	-3.881E+03	-83.5131	1.700E+04	-1.758E+04	1.729E+04	2.995E+04
	120	MID	-1.785E+04	1.181E+04	-4.410E+03	-81.7177	1.245E+04	-1.849E+04	1.547E+04	2.696E+04
		Z1	-1.856E+04	7.060E+03	-4.939E+03	-79.4558	7.979E+03	-1.948E+04	1.373E+04	2.446E+04
	219	Z2	-1.714E+04	1.656E+04	-3.881E+03	-83.5131	1.700E+04	-1.758E+04	1.729E+04	2.995E+04
		MID	-1.785E+04	1.181E+04	-4.410E+03	-81.7177	1.245E+04	-1.849E+04	1.547E+04	2.696E+04
220	Z1	-1.855E+04	7.053E+03	4.957E+03	79.4163	7.979E+03	-1.948E+04	1.373E+04	2.446E+04	
	Z2	-1.713E+04	1.655E+04	3.905E+03	83.4735	1.700E+04	-1.758E+04	1.729E+04	2.995E+04	
	MID	-1.784E+04	1.180E+04	4.431E+03	81.6781	1.245E+04	-1.849E+04	1.547E+04	2.696E+04	
	Z1	-1.855E+04	7.053E+03	4.957E+03	79.4163	7.979E+03	-1.948E+04	1.373E+04	2.446E+04	
220	Z2	-1.713E+04	1.655E+04	3.905E+03	83.4735	1.700E+04	-1.758E+04	1.729E+04	2.995E+04	
	MID	-1.784E+04	1.180E+04	4.431E+03	81.6781	1.245E+04	-1.849E+04	1.547E+04	2.696E+04	

8.2.7 General Remarks

Each element stress component (independently of all other stress components) is transformed into the output coordinate system before interpolation. It is necessary that this output coordinate system vary slowly over the surface for the interpolated stresses to be accurate. This approach is certainly valid for stress interpolation over elements that are in a common plane and it should be adequate for engineering analysis when the difference in slope between adjacent elements is small. However, when the elements do vary substantially from a smooth surface, the analyst must utilize engineering judgment relative to the accuracy of the resultant stress data.

Average grid point stresses are only computed at element vertices. For elements with mid-edge grid points, one can estimate the stress at these mid-edge grid points to be the average of the computed stresses at the grid point at the ends of the edge in question.

8.3 Mesh Stress Discontinuities at Grid Points

Error Estimates for Grid Point Stress Data

“[Stress Recovery at Grid Points](#)” on page 466 is devoted to a discussion of the averaging procedures utilized to provide meaningful stresses at the grid points of finite element models in MSC.Nastran. This section will focus on the description of error estimators to assist the analyst in the identification of regions of his model that may require refinement.

The essence of the previous averaging procedures used to obtain grid point stress data is to:

1. Convert these local element stress components into a common user-defined coordinate system.
2. Average the several values of each stress component to obtain a unique value of the stress component that is to be associated with the grid point in question.
3. Compute the stress invariants at the grid points from the stress components at the grid points.

In the general case, the stress components are σ_x , σ_y , σ_z , τ_{xy} , τ_{xz} , and τ_{yz} .

For discussion purposes, the averaging process used to compute the stress components at the grid points can be represented in the form of [Eq. 8-8](#).

$$\sigma_g = \sum_{i=1}^{N_e} (W_i \sigma_{ei}) \quad \text{Eq. 8-8}$$

where:

σ_g = weighted mean value of the stress component computed at the grid point.

σ_{ei} = value of the stress component in the i th element ($i = 1, 2, \dots, N_e$) in the neighborhood of the grid point. δ_{gi} is in the same coordinate system as δ_g .

W_i = weighting factor assigned to the i th element. The sum of the N_e values of W_i must equal 1. (This requirement assures that all computed statistics will be unbiased.) The attribute of being unbiased implies that the variance is equal to the mean square error. Equal weighting (i.e., $W_i = 1/N_e$) is assumed.

An estimate of the error in a particular component of stress at a grid point can then be computed by assuming that the values of the corresponding stress components computed for the elements in the neighborhood of the grid point are data points with uncorrelated random errors. It then follows that an estimate of the probable error in the stress component δ_g at the grid point is:

$$\begin{aligned} \delta_g &= \sqrt{\sum_{i=1}^{N_e} (W_i \delta_{ei})^2} \\ &= \frac{1}{\sqrt{N_e}} \sqrt{\sum_{i=1}^{N_e} (\delta_{ei})^2} \end{aligned} \tag{Eq. 8-9}$$

where $\delta_{ei} = \sigma_{ei} - \sigma_g$. Thus, the probable error δ_g is the root mean square error in δ_{ei} divided by $\sqrt{N_e}$.

It should be noted that the standard deviation or root mean square error is a reasonable measure of precision in many practical cases, but it is easy to provide examples in which the standard deviation is a poor measure of the concentration of the distribution about the mean. Eq. 8-9 is assumed to provide an approximate error estimator for the grid point stress data.

Element and Grid Point Stress Discontinuity Output Requests

It should be clear that element and grid point stress discontinuity output can only be obtained if the analyst has requested grid point stress output via the GPSTRESS command in the Case Control Section of the MSC.Nastran input file. The GPSTRESS command, of course, requires that the analyst define all required SURFACES and VOLUMES in the OUTPUT(POST) portion of the Case Control Section.

To output Grid Point Stress DisCONTinuities:

GPSDCON = ALL

or

SET j = k, l, m,
GPSDCON = j

To output ELement Stress DisCONTinuities:

ELSDCON = ALL

or

SET j = k, l, m,

ELSDCON = j

The following remarks should be noted relative to the use of the GPSDCON and/or ELSDCON commands.

1. The GPSDCON and ELSDCON commands are honored only in SOLUTION Sequences 101, 114, 144, and 200.
2. The GPSDCON and ELSDCON commands may be placed above the SUBCASE level or in individual SUBCASEs and/or SUBCOMs. The use of these commands above the SUBCASE level causes stress discontinuity data to be output for ALL SUBCASEs in the MSC.Nastran input file. If the analyst wishes to restrict stress discontinuity output to specific SUBCASEs, GPSDCON and/or ELSDCON commands should appear only under these specific SUBCASEs.
3. The analyst is cautioned that these commands currently produce a substantial amount of output.
4. The GPSDCON and/or ELSDCON commands may be effectively used in restarts. For example, one could include requests for GPSTRESS data during the initial run of a project and then obtain data on stress discontinuities on a subsequent restart by including GPSDCON and/or ELSDCON commands in the Case Control Section.
5. Stress discontinuity output will only be provided when grid point stresses have been previously computed by the methods described in this article. Thus, stress continuity will not be computed when, for example, both plate and solid elements are connected to a grid point that is involved in stress discontinuity calculations.

Discussion of Error Measures

Large values of error estimators for CQUAD4 and CTRIA3 elements are not uncommon occurrences in statistical error measures. For example, the mean and variance have direct analogs in engineering mechanics. The mean is analogous to the centroid of a body and is thus a measure of where the mass is centered. The variance is the second moment about the mean, and it tends to be small if the majority of the mass is concentrated about the centroid. As the mass is dispersed further from the centroid, the moment of inertia tends to increase. If the mass is concentrated at the centroid (as it might be in some idealized lumped mass models), the moment of inertia becomes zero.

Thus, the error estimates for CQUAD4 and CTRIA3 elements will tend toward larger values because the data used to compute the estimators are dispersed relatively far from the mean. For those elements that provide stress data at vertices, the estimators will tend toward smaller values because the stress data tend to be concentrated about the mean in well-designed finite element models.

Generally, the goal is to design a finite element mesh for static analysis so that all important stress gradients will be adequately represented. If the mesh is not sufficiently detailed, the stress data at element vertices or element centroids will result in the relatively large values of the error estimators.

Very inaccurate values of these error estimators may be obtained at the edges of defined SURFACES and on the faces of defined VOLUMES. Such edges and surfaces will often correspond with the boundaries of superelements.

In summary, error estimators can, in some cases, be highly inaccurate. Nevertheless, error estimate data is quite useful when properly interpreted by the analyst.

Grid Point Stresses and Mesh Stress Discontinuities

Grid point stresses may be requested for grid points connected by CTRIA3, CTRIA6, CTRIAR, CQUAD4, CQUAD8, CQUADR, CHEXA, CPENTA, and CTETRA elements (with linear material properties only). In linear static analysis only, a mesh stress discontinuity error analysis may then be performed based on the computed element and grid point stresses.

Grid point stresses are not computed for laminar stresses.

The grid point stresses are calculated in the user-specified coordinate system. The following real membrane stresses are output on request:

- Normal stresses in the x and y directions.
- Shear stresses on the x face in the y direction.
- Angle between the x axis and the major principal axis.
- Major and minor principal stresses.
- von Mises equivalent or maximum shear stress (see STRESS Case Control command).

The following real stresses are output on request for grid points connected to solid elements:

- The three normal stresses.
- The three shear stresses.

- Magnitude and direction of the three principal stresses.
- The mean pressure.
- The von Mises equivalent or octahedral stress (see STRESS Case Control command).

Only real stresses are available at the grid points. The user has the following options:

- The capability to vary output requests by SUBCASE.
- Choice of elements in stress field.
- Choice of topological or geometric interpolation (surface fields only).
- Choice of output coordinate system.
- Choice of output stress fiber locations (surface fields only).

The grid point stresses are requested in the Case Control Section for printing and postprocessing using the Case Control commands GPSTRESS (Grid Point Stress) and STRFIELD (Stress Field). The user is required to request element stress output (STRESS = N) for all elements referenced by selected SURFACE and VOLUME commands. The grid point stress requests a set of surface or volume fields which are defined in the OUTPUT(POST) part of the Case Control Section on the SURFACE or VOLUME commands, respectively. The postprocessing output requests are separated from case control and plot requests by the OUTPUT(POST) command.

The general rules for Case Control syntax are the same as for plotting (see “[Plotting](#)” on page 501). The grid point stress output is defined on the SURFACE and VOLUME commands. The SURFACE command defines the following: output coordinate system, list of elements in the stress surface field, output stress fiber location for stress evaluation and the choice of topological or geometric interpolation method used to determine the grid point stresses from the elements. The VOLUME command defines the following: output coordinate system, list of elements in the stress volume, and the choice of DIRECT and/or PRINCIPAL stresses.

In the linear static structured solution sequences only (SOLutions 101, 114, 144, and 200) mesh stress discontinuities may be requested to estimate errors due to the coarseness of the finite element mesh. The estimation of the error is computed for the element and grid point stresses and requested by the ELSDCON and GPSDCON Case Control commands, respectively. These commands select the desired surfaces and volumes which are defined in the OUTPUT(POST) Section of Case Control. This output option also requires a corresponding GPSTRESS or STRFIELD request for the desired surfaces and volumes and a STRESS Case Control command for the elements in those surfaces and volumes. If PARAM,POST,0 is specified, then this output will also be written to the DBC database.

The following is computed and output for stress discontinuities on shell elements:

- Normal stress discontinuities in the x and y directions.
- Shear stress discontinuities on the xy direction.
- Major and minor principal stress discontinuities.
- von Mises and maximum shear stress discontinuities.
- Error estimate.

The following is computed and output for stress discontinuities on solid elements:

- Normal stress discontinuities in the x, y, and z directions.
- Shear stress discontinuities on the xy, yz, and xz directions.
- Major and minor principal stress discontinuities.
- von Mises stress and mean pressure discontinuities.
- Error estimate.

CHAPTER

9

DMAP

■ Introduction

9.1 Introduction

MSC.Nastran provides a wide variety of solution sequences. Each solution sequence consist of a series of DMAP (Direct Matrix Abstraction Program) statements.

MSC.Nastran allows the user to modify solution sequences or to write his or her own solution sequences using DMAP. DMAP is a high level language with its own compiler and grammatical rules. The *MSC.Nastran 2005 DMAP Programmer's Guide* provides detailed information on these modules.

CHAPTER
10

Database Concepts

- Introduction
- Delivery Databases
- Dbset Deletion
- Database Autoassignment
- Database Project and Version
- Database Migration
- Database Archival, Compression, and Transfer
- Database Archival and Retrieval
- Database Compression
- Database Transfer Across Different Computer Types

10.1 Introduction

All MSC.Nastran runs require a database. A database allows for the storage and retrieval of data: data blocks, parameters, scratch files, and DMAP object and source files. During a run data may be written onto only one database. This is called the primary database which is attached automatically or by the INIT MASTER statement. Additional databases may be attached for “read-only” purposes. This includes a “delivery” database and one or more “located” databases. The delivery database contains solution sequences and is automatically attached by the SOL Executive statement or the ACQUIRE FMS statement. “Located” databases are attached by the DBLOCATE statement.

A database is divided into several DBsets which are initialized by the INIT statement. There are two types of DBsets: permanent and scratch. Permanent DBsets may be saved at the end of the run and reused in a restart run. Scratch DBsets are automatically deleted at the end of the run. A DBset may be composed of up to twenty concatenated physical files or DBset members. Each member is assigned to a physical file by the ASSIGN statement.

By default, there are four DBsets—two permanent and two scratch—which are predefined and automatically assigned by the program. The DBset names of the predefined permanent DBsets are: MASTER and DBALL. The DBset-names of the predefined scratch DBsets are: SCRATCH and OBJSCR. The DBsets are briefly described below:

- The MASTER DBset is the directory of the database. It contains the names of all DBsets, DBset members and their physical file names, and a directory of projects, versions, data blocks, parameters, DMAP source and object files. It also contains the NDDL scheme used to describe the database.
- The DBALL DBset contains all the DMAP data blocks which may be saved permanently for reuse in a subsequent run. In the solution sequences in which restarts are not possible, (e.g., SOL 1) this DBset is empty.
- The SCRATCH DBset is a temporary DBset for all scratch data blocks and files. It has two partitions: one for DMAP data blocks and one for DMAP module internal scratch files.
- The OBJSCR DBset is a temporary DBset for DMAP compilation.
- If the user desires to create his or her own DMAP source and object files, two other DBsets (USROBJ and USRSOU) need to be allocated.

- The USROBJ and USRSOU DBsets contain DMAP source and object files that may be saved permanently for execution in a subsequent run. See the COMPILE Executive Control statement.

If an INIT and ASSIGN statement are not specified for one or more of the predefined DBsets, then the corresponding statement(s) are provided by default.

```

ASSIGN MASTER='dbs-name.MASTER'
      INIT    MASTER(RAM) LOGICAL=(MASTER(5000))
ASSIGN DBALL='dbs-name.DBALL'
      INIT    DBALL LOGICAL=(DBALL(25000))
ASSIGN USROBJ='dbs-name.USROBJ'
      INIT    USROBJ LOGICAL=(USROBJ(5000))
ASSIGN USRSOU='dbs-name.USRSOU'
      INIT    USRSOU LOGICAL=(USRSOU(5000))
ASSIGN SCRATCH='temp-name.SCRATCH'
ASSIGN SCR300='temp-name.SCR300'
      INIT    SCRATCH(MEM) LOGICAL=(SCRATCH(175000)),
      SCR300=(SCR300(175000))
ASSIGN OBJSCR='temp-name.OBJSCR'
      INIT    OBJSCR LOGICAL=(OBJSCR(5000))

```

The numbers inside the parentheses are the maximum sizes of the DBsets in GINO blocks. By default, one member with the same log-name as the DBset-name is assigned to each DBset. (For a description of a GINO block, DBset-name and log-name, see the INIT statement). For example, the DBALL DBset has one member also called DBALL. The dbs-name is determined from the dbs keyword on the nastran command (see “[nastran Command and NASTRAN Statement](#)” on page 1 of the *MSC.Nastran Quick Reference Guide*) and temp-name is generated by the command procedure (see the *MSC.Nastran 2004 Installation and Operations Guide*). The scratch DBsets are assigned special names on a scratch disk. (For a description of the dbs-name and temp-name, see the *MSC.Nastran 2004 Installation and Operations Guide*). The filenames above are those that would be generated on UNIX-type computers. For example, if the name of the input file is called MYJOB.DAT and the dbs keyword is not specified, then the default filenames for the permanent DBsets would be:

```

MYJOB.MASTER
MYJOB.DBALL

```

Also, they would have the same directory or path as the input file.

A physical filename is automatically created for any new DBsets specified by an INIT statement. For example, to create a new DBset called DBUP for split database operations, specify:

```
INIT DBUP
```

The physical filename will then be: MYJOB.DBUP. For another example, to create two members for DBALL, specify

```
INIT DBALL LOGI=(DB1 ,DB2)
```

Their physical filenames will be MYJOB.DB1 and MYJOB.DB2.

The ASSIGN statement may be used to override these default names. For example, to rename MYJOB.DBUP to FENDER.DBUP, specify

```
ASSIGN DBUP=' FENDER .DBUP '
INIT DBUP
```

To rename MYJOB.DB1 and MYJOB.DB2, specify

```
ASSIGN DB1 = ' FENDER .DBALL1 '
ASSIGN DB2= ' FENDER .DBALL2 '
INIT DBALL LOGI=(DB1 ,DB2)
```

The INIT statement may be used to change the maximum size of the predefined DBsets. (The size may be specified in blocks, kilowords, megawords, kilobytes, or megabytes.) For example, to specify a size of 50,000 blocks for DBALL,

```
INIT DBALL LOGICAL=(DBALL(50000))
```

or 50,000 bytes,

```
INIT DBALL LOGICAL=(DBALL(50KB))
```

The INIT and ASSIGN statement may be used to specify more than one member for a DBset which may exist on different physical devices.

```
ASSIGN DB1 = ' physical file name of DB1 on disk 1 '
ASSIGN DB2= ' physical file name of DB2 on disk 2 '
INIT DBALL LOGICAL=(DB1(5000) ,DB2(5000))
```

10.2 Delivery Databases

The Delivery Database contains the solution sequences. This database, in addition to the primary database, is automatically assigned in order to execute an MSC.Nastran solution sequence. For example, on UNIX-type computers, the filenames of the database are

```
SSS.MASTERA  
SSS.MSCOBJ  
SSS.MSCSOU
```

If the solution sequence is not an MSC.Nastran solution sequence, as in the case of a user's solution sequence; then the ACQUIRE FMS statement may be used to select the appropriate database. ACQUIRE NDDL selects the SSS database. The ACQUIRE statement is required when the DBLOAD FMS statement is used with the ENDJOB FMS statement.

10.3 Dbset Deletion

The disposition of permanent DBsets in the primary database at the end of the run is controlled in several ways.

1. If `scr=yes` is specified on the `nastran` command (see “[The nastran Command](#)” on page 2 of the *MSC.Nastran Quick Reference Guide*) or the `INIT MASTER(S)` statement appears in the FMS Section then all DBsets in the primary database are deleted.
2. An individual DBset member is automatically deleted at the end of the run by specifying the `TEMP` keyword on the `ASSIGN` statement. For example, to delete the `USROBJ` and `USRSOU` DBsets at the end of the run, specify:

```
ASSIGN    USROBJ='A' TEMP
ASSIGN    USROBJ='B' TEMP
```

However, DBsets deleted in this manner or manually cannot be recreated in a future run.

3. The `DBSETDEL` FMS statement provides a better method than the `TEMP` keyword on the `ASSIGN` statement except that the DBsets are deleted at the beginning of the run. The deleted DBsets may be recreated in a future run. For example:

```
DBSETDEL USROBJ,USRSOU
```

10.4 Database Autoassignment

When a database is created, the physical filenames of all DBset members are loaded into the MASTER DBset. Therefore, when using DBLOCATE or RESTART, it is sufficient to assign the MASTER DBset member; and all other DBset members previously initialized, i.e., DBALL, DBUP, etc., are automatically assigned by the program. For example, on restart, the following input will automatically assign the DBALL, USROBJ, and USRSOU DBsets:

```
ASSIGN MASTER= 'filename of the MASTER DBset member'  
RESTART
```

If DBLOCATE is used, then:

```
ASSIGN MSTR1 = 'filename of the MASTER DBset member'  
DBLOCATE LOGI=MSTR1
```

The autoassignment feature cannot assign the physical file of a DBset member that has been renamed. This is because the MASTER DBset member contains the old physical filename. Therefore, an additional ASSIGN statement is required to specify the new filename of the DBset member. If in the previous restart example DBALL is renamed, then the following input is required:

```
ASSIGN MASTER= 'filename of the MASTER DBset member'  
ASSIGN DBALL= 'new filename of the DBALL DBset member'  
RESTART
```

In split database operations, it is often necessary to assign only some of the DBsets because the others are “offline.” The NASTRAN AUTOASGN statement is used to specify the types of databases that will be autoassigned:

1. AUTOASGN = 0: No databases will be autoassigned. This includes the primary database, Delivery Database, and any located databases.
2. AUTOASGN = 1: All databases will be autoassigned. This is the default.
3. AUTOASGN = 2: Only the Delivery Database will be autoassigned.
4. AUTOASGN = 4: Only located databases will be autoassigned.

If some combination of the options above is desired, then sum their values. For example, if the delivery and located databases are to be autoassigned, then specify NASTRAN AUTOASGN = 6.

10.5 Database Project and Version

All data stored on the primary database is assigned to a particular project identification string (ID) and version identification number (ID). The current project ID is specified on the PROJECT FMS statement. The current version ID is an integer value assigned by the program each time the database is attached as the primary database. When a primary database is initialized or a new project is specified, then all items stored on the database during the run are stored under a version ID of integer 1. Then, in subsequent runs a new version is created with an ID equal to the last version ID incremented by one. If the RESTART FMS statement is specified, then a new version of data is created which is equivalent to the last version. In other words, all items in the old version are made available for reuse in the new version.

10.6 Database Migration

Databases between versions in general are not compatible and must be converted, or migrated, before they can be used in a higher version. This is due primarily to the fact that the NDDL sequences have changed. For example, some qualifiers have been added to the paths of some data blocks in order to support new capabilities in the solution sequences. Also, NDDL description of some data blocks have changed. In order to use or restart from a database created in prior versions, the database must be migrated. However, a database created in nonlinear analysis, heat transfer analysis, or optimization analysis cannot be fully migrated. This is because of changes in Bulk Data entry formats and/or data blocks. Also, it would be inefficient to recreate because it is created by modules which perform the bulk of a nonlinear analysis.

Database migration may be performed in a single run which consists of two phases. In the first, the DBLOCATE FMS statement with the CONVERT and COPY keywords is used to copy most of the data blocks and all of the parameters from the earlier version database to the current version primary database.

In the second phase, the remaining data blocks which cannot simply be copied into the current version have to be regenerated. This involves the execution of a short solution sequence that regenerates those data blocks: specify SOL DBTRANS (or SOL 190).

For example, the input file necessary to migrate a database created by one of the structured solution sequences (SOLs 100 through 200) is:

```
ASSIGN MIG= 'physical filename of the V67+ MASTER DBset '  
RFINCL DBTRANS  
SOL DBTRANS  
CEND
```

With the DBTRANS solution sequence, it is assumed that the early version database contains all of the data from a completed and successful analysis. If not, then the solution sequences may fail and the migration may not be successful. In this case, it is possible to alter DBTRANS with the ALTER statement to obtain the appropriate data.

If the earlier version database was not created by an MSC.Nastran solution sequence, then it may be necessary to develop new DBLOCATE statements and a new migration solution sequence.

10.7 Database Archival, Compression, and Transfer

Database archival, compression and transfers are performed using the DBUNLOAD and DBLOAD FMS statements. The user may select all items on the database or selected items according to the WHERE clause, see “WHERE and CONVERT Clauses” in the “[File Management Statements](#)” on page 31 of the *MSC.Nastran Quick Reference Guide*.

10.8 Database Archival and Retrieval

The DBUNLOAD FMS statement is used to archive the primary database. It writes out the data blocks and parameters stored in the database to a sequential file in a binary format which is similar to that of the OUTPUT2 module (see *MSC.Nastran 2005 DMAP Programmer's Guide*). A database can only be unloaded in a run separate from the run in which it is created or modified. The default action is to archive all data blocks and parameters under all projects and versions in the database. The ASSIGN FMS statement is also required to assign a FORTRAN unit on which to unload the database. The default unit number is 50. This unit number and other default attributes are predefined under the logical keyword DBUNLOAD. For example, a simple request to unload or archive a database is as follows:

```
ASSIGN DBUNLOAD= 'physical filename of archive file '  
DBUNLOAD  
ENDJOB
```

The DBLOAD FMS statement is used to read an archived database and copy it to the primary database. The ACQUIRE statement is required to select the appropriate NDDL sequence. The ASSIGN FMS statement is also required to assign a FORTRAN unit to the unloaded database. The default unit number is 51. This unit number and other default attributes are predefined under the logical keyword DBLOAD. For example, a simple request to reload the archived database above is as follows:

```
ACQUIRE NDDL  
ASSIGN DBLOAD= 'physical filename of archive file '  
DBLOAD  
ENDJOB
```

The example above makes use of the ENDJOB statement, which means that only the FMS statements will be processed in the run. A new BUFFSIZE may be assigned to the retrieved database with the NASTRAN statement. SOL DBTRANS or DBTRANU must also be executed:

```
NASTRAN BUFFSIZE=xxxx  
ASSIGN DBLOAD= 'physical filename of archive file '  
DBLOAD  
SOL DBTRANS  
CEND
```

By default, the DBLOAD and DBUNLOAD statements will load and unload all the data under all projects and versions found in the database. The WHERE clause may be used to be more selective according to the project ID, version ID, item name, qualifier values, and/or DBset. Also, the CONVERT clause may be used to modify

the project ID, version ID, item name, qualifier values, and/or DBset. See the “[File Management Statements](#)” on page 31 of the *MSC.Nastran Quick Reference Guide*, for a description of the WHERE and CONVERT clauses.

10.9 Database Compression

When data is deleted from the database, its space is released for the storage of new data and this old space will be reused before any new space is used. However, even if the space is not reused, the size of the database is not reduced. After several restarts or the execution of the DBCLEAN FMS statement, the database may contain a significant amount of released or “dead” space. There are two methods of removing dead space or “compressing” the database. The first method involves the DBUNLOAD and DBLOAD FMS statements, and the examples shown above will also perform a database compression. The second and simplest method involves the DBLOCATE statement specified with the COPY keyword. For example, if the database was created with SOLs 100 through 200, then the input file is:

```
ACQUIRE NDDL
ASSIGN MYDB= 'physical file name of the MASTER DBset'
DBLOCATE LOGI=MYDB COPY
ENDJOB
```

It is also possible to compress an individual DBset and change its maximum size. For example, the input file is:

```
ACQUIRE NDDL
ASSIGN MYDB= 'physical file name of the MASTER DBset'
DBLOCATE LOGI=MYDB COPY WHERE(DBSET='DBALL')
INIT DBALL LOGICAL=(DBALL(10000))
ENDJOB
```

10.10 Database Transfer Across Different Computer Types

It is often desirable to copy a database from one computer to a dissimilar computer (e.g., copying a database created on a CRAY computer to a UNIX workstation). A method is available using the DBUNLOAD and DBLOAD FMS statements. The DBUNLOAD statement is first used to convert the database to a “neutral” file on the first computer. This neutral file is now suitable for a simple copy to the second computer or may be accessed through a network if the two computers are so connected. On the second computer, the DBLOAD statement is then used to convert the neutral file back to a database. It is also necessary to execute the solution sequence DBTRANS (or 190) as described under Database Migration, because some data blocks cannot be converted to a neutral format. (In SOL DBTRANS; data blocks EST, EMAP, ETT, PTELEM, SLT, KDICT, KELM, MDICT, MELM, BDICT, BELM, and ACPT are regenerated.)

The ASSIGN statement is also required to assign the physical filename of the neutral file and specify that the file is formatted since the default format (UNFORMATTED) is not transferable across dissimilar computers.

The following example unloads a database created by one of the unstructured solution sequences on the first computer to a neutral file:

```
ASSIGN DBUNLOAD= 'physical filename of neutral file'  FORMATTED
DBUNLOAD  FORMAT=NEUTRAL
ENDJOB
```

Then the neutral file is copied to, or accessed through a network by, the second computer:

```
ASSIGN DBLOAD= 'physical filename of neutral file'  FORMATTED
DBLOAD  FORMAT=NEUTRAL
SOL DBTRANU
CEND
```

With the DBTRANS solution sequence it is assumed that the neutral file contains all of the data from a completed and successful analysis. If not, then the solution sequences may fail and the transfer may not be successful. In this case, it is possible to alter DBTRANS with the ALTER statement to obtain the appropriate data.

If the database was created in a heat transfer analysis then NASTRAN HEAT = 1 must be specified.

If the database on the first computer was not created by a MSC.Nastran solution sequence, then it may be necessary to develop a new transfer solution sequence.

CHAPTER

11

Plotting

- Introduction
- Postprocessors

11.1 Introduction

General Capability

MSC.Nastran can generate the following kinds of plots:

1. Undeformed geometric projections of the structural model.
2. Static deformations of the structural model by either displaying the deformed shape (alone or superimposed on the undeformed shape), or displaying the displacement vectors at the grid points (superimposed on either the deformed or undeformed shape).
3. Modal deformations resulting from real or complex eigenvalue analysis by the same options stated in 2 above. Complex modes for flutter analysis may be plotted for any user-chosen phase lag.
4. Deformations of the structural model for transient response or frequency response by displaying either vectors or the deformed shape for specified times or frequencies.
5. X-Y graphs of transient response or frequency response.
6. V-F and V-G graphs for flutter analysis.
7. Contour plots of displacements, temperature and stress on the structure.

Structural plots (items 1 through 4) are discussed in the “**Case Control Commands**” on page 175 of the *MSC.Nastran Quick Reference Guide*. X-Y plots (item 5) are discussed in “**X-Y PLOT Commands**” on page 525 of the . Requests for structure plots or X-Y plots are made in the Case Control Section by submitting a structure plot request or an X-Y output request. The optional PLOTID command is considered to be part of the plot request although it must precede any OUTPUT(PLOT), OUTPUT(XYOUT) or OUTPUT(XYPLOT) commands. See “**PLOTID**” on page 378 of the *MSC.Nastran Quick Reference Guide* for a description of the command.

Plot requests are separated from Case Control by the OUTPUT(PLOT), OUTPUT(XYPLOT) or OUTPUT(XYOUT) commands. Data above this command (except PLOTID) will not be recognized by the plotter, even though it may have the same name (for example, the SET command).

Superelement Plotting

Plotting can occur at four different places in the superelement solution sequences. There are two plot commands, SEPLOT and SEUPPLOT, used with other Case Control and PARAM commands to control the type of plot to be prepared. A flow chart of the solution process is given in **Figure 11-1**, **Figure 11-2**, and **Figure 11-3**.

Undeformed structure plots are made early during Phase 0 in the structured solution sequences and during Phase I in the unstructured solution sequences. They can be made for either one superelement only, or for a superelement and all of its upstream members, as controlled by `PARAM,PLOTSUP` (see [Figure 11-1](#) or [Figure 11-2](#)). Plots are made for superelements selected by the `SEMG` command (see [Figure 11-2](#), block I), and listed in a plot request headed by `SEPLOT SEID`, where `SEID` is the superelement identification number. In the structured solution sequences, the `SEMG` command is not required and all superelements which appear on `SEPLOT` commands will be plotted (see [Figure 11-1](#), block 0). Undeformed plots can be used to check geometry and connectivity, and do not require the presence of property or material commands. In the unstructured solution sequences, a branch to the end of the loop immediately after the plot module can be made by using the `PARAM,PLOT,-1` command.

In dynamic analysis, solution set XY-plots are requested by use of `SEPLOT 0` (see [Figure 11-3](#), block II). In the data recovery phase, XY-plots and deformed structure plots for elements in one superelement only are requested by the `SEPLOT` command (see [Figure 11-3](#), block III). Deformed structure plots for a superelement and all its upstream plots are requested with the `SEUPPLOT` command (see [Figure 11-3](#), block IV).

XY-plots and deformed structure plots are regarded as output requests and will result in automatic execution of the data recovery loop as is required to produce the plots requested. For example, the command `SEUPPLOT 0` will result in data recovery being performed on the entire model, even in the absence of any other output requests. For `SEUPPLOT` requests, if `SUBCOM` or `SYMCOM` subcases are used, each superelement must have identical `SUBCOM` and `SYMCOM` structure.

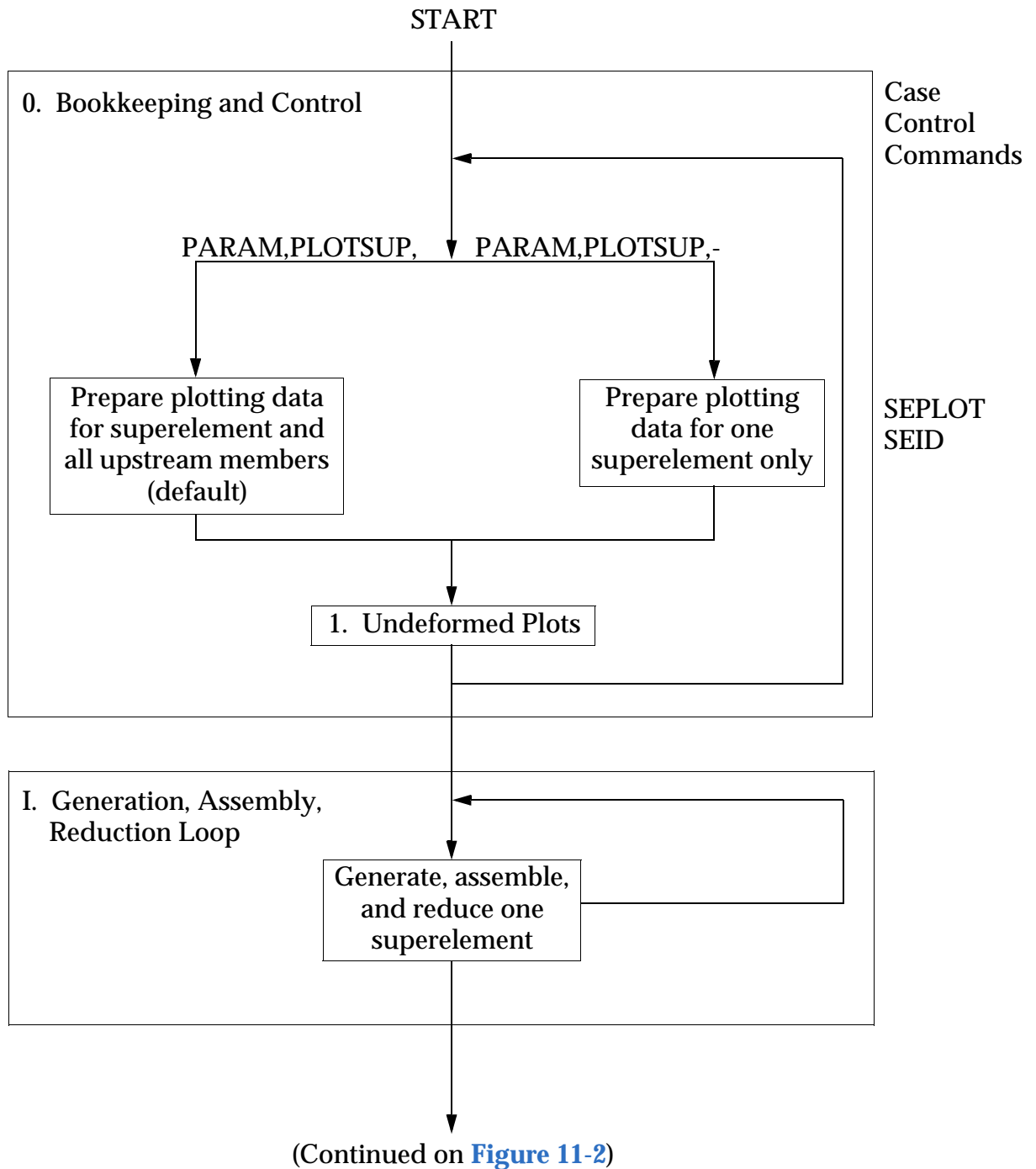
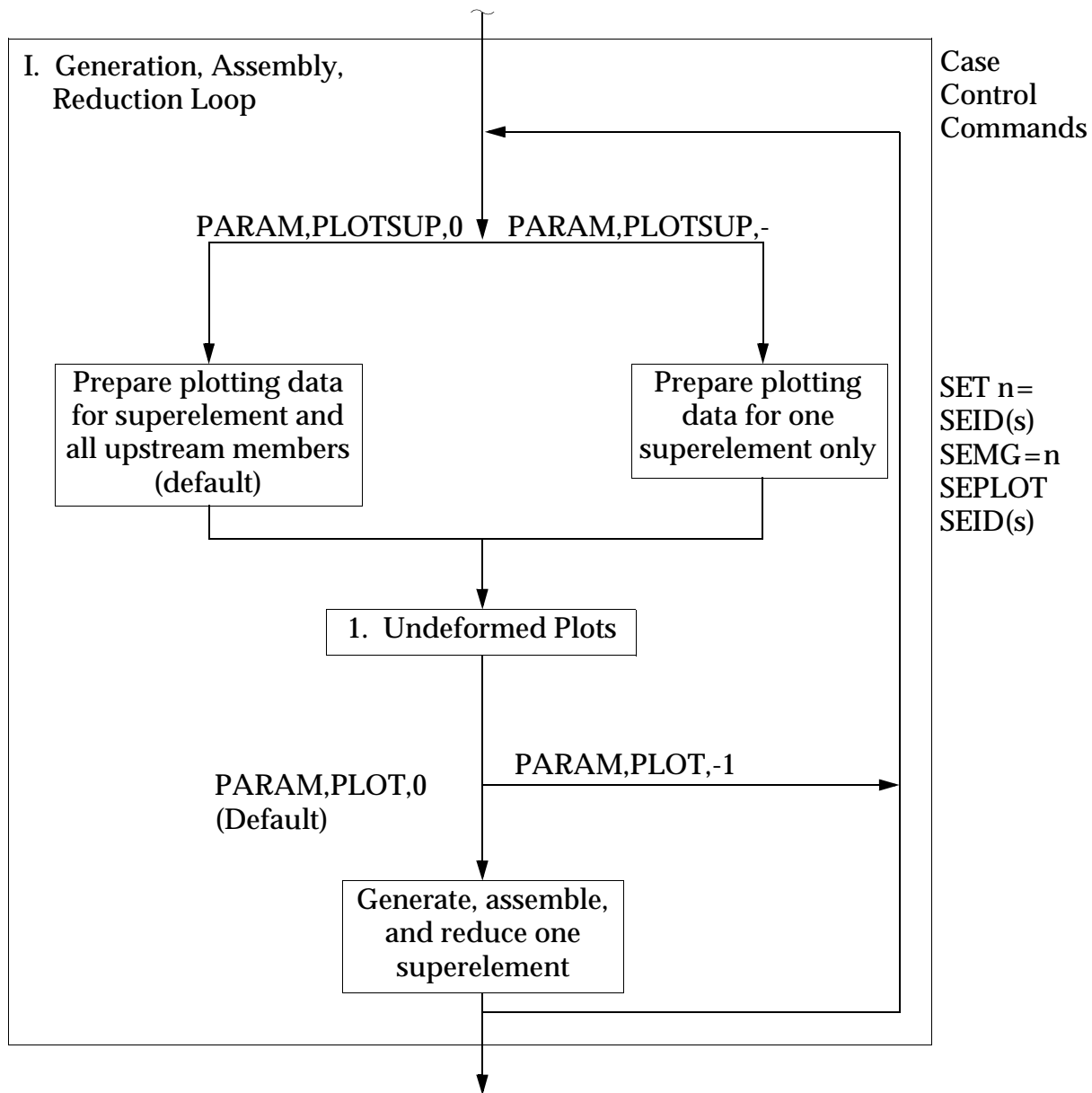


Figure 11-1 Phase 0 and I Superelement Plot Control In the Structured Solution Sequences



(Continued on [Figure 11-3](#))

Figure 11-2 Phase I Superelement Plot Control In the Unstructured Solution Sequences

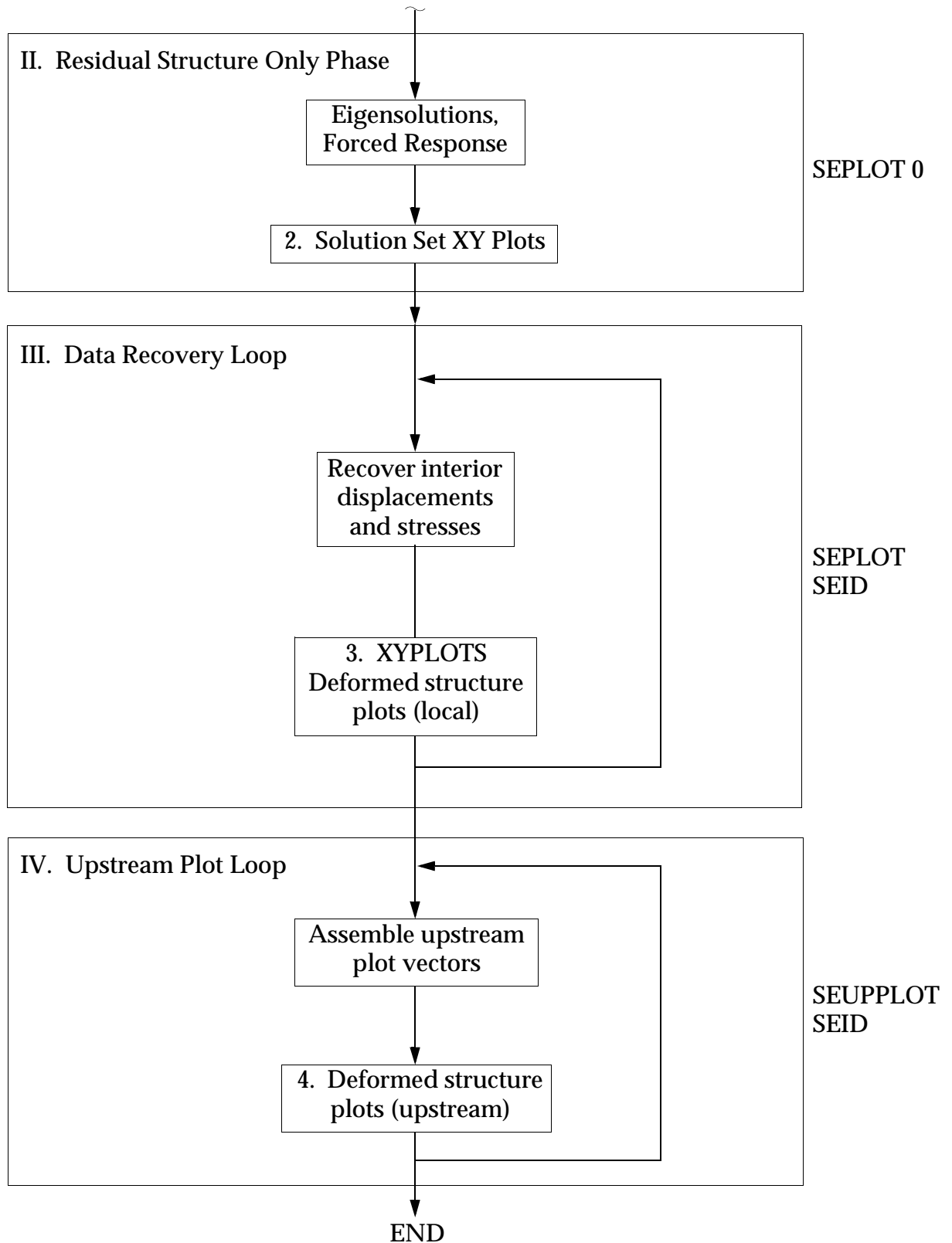


Figure 11-3 Phase II, III, and IV Superelement Plot Control in All Solution Sequences

11.1.1 Structure Plotter Coordinate System and Orthographic Projection

In order to define the coordinates of an orthographic projection of the structural model, an R, S, T plotter coordinate system is defined as shown in [Figure 11-4](#).

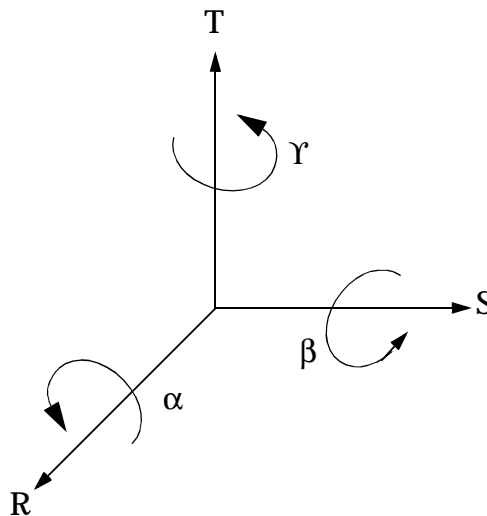


Figure 11-4 Plotter Coordinate System

The S-T plane is taken as the plan of projection. The structural model is defined in the basic coordinate system, which is denoted as the X, Y, Z coordinate system. The user specifies the position of the structural model with respect to the S-T projection plane by the angles γ , β , and α . These angles position the X, Y, Z coordinate systems with respect to the R, S, T coordinate system. The two coordinate systems are coincident for $\gamma = \beta = \alpha = 0$. The sequence in which the rotations are taken is crucial and has been arbitrarily chosen as γ , the rotation about T-axis followed by β , the rotation about the S-axis, followed by α , the rotation about the R-axis. Normally, α is not used since it does not affect the appearance of the S-T projection, but only its orientation on the page.

The orthographic projection is obtained by computing the S and T coordinates of each point having coordinates X, Y, Z from the transformation equation

$$\begin{Bmatrix} R \\ S \\ T \end{Bmatrix} = [A_\alpha][A_\beta][A_\gamma] \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix} \quad \text{Eq. 11-1}$$

where:

$$[A_\gamma] = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \text{Eq. 11-2}$$

$$[A_\beta] = \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix}, \quad \text{Eq. 11-3}$$

and

$$[A_\alpha] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad \text{Eq. 11-4}$$

In order to illustrate clearly the orientation process, **Figure 11-5** shows a rectangular parallelepiped as it is rotated through the γ , β , α sequence. The final S-T plane shown in **Figure 11-5d** contains the desired orthographic projection.

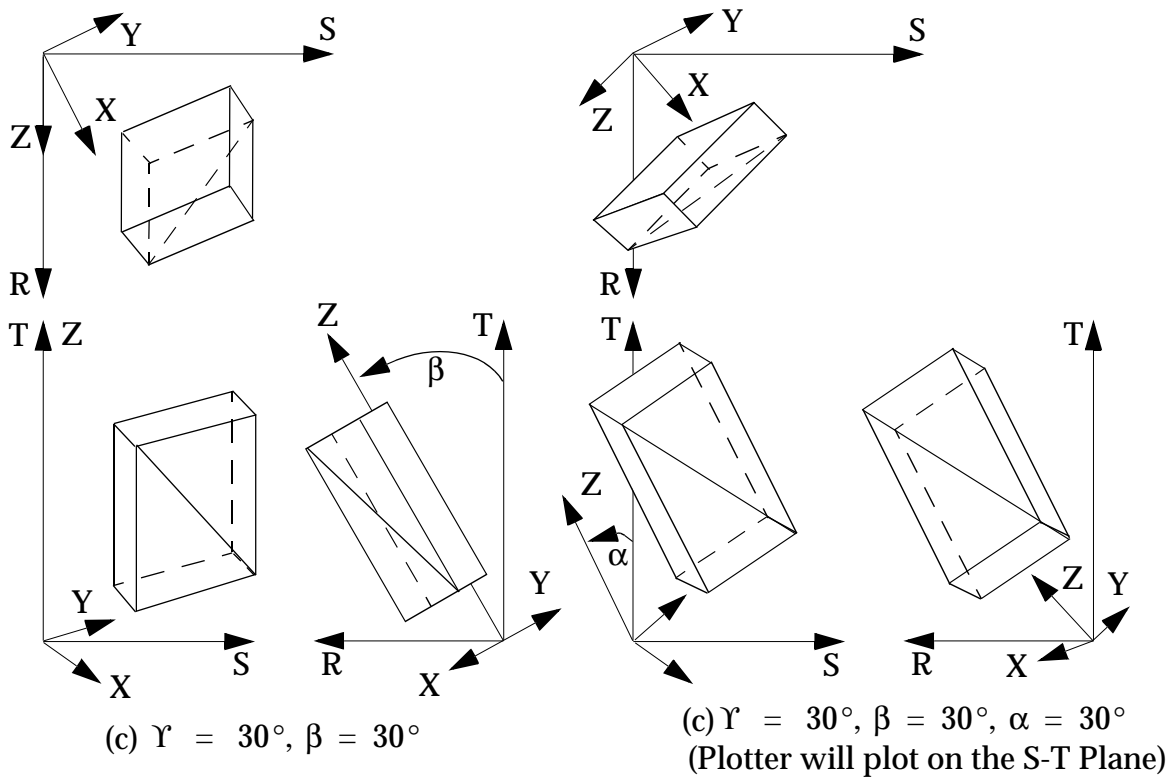
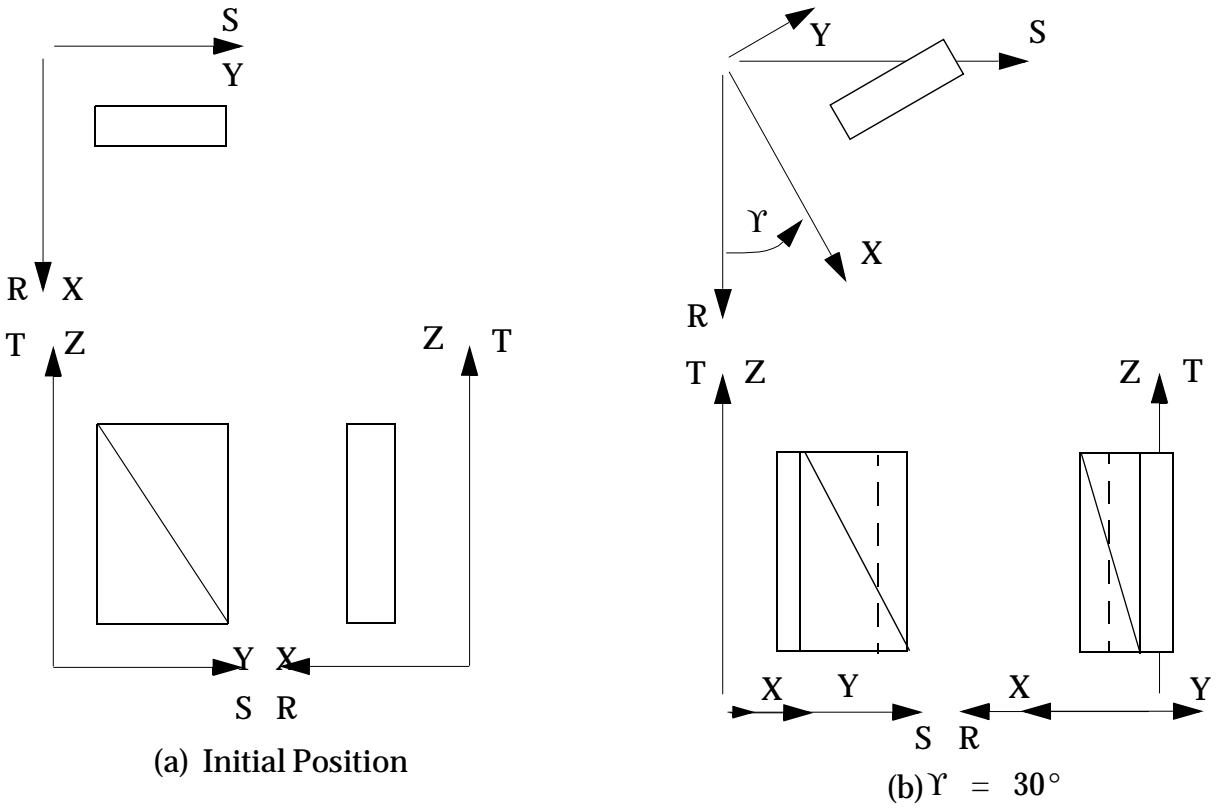


Figure 11-5 Plotter -- Model Orientation.

11.1.2 Perspective Projection

In addition to the three angular relationships required for orthographic projection, the perspective projection requires knowledge of the vantage point in the R-S-T system (i.e., the three coordinates of the observer), and the location of the projection plane (plotter surface). The vantage point is selected by the user (or automatically by the program), and lies in the positive R half space as shown in [Figure 11-6](#). The projection plane is chosen to lie between the observer and the S-T plane.

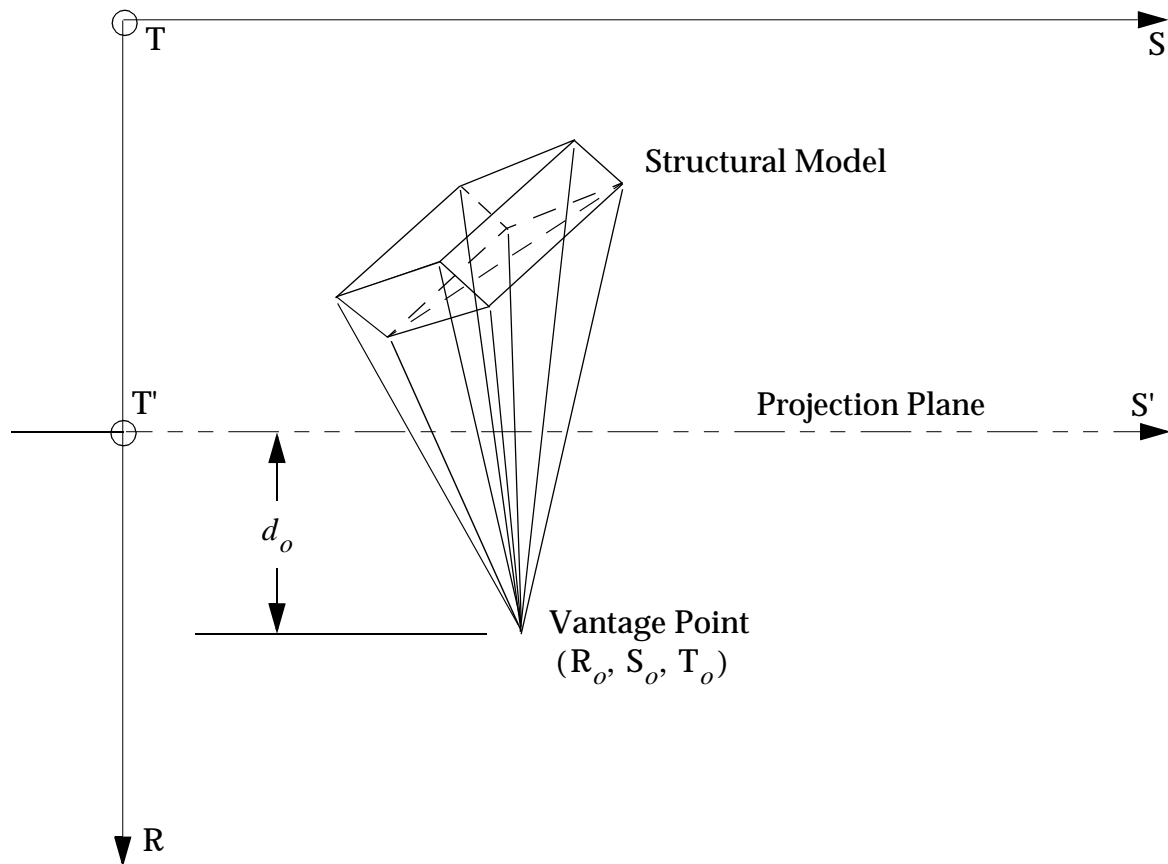


Figure 11-6 Perspective Projection Geometry

For each point, the coordinates S' and T' on the projection plane (see [Figure 11-6](#)) are obtained from the orthographic projection coordinates R, S, T by

$$\begin{Bmatrix} S' \\ T' \end{Bmatrix} = \begin{Bmatrix} S_o \\ T_o \end{Bmatrix} + \frac{d_o}{R_o - R} \begin{Bmatrix} S - S_o \\ T - T_o \end{Bmatrix} \quad \text{Eq. 11-5}$$

where R_o, S_o, T_o are the coordinates of the vantage point and d_o is the separation distance between the vantage point and the projection plane.

11.1.3 Stereoscopic Projection

The stereoscopic effect is obtained through the differences in images received by the left and right eyes. Each is a perspective image, but with a different vantage point. The two vantage points are separated horizontally by 70 mm (2.756 inches), the nominal ocular separation standard used in commercially available stereoscopic cameras and viewers. Two plots are produced for viewing with a stereoscopic viewer.

11.1.4 Projection Plane to Plotter Transformations

Since the plotter surface is defined differently for each of the plotters used in MSC.Nastran, and since it is desirable to minimize the amount of special coding for each plotter, a common interface with the plotter routines is provided wherein the plotted surface is assumed to have a lower left corner defined by x, y coordinates (0,0) and an upper right corner defined by x, y coordinates (1,1). Plotter utility routines are utilized to convert from this system to each individual plotter.

Since the coordinates of points in any of the projection planes previously discussed may have arbitrary numerical values, a linear mapping of the form

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = [A_{ij}] \begin{Bmatrix} S \\ T \end{Bmatrix} + \begin{Bmatrix} x_o \\ y_o \end{Bmatrix} \quad \text{Eq. 11-6}$$

is used, where the transformation constants $[A_{ij}]$ and x_o, y_o are determined as shown below so as to fill that portion of the x, y space required by the user or automatically selected by the program. In the case of a perspective projection, $\{S\}$ and $\{T\}$ are replaced by $\{S'\}$ and $\{T'\}$. Since we desire no distortion of the plotted object and the orientation has already been specified, we set

$$[A_{ij}] = A \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{Eq. 11-7}$$

Let

$$M = \frac{1}{2}[(T_{max} - T_{min}) - (S_{max} - S_{min})] \quad \text{Eq. 11-8}$$

Then, to fill the available x, y space we require, for $M > 0$,

$$\begin{aligned}
 0 &= AT_{min} + y_o \\
 1 &= AT_{max} + y_o \\
 m &= AS_{min} + x_o \\
 1 - m &= AS_{max} + x_o
 \end{aligned}
 \tag{Eq. 11-9}$$

or, for $M < 0$

$$\begin{aligned}
 m &= AT_{min} + y_o \\
 1 - m &= AT_{max} + y_o \\
 0 &= AS_{min} + x_o \\
 1 &= AS_{max} + x_o
 \end{aligned}
 \tag{Eq. 11-10}$$

where m is the horizontal margin for $M > 0$ and the vertical margin for $M < 0$.

From [Eq. 11-9](#) and [Eq. 11-10](#) the quantities m , A , x_o , y_o , and thus the plotter coordinates, are determined.

11.1.5 Deformation Scaling

In plotting deformed structures, the components of displacement are added to the coordinates of the undeformed grid points to obtain the coordinates of the grid points of the deformed structure. Since the numerical magnitude of the physical structural displacements is usually much smaller than the size of the structure itself, additional scaling must be performed on the displacement vectors in order to obtain a viewable plot. In MSC.Nastran this is done by the user who specified a value for the magnitude of the maximum structural deflection in units of length of the undeformed structural model. Thus, if $\max \{T_{max} - T_{min}, S_{max} - S_{min}\}$ were, say, 1000 units, a specification of 50 would result in a plotted maximum deformation equal to 5% of the maximum plot size. In addition, the scale of the deformed structure, described in “[Projection Plane to Plotter Transformations](#)” on page 511, is reduced by 5% to accommodate the deformation vectors.

11.2 Postprocessors

The plotps postprocessor reads plotting commands from a single MSC.Nastran binary or neutral format plot file and produces a file which can be printed or viewed on a PostScript device. Each image will be oriented to best correlate the plot's page size with the printable page size (assumed to be 7.5 inches wide and 10.0 inches tall). Command line arguments can be specified in any order, command keywords can be abbreviated by truncation. These command keywords are discussed in “**PLOTPS**” on page 232 of the *MSC.Nastran 2004 Installation and Operations Guide*.

On UNIX and VMS systems, the plot file type, '.plt' for binary files and '.neu' for neutral files, may be omitted.

Examples

- Translate a binary format plot file into PostScript:
plotps example.plt
- Translate a neutral format plot file into PostScript:
plotps example.neu

CHAPTER
12

Interface With Other Programs

- Introduction
- POST Case Control Command
- MSC.Patran Support of Heat Transfer Analysis

12.1 Introduction

MSC.Nastran provides for direct support of interface to MSC's interactive graphics products, MSC.Access, MSC.Patran, and SDRC I-DEAS programs, and to other outside vendor programs.

For MSC.Patran, the support is provided via the MSC.Nastran DBC module, which creates a "graphics" database. The interface to the DBC module is described in the "**POST**" on page 715 of the *MSC.Nastran Quick Reference Guide*.

The graphics database created by the MSC.Nastran DBC module, can also be read by the MSC.Access library of object routines. This object library can be linked with a user-created program that extracts data from the database. This is an indirect method for outside vendors to obtain MSC.Nastran model and results information. See the *MSC.Access Reference Manual* for more information.

Communication with MSC.Patran, SDRC I-DEAS, LMS International's MSC_NF interface, and Dynamic Design Solutions' SYSTUNE interface, is through the MSC.Nastran OUTPUT2 module, which creates a FORTRAN readable file that is converted by these outside programs to their own particular data formats. The user interface and the supported data for these outside programs are described in "**POST**" on page 715 of the *MSC.Nastran Quick Reference Guide* (-1, -2, -4, and -5).

Users can also add their own features or modifications to MSC.Nastran. This capability is possible for users who have requested the "User Modifiable" version of MSC.Nastran. (See the *User Modifiable MSC.Nastran User's Guide*).

12.2 POST Case Control Command

Introduction

Finite element analysis of large structural components sometimes requires the consideration of many different loading scenarios to complete the simulation. This can be especially true for static analysis, which may require several thousand different load cases to be evaluated. During this process, recovery and post-processing of large amounts of response data usually occurs. Many analysis organizations use commercially available pre-/post-processors, like MSC.Patran, to graphically display results data. These programs typically read the results data from a fortran file generated by the OUTPUT2 module in MSC.Nastran. This file contains all of the output requested by the case control data recovery requests in each and every subcase. Occasionally, output data is required for a subcase that does not need to be handled by the post-processing program.

Theory

The amount of data recovered for each subcase is dependent upon the case control commands that are present. If a request is placed above all subcases, that request applies to all of the subcases. Data is generated and produced one time for all subcases by examining the case control requests in each subcase. Each separate type of response is stored in its own data block. For example, displacement results are computed and stored separately from element stress results. These data blocks will include results for any subcase that contains a data recovery request with a print, plot or punch destination. The data blocks are then written to an external file by the OUTPUT2 module. There is no way to control the placement of the results data on the output file except through the case control request for that data. For example, if there are three subcases and displacement results for the first and third subcases are required for post-processing, but not the second subcase, one could simply request displacement output for only the first and third subcases. However, if punched displacement results for the second subcase were also required, displacements for all three subcases would be placed on the output file, even though results for the second subcase are not needed nor wanted in the post-processor. This applies to all output quantities requested by case control commands. There is no way to handle this situation except to run the analysis twice and change the data recovery requests. A simple method is needed to exercise more control over the contents of the output file.

The new POST Case Control command gives the user more control over the contents of the output file produced by the MSC.Nastran program when a PARAM,POST Bulk Data entry is present in the input. It does not affect the contents of the results data blocks, only the amount of data that is transferred from those data blocks to the output

file. This is accomplished by giving the OUTPUT2 module access to the case control requests for each subcase. The POST command is examined for the output requested in each subcase. Only that output requested by the POST command is placed on the output file for the subcase. Note that the data must be made available by the presence of a data recovery command in the subcase. For example, the POST command cannot output stress data if no stress data has been requested.

The operations associated with the POST Case Control command are closely related to the operations described under the PARAM POST discussion in “[Parameters](#)” on page 601 of the *MSC.Nastran Quick Reference Guide*. All of the DMAP parameters discussed in the QRG related to the POST parameter are honored and can be used to control the output produced. The POST parameter value itself is supplied by the POST Case Control command through selection of one of the supported post-processor programs shown in the following table.

ppname	Product	PARAM,POST,Value
PATRAN	MSC/PATRAN V3	-1
SDRC	SDRC IDEA-S	-2
NF	MSC/LMS NF	-4
FEMTOOLS	DDS/FemTools	-5
UNIGRAHICS	EDS/Unigraphics	-6

Inputs

The POST Case Control command is designed to give the user a way to limit the output of data recovery results for a particular subcase. The general format of the POST command is:

```
POST [ ppname ] [ TOUNIT furn ] oplist
```

The complete description of the POST command can be found in “[Statements, Commands, Entries, and Parameters](#)” on page 621. The POST command recognizes only a sub-set of all of the results data that can be produced by MSC.Nastran. This sub-set is summarized in the following table.

Output Item	oplist Keyword	Case Command
Displacements	[NO]DISPLACE	DISP
Forces of Single Point Constraint	[NO]SPCFORCE	SPCFORCE
Element Forces	[NO]FORCES	ELFO/FORCE
Element Stresses	[NO]STRESS	ELST/STRESS
Element Strain Energy	[NO]ESE	ESE
Grid Point Force Balance	[NO]GPFORCE	GPFORCE
Stress at Grid Points	[NO]GPSIGMA	STRESS
Strain/Curvature at Grid Points	[NO]GPEPSILON	STRAIN
Composite Element Failure Indices	[NO]PLYFAILURE	STRESS
Element Kinetic Energy	[NO]EKE	EKE
Element Energy Loss	[NO]EDE	EDE
Multi-point Constraint Forces	[NO]MPCFORCE	MPCFORCE
Composite Lamina Stresses	[NO]PLYSIGMA	STRESS
Composite Lamina Strains	[NO]PLYEPSILON	STRAIN
Element Strains	[NO]STRAIN	STRAIN
Grid Point Stresses	[NO]GPSTRESS	GPSTRESS
Grid Point Strains	[NO]GPSTRAIN	GPSTRAIN
Applied Loads	[NO]LOAD	OLOAD
No items to be output	NONE	-----

The POST command permits the optional selection of the target post-processor program as well as the target fortran file unit reference number where the output will be stored.

Outputs

There are no special output considerations when using the POST Case Control command.

Guidelines and Limitations

The new POST Case Control command is intended primarily for static analysis. It is considered to be an extension of the parameters associated with the POST DMAP parameter discussed in “[Parameters](#)” on page 601 of the *MSC.Nastran Quick Reference Guide*. The POST command recognizes specification of any of the commercial post-processor products currently supported by the MSC.Nastran solution sequences. The following guidelines should be kept in mind when using the POST command.

- Use of the POST command is entirely optional. If it is not used and a PARAM,POST,x entry is present in the input file, a post-processor output file is generated just as in previous releases of the program.
- The default post-processor program is MSC.Patran Version 3.0.
- If any POST command operations are to be performed, a POST command must be specified above all subcases.
- When a POST command is encountered, a “PARAM,POST,x” entry is inserted in the Case Control at that point. All other POST-related DMAP parameters are available to the user to control output of data as before.
- Once a particular post-processor program is specified, it cannot be changed in any subsequent subcases.
- The POST command does not allow a change in specification of the output fortran unit reference number if it is associated with a form=formatted OUTPUT2 file.
- POST command options are limited to SORT1 formatted output data at the present time.
- As is presently the case, only data that has been generated by the presence of a case control data recovery request command will be available for post-processing output.
- There is no control over any geometry data output or over any data blocks that are not generated from a case control request. These data blocks will continue to be placed on the OUTPUT2 unit specified by the OUNIT2 parameter depending upon the value of the POST parameter (and the values of any related parameters).
- The placement of the POST command above the subcase level causes a cumulative effect on POST commands in subsequent subcases. Any options specified above the subcase level propagate down into the POST command within a subsequent subcase. Thus, if a POST command specifies NODISP (no displacement output wanted) above the subcase level, then a POST

command with the DISP option would be required within a subcase to produce any output on the OUTPUT2 file for displacements. This also implies that changing the OUTPUT2 file unit reference number with the TOUNIT option in a subcase causes **all** output quantities currently scheduled for output to be switched to the new unit number, not just those in the option list for the current POST command.

Demonstration Example

A simple model is presented to demonstrate the usage of the POST Case Control command to eliminate unwanted output from being placed on the post-processor data file generated by MSC.Nastran. The model data itself is not important for this example. It is the subcase structure that is the essence of the discussion. As an example of how this new feature might be used, consider the case of a static simulation performed on a model requiring three load cases. The output required is displacements at three grid points in the model. Furthermore, the output for the second load case must also include punched displacement data for all grid points. Displacement results for subcases 100 and 300 are to be placed on the default OUTPUT2 file for use by the MSC.Patran post-processor. No output for subcase 200 is required for post-processing by MSC.Patran. The subcase structure might look something like:

```

SET 1000 = 10, 11, 12
DISP = 1000
$
POST          $ use all defaults: Patran v3.0, unit=12
$
SUBCASE       100
LOAD = 100
$
SUBCASE       200
LOAD = 200
DISP(PUNCH) = ALL
POST NODISP  $ stop any displacement output from going to POST file
$
SUBCASE       300
LOAD = 300

```

In this example, the presence of the POST command above all of the subcases indicates that all output requested (DISP for SET 1000 is the only output requested and generated) is to be stored on the default OUTPUT2 fortran unit for use by MSC.Patran V3.0. The presence of the POST command in SUBCASE 200 with the NODISP option prevents the large output produced by the DISP(PUNCH)=ALL request in SUBCASE 200 from being placed on the OUTPUT2 file.

Model Description

The model used for this example is very simple as the intent is to focus on the subcase structure. It is a cantilever plate structure consisting of only six CQUAD4 shell elements and fourteen grid points. There are three subcases used to apply loads and recover displacements.

Input File

```

$
$*****
$
$Minor Enhancement A01130/A01131 - new POST case control capability
$VERSION: 2004
$TEST DECK NAME: postrnex.dat
$
$PURPOSE:
$ Simple case control setup to demonstrate usage of the new POST Case
$ Control Command for the Release Notes
$
$DESCRIPTION:
$ An OP2 file is to be generated for the MSC.Patran V3.0 post-
$ processor. Three subcases are present in case control.
$ Displacement output is requested above the subcase level.
$ Displacements for subcases 100 and 300 are to be placed on the
$ default OP2 file. Displacements for subcase 200 are to be placed on
$ system punch file but not on the OP2 file.
$
$EXPECTED RESULTS:
$ The following UIM 4114 should be present in the .f06 output:
$
$*** USER INFORMATION MESSAGE 4114 (OUTPX2)
$ DATA BLOCK OUGV1 WRITTEN ON FORTRAN UNIT 12, TRL =
$ 101 0 160 0 0 0 0
$ (MAXIMUM POSSIBLE FORTRAN RECORD SIZE = xxxxx WORDS.)
$ (MAXIMUM SIZE OF FORTRAN RECORDS WRITTEN = 146 WORDS.)
$ (NUMBER OF FORTRAN RECORDS WRITTEN = 30 RECORDS.)
$ (TOTAL DATA WRITTEN FOR DATA BLOCK = 381 WORDS.)
$ (DATA FOR 2 CASE(S) WAS OUTPUT TO THIS UNIT. CASE ID LIST FOLLOWS.)
$ 100 300
$
$*****
$
ID MSC,POSTRNEX
SOL 101
TIME 5
CEND
TITLE = POST CASE CONTROL COMMAND EXAMPLE FOR RELEASE NOTES
SUBTITLE = DEMONSTRATE POST COMMAND USAGE
$
SPC = 1
SEALL = ALL

```

```

SET 1000 = 10,11,12
DISP    = 1000
$
$ place post command above subcase level.  Nothing specified causes
$ all defaults to be taken:
$ POST PATRAN TOUNIT 12 (all outputs requested via CC commands)
$ or, for the example here where only DISP is requested,
$ POST PATRAN TOUNIT 12 DISP
$ Note that POST command options specified above the subcase level
$ flow down into all of the subcases.
$
POST  patran tounit 12
$
$-----
subcase 100
LOAD = 100
$-----
subcase 200
LOAD = 200
DISP(PUNCH)=ALL    $ request punched displacement output for all grids
POST NODISP        $ no displacement output to default OP2 file.
$-----
subcase 300
LOAD = 100
$-----
BEGIN BULK
CQUAD4  1      1      1      4      5      2
CQUAD4  2      1      5      6      3      2
CQUAD4  3      1      4      7      8      5
CQUAD4  4      1      8      9      6      5
CQUAD4  5      1      7     10     11     8
CQUAD4  6      1     11     12      9      8
$
PSHELL  1      100     0.001   100
MAT1    100     10.7+6     0.33
$
FORCE   1      10      1.      10.
FORCE   1     11      1.      20.
FORCE   1     12      1.      10.
GRID    1      0.      1.75   0.
GRID    2      0.      2.275  0.
GRID    3      0.      2.8     0.
GRID    4      0.6     1.75   0.
GRID    5      0.6     2.275  0.
GRID    6      0.6     2.8     0.
GRID    7      1.2     1.75   0.
GRID    8      1.2     2.275  0.
GRID    9      1.2     2.8     0.
GRID   10      1.8     1.75   0.
GRID   11      1.8     2.275  0.
GRID   12      1.8     2.8     0.
GRID   13      0.0     1.75   1.
GRID   14      1.0     1.75   1.
LOAD   100     1.      1.      1

```

```
LOAD    200    1.    2.    1
LOAD    300    1.    3.    1
PARAM   AUTOSPC YES
SPC     1      1      13456
SPC     1      2      123456
SPC     1      3      13456
ENDDATA
```

12.3 MSC.Patran Support of Heat Transfer Analysis

MSC.Patran offers a comprehensive and integrated graphics interface to the linear, nonlinear, steady-state, and transient thermal analysis capabilities in MSC.Nastran. From within the MSC.Patran environment, you can completely define your thermal analysis model, submit it to MSC.Nastran for analysis, import MSC.Nastran thermal analysis results into the MSC.Patran database, and visualize the results. Complete support is provided for analyses including conduction, convection, advection, and radiation.

The interface is designed for both new and advanced users of MSC.Nastran and MSC.Patran. The online help system provides several guided example problems and describes all of the MSC.Patran menu interface to MSC.Nastran's thermal solver. Basic and Advanced modes are included for most of the menu forms for thermal loads and boundary conditions. For example, the Basic convection form only requires you to define the convection coefficient and ambient fluid temperature, while the Advanced form includes options for a user-defined convection equation, time-dependent and temperature-dependent coefficients, and variable ambient temperatures.

A brief overview of capabilities and limitations is given in the sections that follow.

12.3.1 Loads and Boundary Conditions

The following thermal loads and boundary conditions are supported, each of which can vary both spatially and temporally:

- Prescribed nodal temperatures
- Heat flux normal to a surface
- Vector heat flux from a distant radiant heat source
- Heat applied directly to a node (nodal source)
- Volumetric heat generation within conduction elements
- Basic convection
- Advection with forced convection boundaries
- Radiation exchange with space
- Radiation exchange within enclosures (cavity radiation)
- Multiple enclosures

12.3.2 Results Processing

The following results quantities can be postprocessed within MSC.Patran:

- Nodal temperatures
- Applied heat loads
- Heats of constraint for temperature boundary conditions
- Heat fluxes within conduction elements
- Heat fluxes at boundaries due to applied heat loads, free/forced convection, advection, and radiation
- Temperature gradients
- Enthalpies
- Rate of change of enthalpies

12.3.3 Limitations

When using MSC.Patran to pre- and postprocess heat transfer analyses in MSC.Nastran, keep in mind the following limitations:

1. There is currently no support of MSC.Nastran heat transfer Bulk Data entries through MSC.Patran's NASTRAN Input File Reader.
2. Emissivity as a function of wavelength is currently not supported. The RADBND Bulk Data entry is used to specify the wavelength break points for radiation problems.
3. Automatic restart is not supported for the MSC.Patran interface to MSC.Nastran heat transfer analysis; this includes:
 - Restarting from a previously converged time step (SOL 159 to SOL 159)
 - Using a steady-state analysis as a restart to a transient analysis (SOL 159 from SOL 153)
 - Using a steady-state analysis as a restart to a thermal stress analysis (SOL 106 from SOL 153)
 - Using various time steps from a transient thermal analysis as a restart to a quasi-static thermal stress analysis (SOL 106 from SOL 159)

To perform many of the restarts listed above, you must modify the cold start run and manually insert the bulk data entries for the restart.

4. The global view factor is currently not supported. Specifically, the RADCAV Bulk Data entry, SETij, are not supported. Basic view-factor calculations are supported.
5. NOLIN Bulk Data entries are not supported. The NOLIN entries are used for various thermal-state control functions (e.g., the nonlinear transient power function).
6. The TLOAD2 Bulk Data entry is not supported. This entry enables you to specify various functions such as SIN, COS, and EXP when applying a thermal load. One work-around is to use the Field capability to create these functions. For example, you can create a PCL (PATRAN Command Language) function that can then be mapped to a table using the Create/Non-Spatial/Tabular Input field capability.
7. The ambient element (ELEAMB) field for the RADCAV Bulk Data entry is not supported. For radiation enclosure problems, this means that for an incomplete cavity, all energy exchange carried on outside the enclosure surface is lost to zero-degree space. MSC.Patran will not automatically generate an ambient element that could have a nonzero temperature and a distinct area associated with it.

12.3.4 Documentation

For detailed documentation, refer to the *MSC.Patran MSC.Nastran Preference Guide, Volume 2: Thermal Analysis*. This document is available online as part of the help system for MSC.Patran.

CHAPTER
13

Additional Topics

- Introduction
- Composites
-
- Prerelease of the General Nonlinear Solution Sequence SOL 400
- MSC.Nastran Implicit Nonlinear -- SOL 600
- Coupled Fluid-Structure Interaction
- Acoustic Cavity Modeling
- Cyclic Symmetry
- p-Elements
- Response Spectrum Analysis
- Viscoelastic Materials in Frequency Response Analysis
- Introduction to Parallel Numeric Enhancements
- MSC.Nastran/MSC.ADAMS Interface
- Rotordynamics
- References

13.1 Introduction

This chapter contains the documentation and supporting material for new and existing capabilities that are not yet covered in other MSC.Nastran documentation (see “[List of MSC.Nastran Books](#)” on page xii for an overview of MSC.Nastran documentation).

The following topics are presented in this chapter:

- Composites data recovery
- Nonlinear topics
- Coupled fluid-structure interaction
- Acoustics
- Cyclic symmetry
- p-element application examples
- RF Alters for special applications and capabilities
- Design sensitivity analysis
- Response spectrum analysis
- Viscoelastic materials in frequency response
- Introduction to Parallel Numeric Enhancements
- MSC.Nastran/MSC.ADAMS Interface
- Rotordynamics
- References

13.2 Composites

13.2.1 Introduction

The CQUAD4, CQUAD8, CTRIA3, and CTRIA6 surface elements can be used to model laminated composites in finite element models. Laminated composites may be conceptually viewed as a "stack" of laminae with different orientations of the principal material directions in the individual lamina. An exploded view of three cross-ply laminated plates is illustrated in [Figure 13-1](#). The n -laminae ($n = 1,2,3,4$) of each of the three configurations are normal to the z -axis of the indicated coordinate system and the 1- and 2-axes appended to the individual lamina denote principal material axis directions. The directions of the principal material axes of each lamina alternate as implied by the use of the word "cross-ply" to describe the configuration. The xy -plane of the coordinate axes is defined in the geometric middle plane of the laminae.

An entire stack of laminae may be modeled with a single plate or shell element because the material properties of the "stack" are completely reflected in the matrices of elastic moduli for the element. These matrices are automatically calculated from user-supplied definitions of the thickness, the material properties, and the relative orientation of these properties for the individual lamina. Once these matrices of elastic moduli are calculated, the analysis proceeds in a standard manner as described in "[Solution Sequences](#)" on page 341. This capability for the automatic representation of laminated composites is available in linear static analysis, real and complex eigenvalue analysis, buckling analysis, and dynamic analysis.

In static analysis and in static analysis with cyclic symmetry, the analyst has the option to request that the program evaluate stresses or strains and appropriate failure indices in individual laminae. Interlaminar shear stresses and strains and a failure index for bonding may also be requested.

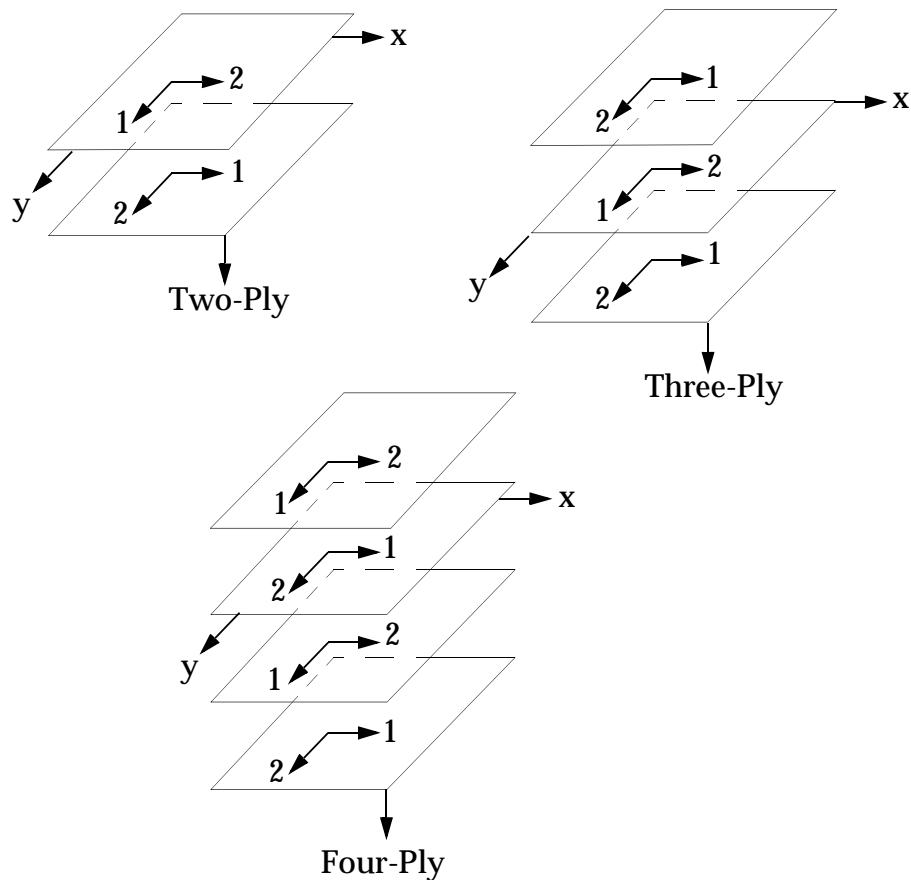


Figure 13-1 Exploded View of Three Cross-Ply Laminated Plates

Layered Composite Material Analysis

Two Bulk Data entries, MAT8 and PCOMP, are used in the modeling of composite structures. The MAT8 Bulk Data entry (defined in “[Nonuniform Material and Composite Properties for Shells](#)” on page 280) defines the material property for an orthotropic material. The material properties of the individual lamina along the longitudinal direction (also called the fiber direction or 1-direction) and the transverse direction (also called the matrix direction or 2-direction) are supplied using this entry. The PCOMP Bulk Data entry defines the properties of the n-ply composite material laminate. The thickness, orientation, and the material identification number of each of the individual lamina is provided using this entry.

From the data supplied by the PCOMP and MAT8 Bulk Data entries, the program automatically calculates the membrane, bending, transverse shear, and coupled membrane-bending material properties of the laminate as a whole. These calculated properties are output in the form of PSHELL and MAT2 entries when a sorted echo of the Bulk Data Section is requested.

The coefficients of thermal expansion derived for membrane-bending coupling, which appear in the A1, A2, and A12 fields of MAT2 entry and correspond to the MID4 Field on the PSHELL, require special interpretation. They are given by:

$$\{\alpha_{MAT2}\} = [G_{ij_{MAT2}}]\{\alpha_{ACTUAL}\} \quad \text{Eq. 13-1}$$

In order to obtain the actual values of A1, A2, and A12 $\{\alpha_{MAT2}\}$, the user must solve [Eq. 13-1](#).

If the user specifies ECHO = PUNCH in the Case Control Section, the generated PSHELL and MAT2 entries are routed to the PUNCH file, where they are written after the sorted bulk data echo. Since the MIDi fields use more than eight digits, a large field format is used for the PSHELL and MAT2 entries. The PUNCH file thus contains the sorted Bulk Data echo and the generated PSHELL and MAT2 entry images. After the MAT8 and PCOMP entry images are deleted, the entry images contained in the PUNCH file may be included in the Bulk Data to be submitted in a subsequent run. The existence of composite elements is recognized by a large value (more than eight digits) on the MIDi fields of the PSHELL entry, which may be generated by PCOMP entries. If stress output is desired, ensure that PARAM,NOCOMPS is set to -1 for any subsequent runs because composite stress recovery is not possible without the PCOMP entries.

Certain postprocessing features are available for static analysis using SOLutions 24, 38, 47, and superelement Solution Sequences 21, 61, 62, 63, 65, 66, 81, 82, 83, 91, 101, 103, 105, 106, 114, 115, and 153, which include all the statics, normal modes, steady nonlinear heat transfer, and buckling analyses. RFALTERs, RF24D79 and RF47D79, are required in SOLutions 24 and 47. PARAM,NOCOMPS may be used to modify these post processing options in superelement solution sequences. These features are (1) stresses in individual laminae including interlaminar shear stresses and (2) a failure index for each lamina.

It is to be noted that two-dimensional plate theory used in the CQUAD4, CQUAD8, CTRIA3 or CTRIA6 elements does not allow for the exact calculation of interlaminar stresses. An approximate procedure is therefore used to determine the interlaminar shear stresses. The approximation consists of neglecting the effects of twisting moments in the computation of interlaminar shear stresses. For the majority of composite material structural analyses, this approximation is expected to yield satisfactory results for all elements in the model except those at the edges of the structure.

Voluminous output will result if the stresses for all elements for all laminae are requested. To help the analyst to review this information, provisions are made to calculate a failure index for each ply. This failure index is obtained by considering the failure criteria for unidirectional fiber composites as in the commonly used failure theories. The user has the option to choose any one of the four commonly used failure criteria for composites. These are: (1) Hill's Theory, (2) Hoffman's Theory, (3) Tsai-Wu Theory, and (4) Maximum Strain. The user must supply allowable stresses/strains for the laminae if a failure index is to be calculated.

13.2.2 Classical Lamination Theory

Classical lamination theory makes the following assumption regarding the behavior of the laminae:

- The laminae are perfectly bonded together.
- The bonds are infinitesimally thin and no lamina can slip relative to another.
- Linear variation of strain through the laminate thickness is assumed.

Deformation in the X-Y plan of the plate at any point C at a distance z in the normal direction to plate middle surface is

$$U = U_0 + z\theta_y \quad \text{Eq. 13-2}$$

$$V = V_0 + z\theta_x \quad \text{Eq. 13-3}$$

where U , v , and w are the displacements along the X, Y, and Z directions in the element coordinate system, and θ_x , θ_y are the rotations.

The strain-displacement-middle surface strain and curvatures relationship is given by:

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial U_0}{\partial x} \\ \frac{\partial V_0}{\partial y} \\ \frac{\partial U_0}{\partial y} + \frac{\partial V_0}{\partial x} \end{Bmatrix} + z \begin{Bmatrix} \frac{\partial \theta_y}{\partial x} \\ -\frac{\partial \theta_x}{\partial y} \\ \frac{\partial \theta_y}{\partial y} - \frac{\partial \theta_x}{\partial x} \end{Bmatrix} = \begin{Bmatrix} \epsilon_x^0 \\ \epsilon_y^0 \\ \gamma_{xy}^0 \end{Bmatrix} - z \begin{Bmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix} \quad \text{Eq. 13-4}$$

where the ϵ^0 's and χ 's are the middle surface strains and curvatures, respectively.

The stress resultants for an N-layer laminate are obtained by integration of the stresses in each lamina through the laminate thickness as:

$$\begin{Bmatrix} N_x \\ N_y \\ N_{xy} \end{Bmatrix} = \int_{-\frac{t}{2}}^{\frac{t}{2}} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} dz = \sum_{k=1}^N \int_{z_{k-1}}^{z_k} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix}_k dz \quad \text{Eq. 13-5}$$

$$\begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = - \int_{-\frac{t}{2}}^{\frac{t}{2}} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} z dz = - \sum_{k=1}^N \int_{z_{k-1}}^{z_k} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix}_k z dz \quad \text{Eq. 13-6}$$

The stress resultant to strain relationship is:

$$\begin{Bmatrix} N_x \\ N_y \\ N_{xy} \end{Bmatrix} = \sum_{k=1}^N [G]_k \left\{ \int_{z_{k-1}}^{z_k} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix}_k dz - \int_{z_{k-1}}^{z_k} \begin{Bmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix}_k z dz \right\} \quad \text{Eq. 13-7}$$

$$\begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \sum_{k=1}^N [G]_k \left\{ - \int_{z_{k-1}}^{z_k} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix}_k z dz + \int_{z_{k-1}}^{z_k} \begin{Bmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix}_k z^2 dz \right\} \quad \text{Eq. 13-8}$$

where $[G]_k$ is the material matrix transformed from the laminate coordinate system into the lamina coordinate system.

These relations can be written in the following form used to describe composite elements:

$$\begin{Bmatrix} F \\ M \end{Bmatrix} = \begin{bmatrix} A & B \\ B & D \end{bmatrix} \begin{Bmatrix} \epsilon^0 \\ \chi \end{Bmatrix} \quad \text{Eq. 13-9}$$

where:

$$[A] = \sum_{k=1}^N [G]_k (z_k - z_{k-1})$$

$$[B] = \frac{1}{2} \sum_{k=1}^N [G]_k (z_k^2 - z_{k-1}^2)$$

$$[D] = \frac{1}{3} \sum_{k=1}^N [G]_k (z_k^3 - z_{k-1}^3)$$

are named in composite element literature as the membrane, membrane-coupling, and bending matrices, respectively.

In the shell element formulation in MSC.Nastran these relationships take the following form:

$$\begin{Bmatrix} F \\ M \\ Q \end{Bmatrix} = \begin{bmatrix} TG_1 & T^2G_4 & 0 \\ T^2G_4 & \frac{T^3}{12}G_2 & 0 \\ 0 & 0 & T_sG_3 \end{bmatrix} \begin{Bmatrix} \varepsilon^0 \\ \chi \\ \gamma \end{Bmatrix} \quad \text{Eq. 13-10}$$

where:

$$[A] = TG_1$$

$$[B] = -T^2G_4$$

$$[D] = \frac{T^3}{12}G_2$$

$$\{Q\} = \begin{Bmatrix} Q_x \\ Q_y \end{Bmatrix} = \text{transverse shear resultants}$$

$$\{\gamma\} = \begin{Bmatrix} \gamma_x \\ \gamma_y \end{Bmatrix} = \text{transverse shear strains}$$

$$T = \text{nominal plate thickness}$$

T_s = effective transverse shear material thickness

G_3 = effective transverse shear material matrix

MSC.Nastran allows G_1 , G_2 , G_4 , T , G_3 , and T_s to be input directly in PSHELL or to have the composite equivalent material matrices calculated internally from the PCOMP data.

Equivalent thermal properties are determined as follows:

$$\begin{Bmatrix} \alpha_{\varepsilon_0} \\ \alpha_{\chi} \\ \alpha_{\chi\varepsilon_0} \end{Bmatrix} = \begin{Bmatrix} T\alpha_1 \\ \frac{T^3}{12}\alpha_2 \\ T^2\alpha_3 \end{Bmatrix} = \begin{bmatrix} A & B & 0 \\ B & D & 0 \\ 0 & 0 & T_s G_3 \end{bmatrix}^{-1} \begin{Bmatrix} G\alpha_{\varepsilon_0} \\ G\alpha_{\chi} \\ G\alpha_{\chi\varepsilon_0} \end{Bmatrix} \quad \text{Eq. 13-11}$$

The following coefficients are used to determine equivalent thermal properties:

$$\{G\alpha_{\varepsilon_0}\} = \sum_{k=1}^N [G]_k \{\alpha\}_k (z_k - z_{k-1})$$

$$\{G\alpha_{\chi}\} = \frac{1}{3} \sum_{k=1}^N [G]_k \{\alpha\}_k (z_k^3 - z_{k-1}^3)$$

$$\{G\alpha_{\chi\varepsilon_0}\} = \frac{1}{2} \sum_{k=1}^N [G]_k \{\alpha\}_k (z_k^2 - z_{k-1}^2)$$

These coefficients are used to calculate the equivalent thermal properties as follows:

$$\{\alpha_{\varepsilon_0}\} = [A]^{-1} \{G\alpha_{\varepsilon_0}\}$$

and

$$\{\alpha_x\} = [D]^{-1}\{G\alpha_x\}$$

where $\{\alpha_{\xi_0}\}$ and $\{\alpha_x\}$ are the membrane and bending equivalent thermal properties.

Note that $\{\alpha_{\xi_0}\}$ is not directly calculated, but is determined from $\{G\alpha_{\xi_0}\}$ in MSC.Nastran when the PCOMP input is used when the MID4 field on the PSHELL is $> 400,000,000$. Also note that $\{G_x\xi_0\}$ cannot be input directly using PSHELL and $\{\alpha_{\xi_0}\}$ can be input only if $[B]$ is invertible (which is generally not true).

The thermal expansion relationships in the shell element formulation take the following form:

$$\alpha_{\varepsilon_0} = T\alpha_1$$

$$\alpha_{\chi} = \frac{T^3}{12}\alpha_2$$

$$\alpha_{\chi\varepsilon_0} = T^2\alpha_3$$

where α_1 , α_2 , and α_3 are the thermal expansion inputs on the materials referenced by the MID1, MID2, and MID4 fields on the PSHELL entry. If PCOMP is used, these relationships are automatically calculated.

The terms G_1 , G_2 , and G_4 are defined by the following integrals:

$$G_1 = \frac{1}{T} \int [G_e] dz$$

$$G_2 = \frac{1}{I} \int z^2 [G_e] dz$$

$$G_4 = \frac{1}{T^2} \int (-z) [G_e] dz$$

Eq. 13-12

The limits on the integration are from the bottom surface to the top surface of the laminated composite. The matrix of material moduli, $[G_e]$, has the following form for isotropic materials:

$$[G_e]_l = \begin{bmatrix} \frac{E}{1-\nu^2} & \frac{\nu E}{1-\nu^2} & 0 \\ \frac{\nu E}{1-\nu^2} & \frac{E}{1-\nu^2} & 0 \\ 0 & 0 & \frac{E}{2(1+\nu)} \end{bmatrix} \quad \text{Eq. 13-13}$$

For orthotropic materials, the matrix, $[G_e]$, is written as follows:

$$[G_e]_0 = \begin{bmatrix} \frac{E_1}{1-\nu_1\nu_2} & \frac{\nu_1 E_2}{1-\nu_1\nu_2} & 0 \\ \frac{\nu_2 E_1}{1-\nu_1\nu_2} & \frac{E_2}{1-\nu_1\nu_2} & 0 \\ 0 & 0 & G_{12} \end{bmatrix} \quad \text{Eq. 13-14}$$

Here, $\nu_1 E_2 = \nu_2 E_1$ in order to satisfy the requirement that the matrix of elastic moduli be symmetric. In general, the analyst may supply element properties with respect to a particular orientation which does not necessarily correspond to the principal material axes. In this case, the analyst must also supply the value of the angle, θ or material coordinate system that orients the element material axis relative to the side G1-G2 of the element. The material elastic modulus matrix is then transformed by the program into the element modulus matrix through the relation

$$[G_e] = [U]^t [G_m] [U] \quad \text{Eq. 13-15}$$

where:

$$[U] = \begin{bmatrix} \cos^2 \theta & \sin^2 \theta & \cos \theta \sin \theta \\ \sin^2 \theta & \cos^2 \theta & -\cos \theta \sin \theta \\ -2 \cos \theta \sin \theta & 2 \cos \theta \sin \theta & \cos^2 \theta - \sin^2 \theta \end{bmatrix} \quad \text{Eq. 13-16}$$

The finite element model for a structure composed of composite materials requires the evaluation of the matrix of elastic moduli for each plate element of the model. The characteristics of the composite media are totally contained in these matrices.

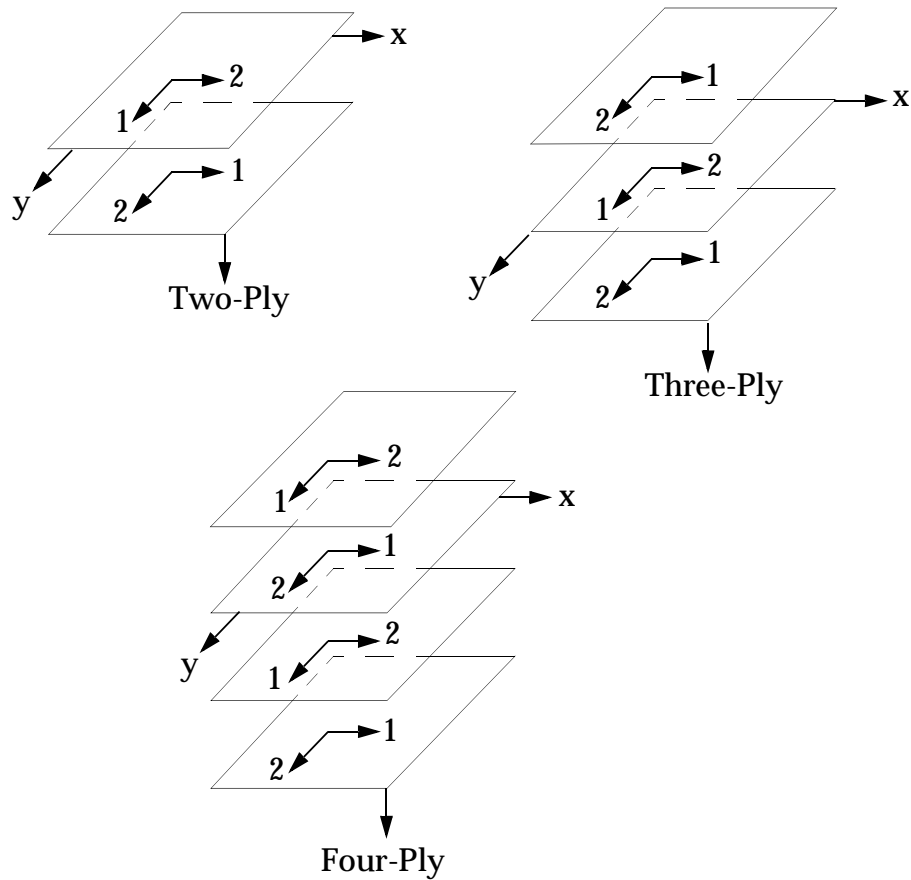


Figure 13-2 Exploded View of Three Cross-Ply Laminated Plates

To illustrate evaluation of these matrices, consider the cross-ply laminates of **Figure 13-2**. In this portion of the discussion, the three configurations shown in **Figure 13-2** are each assumed to be represented by a single quadrilateral plate element and the coordinate axes shown are coincident with the element coordinate axes. Then, if it is assumed that each lamina of the n -ply laminates is of thickness T/n , where T is the total thickness of each of three configurations, the matrices of elastic moduli may be evaluated from the following relations:

$$[G_1] = \frac{1}{T} \left\{ \begin{array}{c} -\frac{T}{2} + \frac{T}{n} \quad -\frac{T}{2} + \frac{2T}{n} \quad \frac{T}{2} \\ \int [G_e]_1 z^2 dz + \int [G_e]_2 dz + \dots + \int [G_e]_n dz \\ -\frac{T}{2} \quad -\frac{T}{2} + \frac{T}{n} \quad -\frac{T}{2} + \frac{(n-1)T}{n} \end{array} \right\} \quad \text{Eq. 13-17}$$

$$[G_2] = \frac{1}{T} \left\{ \begin{array}{l} -\frac{T}{2} + \frac{T}{n} \quad -\frac{T}{2} + \frac{2T}{n} \quad \frac{T}{2} \\ \int [G_e]_1 dz + \int [G_e]_2 z^2 dz + \dots + \int [G_e]_n z^2 dz \\ -\frac{T}{2} \quad -\frac{T}{2} + \frac{T}{n} \quad -\frac{T}{2} + \frac{(n-1)T}{n} \end{array} \right\} \quad \text{Eq. 13-18}$$

$$[G_4] = \frac{1}{T^2} \left\{ \begin{array}{l} -\frac{T}{2} + \frac{T}{n} \quad -\frac{T}{2} + \frac{2T}{n} \\ \int [G_e]_1(-z) dz + \int [G_e]_2(-z) dz \\ -\frac{T}{2} \quad -\frac{T}{2} + \frac{T}{n} \\ \dots + \int [G_e]_n(-z) dz \\ \frac{T}{2} + \frac{(n-1)T}{n} \end{array} \right\} \quad \text{Eq. 13-19}$$

These relations reflect the assumption that the xy -plane of the element coordinate system is coincident with the geometric middle plane of the laminate. The xy -plane of the element coordinate system is defined in the mean plane of the element so that any offset between the mean plane of the connected grid points and the geometric middle plane of the laminate would be reflected in the integration limits of the preceding relations.

13.2.3 Transverse Shear Theory

Note that $I = T^3/12$ in the evaluation of $[G_2]$, i.e., the value $(12I)/\Gamma^2$, will be assigned the default value of 1.0 on the plate element property entry.

The matrix of elastic moduli for transverse shear, $[G_3]_m$ is defined as a two-by-two matrix of the form

$$[G_3]_m = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \quad \text{Eq. 13-20}$$

and the corresponding matrix transformed into an element coordinate system is given by

$$[G_3]_e = [W]^T [G_3]_m [W] \quad \text{Eq. 13-21}$$

where $[W] = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$

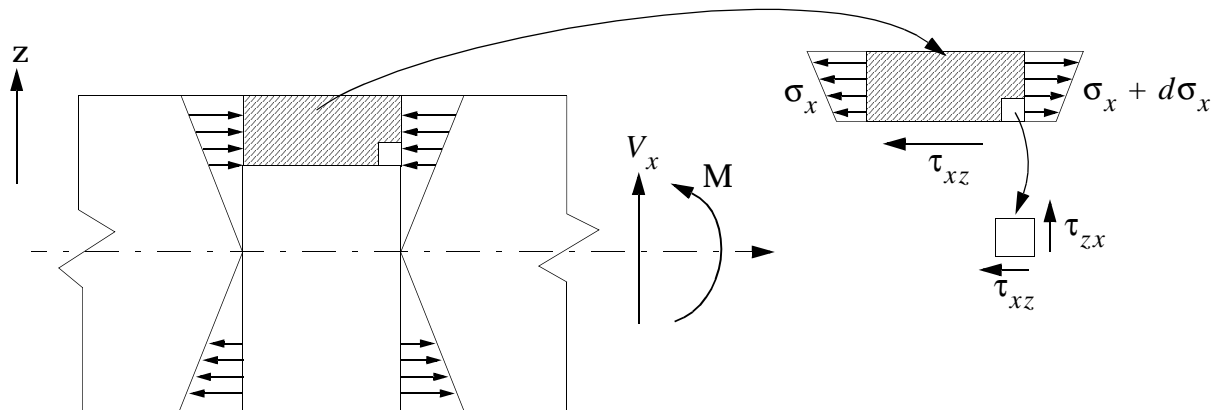
The mean value of the transverse shear modulus \bar{G} for the laminated composite is defined in terms of the transverse shear strain energy, U , through the depth

$$U = \frac{1}{2} \frac{V^2}{\bar{G}T} = \frac{1}{2} \int \frac{(\tau(z))^2}{G(z)} dz \quad \text{Eq. 13-22}$$

A unique mean value of the transverse shear strain is assumed to exist for both the x and y components of the element coordinate system, but for ease of discussion, only the evaluation of an uncoupled x component of the shear moduli will be illustrated here. From Eq. 13-22 the mean value of transverse shear modulus may be written in the following form

$$\frac{1}{G_x} = \frac{T}{V_x^2} \sum_{i=1}^N \int_{z_{i-1}}^{z_i} \frac{(\tau_{zx}(z))^2}{(G_x)_i} dz \quad \text{Eq. 13-23}$$

where G is an “average” transverse shear coefficient used by the element code and $(G_x)_i$ is the local shear coefficient for layer i . To evaluate Eq. 13-14, it is necessary to obtain an expression for $(\tau_{zx}(z))$. This can be accomplished by assuming that the x- and y-components of stress are decoupled from one another. This assumption allows the desired equation to be deduced through an examination of a beam unit cross-sectional width.



The equilibrium conditions in the horizontal direction and for total moment are

$$\frac{\partial \tau_{xz}}{\partial z} + \frac{\partial \sigma_x}{\partial x} = 0 \quad \text{Eq. 13-24}$$

$$V_x + \frac{\partial M_x}{\partial x} = 0 \quad \text{Eq. 13-25}$$

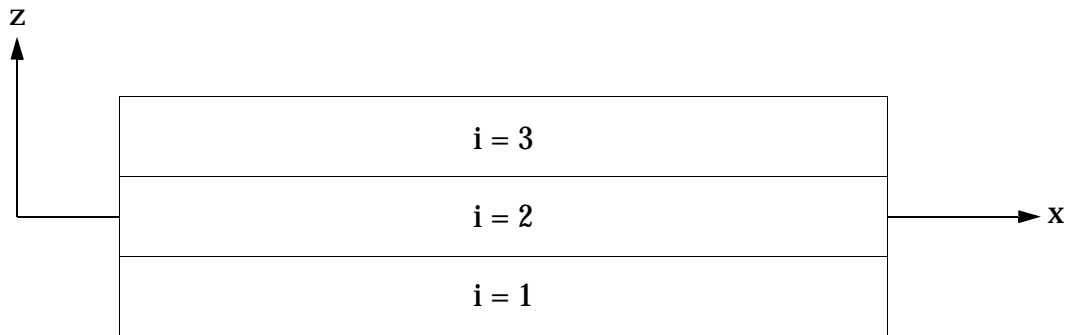
Now, if the location of the neutral surface is denoted by \bar{z}_x and ρ is the radius of curvature of the beam, the axial stress E_x may be expressed in the form

$$\sigma_x + \frac{E_x(\bar{z}_x - z)}{(\bar{EI})_x} = 0 \quad \text{Eq. 13-26}$$

Relation 20 may be differentiated with respect to x combined with **Eq. 13-24** and **Eq. 13-25**. In a region of constant E_x the result may be integrated to yield the following expression

$$\tau_{xz} = C_i + \frac{V_x}{(\bar{EI})_x} \left(\bar{z}_x z - \frac{z^2}{2} \right) E_{xi} \quad z_{i-1} < z < z_i \quad \text{Eq. 13-27}$$

Eq. 13-31 is particularly convenient to use in the analysis of n -ply laminates because sufficient conditions exist to determine the constants C_i ($i = 1, 2, \dots, n$) and the “directional bending center” \bar{z}_x . For example, consider the following laminated configuration:



At the bottom surface ($i = 1$, $z = z_0$, and $\tau_{xz} = 0$)

$$C_1 = \frac{-V_x}{(\bar{EI})_x} \left(\bar{z}_x z_0 - \frac{z_0^2}{2} \right) E_{x1} \quad \text{Eq. 13-28}$$

and for the first ply at the interface between plies $i = 1$ and $i = 2$ ($z = z_1$)

$$(\tau_{xz})_1 = + \frac{V_x}{(\bar{EI})_x} \left[\bar{z}_x (z_1 - z_0) \frac{1}{2} [z_1^2 - z_0^2] \right] E_{x1} \quad \text{Eq. 13-29}$$

At this interface between plies $i = 1$ and $i = 2$,

$$(\tau_{xz})_2 = C_2 + \frac{V_x}{(\overline{EI})_x} \left(\bar{z}_x z_1 - \frac{z_1^2}{2} \right) E_{x2} \quad \text{Eq. 13-30}$$

and as $(\tau_{xz})_2 = (\tau_{xz})_1$ at $z = z_1$,

$$C_2 = (\tau_{xz})_1 - \frac{V_x E_{x2}}{(\overline{EI})_x} \left[\bar{z}_x z_1 - \frac{1}{2} z_1^2 \right] \quad \text{Eq. 13-31}$$

Then, in the ply $z_1 < z < z_2$ the shear is

$$\tau_{xz}(z) = (\tau_{xz})_1 \frac{V_x E_{x2}}{(\overline{EI})_x} \left[\bar{z}_x (z - z_1) - \frac{1}{2} (z^2 - z_1^2) \right] \quad \text{Eq. 13-32}$$

In general, for any ply, $z_{i-1} < z < z_i$, the shear is

$$\tau_{xz}(z) = (\tau_{xz})_{i-1} \frac{V_x E_{xi}}{(\overline{EI})_x} \left[\bar{z}_x (z - z_{i-1}) - \frac{1}{2} (z^2 - (z_{i-1})^2) \right] \quad \text{Eq. 13-33}$$

At any ply interface, z_i , the shear is therefore

$$(\tau_{xz})_i = \frac{V_x}{(\overline{EI})_{xj=1}} \sum_{j=1}^i E_{xj} T_j \left[\bar{z}_x - \frac{1}{2} (z_j + z_{j-1}) \right] \quad \text{Eq. 13-34}$$

where $T_j = z_j - (z_{j-1})$.

Note that the shear at the top face, $(\tau_{xz})_n$, is zero and therefore

$$(\tau_{xz})_n = \frac{V_x}{(\overline{EI})} \left[\bar{z}_x \sum_{j=1}^n E_{xj} T_j - \sum_{j=1}^n E_{xj} T_j (z_j + z_{j-1}) / 2 \right] = 0 \quad \text{Eq. 13-35}$$

Eq. 13-35 proves that if \bar{z}_x is the bending center, the shear at the top surface must be zero.

Eq. 13-33 could be substituted into **Eq. 13-24** and integrated. A better form of **Eq. 13-35**, for this purpose is

$$(\tau_{xz}(z))_i = \frac{V_x E_{xi}}{(\overline{EI})_x} \left[f_{xi} + \bar{z}_x (z - z_{i-1}) - \frac{1}{2} (z^2 - (z_{i-1})^2) \right] \quad \text{Eq. 13-36}$$

where

$$f_{xi} = \frac{1}{E_{xi}} \sum_{j=1}^{i-1} E_{xj} T_j \left[\dot{z}_x - \frac{1}{2}(z_j + z_{j-1}) \right] \quad \text{Eq. 13-37}$$

Substituting [Eq. 13-36](#) into [Eq. 13-27](#) and after a considerable effort of integrating the results, we obtain

$$\frac{1}{G_x} = \frac{T}{(\overline{EI})_x} \sum_{i=1}^N \frac{1}{G_{xi}} R_{xi} \quad \text{Eq. 13-38}$$

where

$$\begin{aligned} R_{xi} = & (E_{xi})^2 T_i \left[\left\{ f_{xi} + (\dot{z}_x - z_{i-1}) T_i - \frac{1}{3} T_i^2 \right\} f_{xi} \right. \\ & \left. + \left\{ \frac{1}{3} (\dot{z}_x - 2 z_{i-1}) - \frac{1}{4} T_i \right\} \dot{z}_x T_i^2 + \left\{ \frac{1}{3} z_{i-1}^2 + \frac{1}{4} z_{i-1} T_i + \frac{1}{20} T_i^2 \right\} T_i^2 \right] \end{aligned} \quad \text{Eq. 13-39}$$

This expression for the inverse shear modulus for the x-direction may be generalized to provide for the calculation of each term in the two-by-two matrix of shear moduli.

$$[\bar{G}_{kl}] = \left[\frac{T}{(\overline{EI})_{kk}} \sum_{i=1}^n [G_{kl}^i]^{-1} R_{ki} \right]^{-1} \quad \text{Eq. 13-40}$$

where:

$$k = 1,2$$

$$l = 1,2$$

Note that if no shear is given, $[G^i]^{-1} = 0$, and also that in [Eq. 13-33](#)

where $[G_2]^*$ is calculated in the same manner as $[G_2]$ except that Poisson's Ratio is set to zero.

The moduli for individual plies are provided through user input because, in general, $G_{12} \neq G_{21}$ will be used for the coupling terms. Finally,

$$[G_3] = \begin{bmatrix} \bar{G}_{11} & (\bar{G}_{12})_{\text{avg}} \\ (\bar{G}_{12})_{\text{avg}} & \bar{G}_{22} \end{bmatrix} \quad \text{Eq. 13-41}$$

As an example, consider a single layer element. For this case let $z_{i-1} = -T/2$, $\bar{z} = 0$, $f_o = 0$, and $EI = ET^3/12$. Evaluating [Eq. 13-40](#) we obtain

$$R_i = E^2 T^5 \left[\frac{1}{12} - \frac{1}{8} + \frac{1}{20} \right] = \frac{E^2 T^5}{120}$$

and

$$\frac{1}{G} = \left(\frac{(12)^2 T}{E^2 T^6} \right) \cdot \left(\frac{E^2 T^5}{120 G_1} \right) = \frac{6}{5 G_1}$$

which provides the correct factor for a nonuniform shear distribution in a plate.

13.2.4 Data Recovery

As the material properties of the laminated composite plate are completely contained in the matrices of elastic moduli, standard data recovery methods (see “[Nonuniform Material and Composite Properties for Shells](#)” on page 280) may be used to calculate stresses in individual laminae and the forces sustained by the laminate.

An additional aid to the analyst is provided through optional output of a failure index for individual laminae. Failure indices assume a value of one on the periphery of a failure surface in stress space. If the failure index is less than one, the lamina stress is interior to the periphery of the failure surface and the lamina is assumed safe.

Conversely, if the failure is greater than one, the lamina stress is exterior to the periphery of the failure surface and the lamina is assumed to have failed. These failure indices represent a phenomenological failure criterion in that only an occurrence of a failure is indicated and not the mode of failure.

In the present context, concern is with the analytical definition of a failure surface in stress or strain space for use with laminae under biaxial loading. Four commonly used definitions of the failure surface are provided: Hill's Theory, Hoffman's Theory, Tsai-Wu Theory, and maximum strain theory.

The discussion of data recovery activities concludes with a description of the approximate procedure provided to obtain interlaminar shear stresses.

13.2.5 Failure Indices

In the analysis of isotropic materials, strength is independent of the orientation of the body under load and one may compare the largest computed principal stress with an allowable stress to establish the integrity of the structure. Laminated composites, on the other hand, are orthotropic materials and may exhibit unequal properties in tension and compression. Thus, the strength of these orthotropic laminae is a function of body orientation relative to the imposed stresses.

As the evaluation of the matrices of material moduli for laminated composites provides sufficient information to determine the actual stress field sustained by the material, the determination of structural integrity will depend on the definition of an allowable stress field. The basic ingredient of this definition is the establishment of a set of allowable stresses or strengths in the principal material directions.

X_t = Allowable tensile stress in the principal x (or 1) direction of the material.

X_c = Allowable compressive stress in the principal x (or 1) direction of the material.

Y_t = Allowable tensile stress in the principal y (or 2) direction of the material.

Y_c = Allowable compressive stress in the principal y (or 2) direction of the material.

S = Allowable shear stress in the principal material system.

Expressions for the three available failure theories are shown in [Table 13-1](#).

Table 13-1 Failure Theories Provided in MSC.Nastran

Theory	Failure Index	Remarks
Hill	$\frac{\sigma_1^2}{x^2} - \frac{\sigma_1\sigma_2}{x^2} + \frac{\sigma_2^2}{y^2} + \frac{\tau_{12}^2}{s^2} = \text{FI}$	Orthotropic materials with equal strengths in tension and compression, i.e., $x_t = x_c$ and $y_t = y_c$.
Hoffman	$\left(\frac{1}{X_t} - \frac{1}{X_c}\right)\sigma_1 + \left(\frac{1}{y_t} - \frac{1}{y_c}\right)\sigma_2 + \frac{\sigma_1^2}{x_t x_c} + \frac{\sigma_2^2}{y_t y_c} + \frac{\sigma_{12}^2}{s^2} - \frac{\sigma_1\sigma_2}{x_t x_c} = \text{FI}$	Orthotropic materials under a general state of plane stress with unequal tensile and compressive strengths.

Table 13-1 Failure Theories Provided in MSC.Nastran

Theory	Failure Index	Remarks
Tsai-Wu	$\left(\frac{1}{X_t} - \frac{1}{X_c}\right)\sigma_1 + \left(\frac{1}{y_t} - \frac{1}{y_c}\right)\sigma_2 + \frac{\sigma_1^2}{x_t x_c} + \frac{\sigma_2^2}{y_t y_c} + \frac{\sigma_{12}^2}{s^2} + 2F_{12}\sigma_1\sigma_2 = \text{FI}$	Orthotropic materials under a general state of plane stress with unequal tensile and compressive strengths.
MAX STRAIN	$\left[\left(\frac{\epsilon_1}{X_t}\right), \left(\frac{\epsilon_2}{Y_t}\right), \left(\frac{ \gamma_{12} }{S}\right)\right]$	

The following remarks apply to **Table 13-1**:

1. The failure index of Hill represents an analytical expression for an ellipse in the $\sigma_1\sigma_2$ plane. This original expression for failure was subsequently modified to that shown on the left-hand side of the expression in **Table 13-1** by Tsai and hence this failure theory is sometimes known as the Tsai-Hill Theory.
2. The Tsai-Wu tensor polynomial theory for orthotropic materials is specialized to the expression given in **Table 13-1** for orthotropic laminae under a condition of plane stress. The coefficient F_{12} is experimentally determined from test specimens under biaxial loading. This inconvenience along with the constraint that F_{12} satisfy a stability criterion of the form

$$\left(\frac{1}{x_t x_c}\right)\left(\frac{1}{y_t y_c}\right) - F_{12}^2 > 0 \quad \text{Eq. 13-42}$$

creates some complication in the use of this theory. Narayanaswami and Adelman (see Reference 1.) have suggested that F_{12} be set to zero and the use of Hoffman's Theory or the Tsai-Wu theory with $F_{12} = 0$ are preferred alternatives. The analyst may provide a value of f_{12} to be used in the Tsai-Wu failure index if he wishes.

Hill's Theory (Failure Theory ID = "HILL")

$$\text{Failure Index} = \frac{\sigma_1^2}{X^2} - \frac{\sigma_1\sigma_2}{X^2} + \frac{\sigma_2^2}{Y^2} + \frac{\tau_{12}^2}{S^2} \quad \text{Eq. 13-43}$$

where:

X is allowable stress in 1-direction

Y is allowable stress in 2-direction

S is allowable stress in shear

and $X = X_t$ if σ_1 is positive or $X = X_c$ if σ_1 is negative and similarly for Y and σ_2 . For the interaction term $\sigma_1\sigma_2/X^2$, $X = X_t$ if $\sigma_1\sigma_2$ is positive or $X = X_c$ if $\sigma_1\sigma_2$ is negative.

Note that Hill's theory is used for materials that have the same strength in tension and compression.

A plot of the above equation obtained by setting the failure index to 1 on the σ_1 - σ_2 plane yields an ellipse and is the anisotropic yield criterion of Hill (modified later by Tsai, and hence also sometimes known as the Tsai-Hill theory). Therefore, if the failure index so calculated is less than 1, the ply stresses are inside the yield ellipse and the ply is said to be "safe"; conversely, if the failure index is greater than 1, the ply stresses are outside the yield ellipse and the ply has failed.

Hoffman's Theory (Failure Theory ID = "HOFF")

The Hoffman's theory for an orthotropic lamina in a general state of plane stress is given by

$$\left(\frac{1}{X_t} - \frac{1}{X_c}\right)\sigma_1 + \left(\frac{1}{Y_t} - \frac{1}{Y_c}\right)\sigma_2 + \frac{\sigma_1^2}{X_t X_c} + \frac{\sigma_2^2}{Y_t Y_c} + \frac{\sigma_{12}^2}{S^2} - \frac{\sigma_1\sigma_2}{X_t X_c} = 1 \quad \text{Eq. 13-44}$$

The failure index is obtained by evaluating the left-hand side of the above equation.

Note that this theory takes into account the difference in tensile and compressive allowable stresses by using linear terms in the equation.

Tensor Polynomial Theory of Tsai-Wu (Failure Theory ID = "TSAI")

The theory of strength for anisotropic materials proposed by Tsai and Wu specialized to the case of an orthotropic lamina in a general state of plane stress is

$$F_1\sigma_1 + F_2\sigma_2 + F_{11}\sigma_1^2 + F_{22}\sigma_2^2 + 2F_{12}\sigma_1\sigma_2 + F_{66}\sigma_{12}^2 = 1 \quad \text{Eq. 13-45}$$

where:

$$F_1 = \frac{1}{X_t} - \frac{1}{X_c}$$

$$F_2 = \frac{1}{Y_t} - \frac{1}{Y_c}$$

$$F_{11} = +\frac{1}{X_t X_c}$$

$$F_{22} = +\frac{1}{Y_t Y_c}$$

$$F_{66} = +\frac{1}{S^2}$$

and F_{12} is to be evaluated experimentally.

The magnitude of F_{12} is, however, constrained by the following inequality called a “stability criterion”:

$$F_{11}F_{22} - F_{12}^2 > 0 \quad \text{Eq. 13-46}$$

The necessity of satisfying the stability criterion, together with the requirement that F_{12} be determined experimentally from a combined stress state, poses difficulties in the use of this theory. Narayanaswami and Adelman (see Reference 1.) have suggested that F_{12} be set to zero and that use of Hoffman's theory or Tensor Polynomial theory with $F_{12} = 0$ is a preferred alternative to the experimental determination of F_{12} . If the user has a value for use with F_{12} in the theory, the same may be input in the MAT8 Bulk Data entry; otherwise, F_{12} in the above theory will be set to 0.0. The left-hand side of the above equation will be evaluated as the failure index by this theory.

The failure index of bonding material will be calculated as the maximum interlaminar shear stress divided by the allowable bonding stress.

The failure indices for all the plies will be written into OEFIT (Output Element Failure Index Table) and output if stresses are requested. The failure index for the element is the largest value of the failure indices for all plies of the element.

Maximum Strain Theory (Failure Theory ID = “STRN”)

The midplane strains and curvatures are available in the element coordinate system. From these, the stresses and strains in each individual lamina along the fiber direction and transverse direction can be calculated quite easily. The STRAIN Case Control command is used to request the output of lamina strains.

The maximum strain criteria has no strain interaction terms. The strain allowables specified on the MAT8 entry for each lamina include

X_t, X_c	Allowable strains in tension and compression, respectively, in the longitudinal direction.
Y_t, Y_c	Allowable strains in tension and compression, respectively, in the transverse direction.
S	Allowable strain for inplane shear.

The Failure Index is calculated using

$$\frac{\varepsilon_1}{X_t} \text{ (or } \frac{\varepsilon_1}{X_c} \text{ if } \varepsilon_1 \text{ compressive)}$$

$$\frac{\varepsilon_2}{Y_t} \text{ (or } \frac{\varepsilon_2}{Y_c} \text{ if } \varepsilon_2 \text{ compressive)}$$

and

$$\left| \frac{\gamma_{12}}{S} \right|$$

i.e., the Failure Index = $\text{MAX} \left[\left(\frac{\varepsilon_1}{X} \right), \left(\frac{\varepsilon_2}{Y} \right), \left(\frac{|\gamma_{12}|}{S} \right) \right]$.

In addition, the user needs to be informed which mode of Failure Index is critical; i.e., longitudinal (1), transverse (2), or shear (12). Thus in the output for Failure Indices, the mnemonic 1, 2 or 12 is printed alongside the FP value to indicate the critical direction. There is no change in the way the Failure Index is calculated for interlaminar shear stresses.

It is quite possible that for the maximum strain theory, the user may wish to specify lamina stress allowables instead of strain allowables on the MAT8. This can be done by leaving the STRN field blank on the MAT8.

For this case, the failure indices are calculated using

$$\frac{\varepsilon_1 \cdot E_{11}}{X_t} \text{ (or } \frac{\varepsilon_1 \cdot E_{11}}{X_c} \text{ if } \varepsilon_1 \text{ compressive)}$$

$$\frac{\varepsilon_2 \cdot E_{22}}{Y_c} \text{ (or } \frac{\varepsilon_2 \cdot E_{22}}{Y_c} \text{ if } \varepsilon_2 \text{ compressive)}$$

and

$$\frac{|\gamma_{12}| \cdot G_{12}}{S}$$

i.e., the Failure Index = $\text{MAX} \left[\left(\frac{\epsilon_1 \cdot E_{11}}{X} \right), \left(\frac{\epsilon_2 \cdot E_{22}}{Y} \right), \left(\frac{|\gamma_{12}| \cdot G_{12}}{S} \right) \right]$

13.2.6 Strength Ratios

For laminated composites, Strength Ratio (SR) is a direct failure indicator compared to Failure Index (FI) which indicates only if failure had occurred. Generally Strength Ratio is defined as:

$$\text{Strength Ratio (SR)} = \text{Allowable Stress} / \text{Calculated Stress}$$

For example a SR = 1.2 indicates that the applied loads can be increased by 20% before failure occurs. A FI = 0.8 indicates that failure has not occurred and does not indicate 20% safety margin. Therefore the SR is a much more practical design indicator for both analysis and strength-criteria based design.

This new result quantity is available for all solutions sequences that support composites data recovery.

A new parameter -- '*PARAM,SRCOMPS,YES/NO*' requests Strength Ratio (SR) output. SR Output requires specification of a failure theory and allowable stress/strain values. Note that SRs will be computed for plies with an output request (PCOMP-SOUTi field).

For Maximum Stress/Strain and Transverse Stress theories, the Strength Ratio (SR) is simply the inverse of the FI. For the quadratic failure theories such as Hill, Hoffman and Tsai-Wu the SR is calculated by solving the quadratic equation with the FI set to unity and replacing the applied stress with the SR · applied stress.

Below the SR expression for each Failure Criteria is defined.

HILL Failure Criteria

The HILL failure criteria is defined as:

$$\text{FI} = \left(\frac{\sigma_1^2}{X^2} \right) + \left(\frac{\sigma_2^2}{Y^2} \right) - \left(\frac{\sigma_1 \sigma_2}{X^2} \right) + \left(\frac{\sigma_{12}^2}{S^2} \right) \quad \text{Eq. 13-47}$$

where:

$$X = X_T \text{ if } \sigma_1 = \text{Tensile}$$

$$X = X_C \text{ if } \sigma_1 = \text{Compressive}$$

$$Y = Y_T \text{ if } \sigma_2 = \text{Tensiles}$$

$$Y = Y_C \text{ if } \sigma_2 = \text{Compressive}$$

$$\sigma_1 = \text{Ply longitudinal stress}$$

$$\sigma_2 = \text{Ply transverse stress}$$

$$\sigma_{12} = \text{Ply shear stress}$$

For the product term:

$$\frac{\sigma_1, \sigma_2}{X^2}$$

where:

$$X = X_T \text{ if } \sigma_1 \text{ and } \sigma_2 \text{ are of the same sign.}$$

$$X = X_C \text{ if } \sigma_1 \text{ and } \sigma_2 \text{ are of different signs.}$$

The FI expression can be redefined in terms of a strength ratio. The roots of the resulting quadratic expression are the reserve factors, the first one for the stresses as applied and the second for when the sign of all the components of stress are reversed.

The FI expression is re-arranged with the applied stress replaced by a ratio SR times the applied stress and equated to a FI of unity, the SR can now be determined.

$$1.0 = [(\text{SR}^2 \sigma_1^2) \cdot (1/X^2)] + [(\text{SR}^2 \sigma_2^2) \cdot (1/Y^2)] - \left[\frac{(\text{SR}^2 \sigma_1 \sigma_2)}{(X^2)} \right] + \left[\frac{(\text{SR}^2 \sigma_{12}^2)}{(Y^2)} \right]$$

Eq. 13-48

$$[(\sigma_1^2/X^2) + (\sigma_2^2/Y^2) - (\sigma_1 \sigma_2/X^2) + (\sigma_{12}^2/Y^2)] \text{SR}^2 - 1 = 0$$

$$a\text{SR}^2 + b\text{SR} + c = 0$$

from the general solution of a quadratic equation,

$$SR_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad SR_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

For the HILL criteria,

$$a = (\sigma_1^2/X^2) + (\sigma_2^2/Y^2) - (\sigma_1\sigma_2)/X^2 + (\sigma_{12}^2/Y^2)$$

$$b = 0.0$$

$$c = -1$$

$$SR_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$SR_1 = \frac{\sqrt{-4 \cdot a \cdot 1.0}}{2 \cdot a} = \frac{2 \cdot \sqrt{a}}{2 \cdot a}$$

$$SR_1 = \frac{1}{\sqrt{a}}$$

$$SR_2 = -\frac{1}{\sqrt{a}}$$

where $a = (\sigma_1^2/X^2) + (\sigma_2^2/Y^2) - (\sigma_1\sigma_2/X^2) + (\sigma_{12}^2/Y^2) = FI$

Therefore for the HILL quadratic failure criteria the SR can be directly derived from the FI:

$$SR = \frac{1}{\sqrt{FI}}$$

HOFFMAN Failure Criteria

Unlike the Hill criteria the Hoffman failure criteria takes into account the difference in tensile and compressive strengths in the longitudinal and transverse directions of the ply.

$$FI = [1/(X_T \cdot X_C)]\sigma_1^2 + [1/(Y_T \cdot Y_C)]\sigma_2^2 + [1/(X_T \cdot X_C)](\sigma_1\sigma_2) + (\sigma_{12}^2/S^2) \\ + [(1/X_T) - (1/X_C)]\sigma_1 + [(1/Y_T) - (1/Y_C)]\sigma_2$$

defining:

$$F_1 = (1/X_T) - (1/X_C)$$

$$F_2 = (1/Y_T) - (1/Y_C)$$

$$F_{11} = 1/(X_T \cdot X_C)$$

$$F_{22} = 1/(Y_T \cdot Y_C)$$

$$F_{66} = 1/(S^2)$$

$$FI = F_{11}\sigma_1^2 + F_{22}\sigma_2^2 - F_{11}(\sigma_1\sigma_2) + F_{66}\sigma_{12}^2 + F_1\sigma_1 + F_2\sigma_2$$

Now introduce applied stress as a ratio R times applied stress equated to a FI of unity.

$$1.0 = F_{11}SR^2\sigma_1^2 + F_{22}SR^2\sigma_2^2 - F_{11}SR^2(\sigma_1\sigma_2) + F_{66}SR^2\sigma_{12}^2 + F_1SR\sigma_1 + F_2SR\sigma_2$$

Arrange in quadratic form

$$[F_{11}\sigma_1^2 + F_{22}\sigma_2^2 - F_{11}(\sigma_1\sigma_2) + F_{66}\sigma_{12}^2]R^2 + [F_1\sigma_1 + F_2\sigma_2]R - 1 = 0$$

$$SR_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad SR_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

where:

$$a = [F_{11}\sigma_1^2 + F_{22}\sigma_2^2 - F_{11}(\sigma_1\sigma_2) + F_{66}\sigma_{12}^2]$$

$$b = [F_1\sigma_1 + F_2\sigma_2]$$

$$c = -1.0$$

This is now a quadratic that does not simplify in the same way as the Hill criteria so there is no direct relationship between FI and SR. The roots of the equation have to be determined to obtain the SR.

TSAI-Wu Failure Criteria

The Tsai-Wu failure criterion is very similar to the Hoffman criteria but has a different value for the coefficient associated with the interaction of the direct stresses. In the Hoffman criterion it is the same as the coefficient associated with the direct longitudinal stresses.

For the Tsai-Wu criterion:

$$FI = F_{11}\sigma_1^2 + F_{22}\sigma_2^2 + F_{12}(\sigma_1\sigma_2) + F_{66}\sigma_{12}^2 + F_1\sigma_1 + F_2\sigma_2$$

where:

$$F_1 = (1/X_T) - (1/X_C)$$

$$F_2 = (1/Y_T) - (1/Y_C)$$

$$F_{11} = 1/(X_T \cdot X_C)$$

$$F_{22} = 1/(Y_T \cdot Y_C)$$

$$F_{66} = 1/(S^2)$$

$$F_{12} = -0.5\sqrt{F_{11} \cdot F_{22}}$$

Now introduce applied stress as a ratio R times applied stress equated to a FI of unity.

$$1.0 = F_{11}SR^2\sigma_1^2 + F_{22}SR^2\sigma_2^2 - F_{12}SR^2(\sigma_1\sigma_2) + F_{66}SR^2\sigma_{12}^2 + F_1SR\sigma_1 + F_2SR\sigma_2$$

Arrange in quadratic form

$$[F_{11}\sigma_1^2 + F_{22}\sigma_2^2 + F_{12}(\sigma_1\sigma_2) + F_{66}\sigma_{12}^2]SR^2 + [F_1\sigma_1 + F_2\sigma_2]SR - 1 = 0$$

$$SR_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2c} \quad \text{and} \quad SR_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2c}$$

where:

$$a = [F_{11}\sigma_1^2 + F_{22}\sigma_2^2 - F_{12}(\sigma_1\sigma_2) + F_{66}\sigma_{12}^2]$$

$$b = [F_1\sigma_1 + F_2\sigma_2]$$

$$c = -1.0$$

This is a quadratic equation that does not simplify in the same way as the Hill criteria so there is no direct relationship between FI and SR. The roots of the equation have to be determined to obtain the SR.

Maximum Stress (Strain)

Failure Index FI = Calculated Stress / Allowable Stress

By substituting the Allowable stress by RF*Allowable Stress and setting FI to unity

$$SR = 1 / (FI)$$

Transverse Shear Stress

For this case the Strength Ratio is simply the inverse of the failure index.

13.2.7 "LAM" Options

The PCOMP Bulk Data entry is used to provide input data for the properties of an n-ply composite material laminate. As such, it is widely used in aerospace applications where a composite structure is built up from a number of discrete sheets of material, such as a carbon-epoxy fiber.

1. The SMEAR options to the LAM field smears the ply effects while ignoring stacking sequence.
2. The SMCORE option to the LAM field enables the modeling of a skin/core combination where the ply effects are smeared in the skin, the core inertial terms are treated exactly and the core stiffness terms are ignored.

MSC.Nastran develops mass and stiffness data from PCOMP input in a two-step process. First, the PCOMP input data are considered together with the material data referenced by MIDi entries to produce PSHELL/MAT2 combinations that will lead to the required stiffness results and then this spawned data are used in the actual stiffness and mass calculations. The spawned PSHELL has four MIDis, identifying the MAT2s to be used for membrane, bending, transverse shear and membrane-bending coupling. The SMEAR and SMCORE options use these MIDis in the following manner:

SMEAR

The spawned PSHELL has MID1=MID2 with MID3, MID4, 12I/T**3, and the TS/T fields set as blanks. This results in a bending term given as

$$[D] = \frac{T^2[A]}{12} \quad \text{Eq. 13-49}$$

SMCORE

Computation of the membrane and bending stiffness matrices is performed using the following derivation. Note that membrane-bending coupling is ignored.

Definitions:

$$t_{face} = T_1 + T_2 + \dots + T_{N-1} \text{ (total thickness of both face sheets)} \quad \text{Eq. 13-50}$$

$$t_{core} = T_N \text{ (total core thickness)}$$

Membrane Stiffness Matrix:

[A] is computed using method utilized by LAM=BLANK assuming that core stiffness (layer N) is zero.

Face Sheet Properties:

$$\mu_{xy} = \frac{-[A]_{12}^{-1}}{[A]_{11}^{-1}} \quad \text{Eq. 13-51}$$

$$\mu_{yx} = \frac{-[A]_{12}^{-1}}{[A]_{22}^{-1}} \quad \text{Eq. 13-52}$$

$$E_x = \frac{A_{11}(1.0 - \mu_{xy}\mu_{yx})}{t_{face}} \quad \text{Eq. 13-53}$$

$$E_y = \frac{A_{22}(1.0 - \mu_{xy}\mu_{yx})}{t_{face}} \quad \text{Eq. 13-54}$$

$$G_{xy} = \frac{A_{33}}{t_{face}} \quad \text{Eq. 13-55}$$

Moment of inertia:

$$I_{xx} = I_{yy} = \frac{t_{face} \left(t_{core} + \frac{t_{face}}{2} \right)^2}{4} + \frac{t_{face}^3}{48} \quad \text{Eq. 13-56}$$

Note, this collapses to $t_{face}^3/12$ if t_{core} is zero.

Bending Stiffness Matrix:

$$D_{11} = \frac{E_x I_{xx}}{(1.0 - \mu_{xy}\mu_{yx})} \quad \text{Eq. 13-57}$$

$$D_{22} = \frac{E_y I_{yy}}{(1. - \mu_{xy}\mu_{yx})} \quad \text{Eq. 13-58}$$

$$D_{12} = D_{21} = \frac{D_{11}\mu_{xy}E_y}{E_x} \quad \text{Eq. 13-59}$$

$$D_{33} = G_{xy}I_{xx} \quad \text{Eq. 13-60}$$

$$D_{13} = D_{31} = D_{23} = D_{32} = 0 \quad \text{Eq. 13-61}$$

Input

The format of the PCOMP entry is unchanged, but the description of the LAM field is expanded and an additional Remark is added.

Field	Contents
LAM	Laminate Options (Character or Blank, default=Blank). See Remarks 13 . and 14 .
“Blank”	All plies must be specified and all stiffness terms are developed
“SYM”	Only plies on one side of the element centerline are specified. The plies are numbered starting with 1 for the bottom layer. If an odd number of plies are desired, the center ply thickness (T1 or TN) should be half the actual thickness.
“MEM”	All plies must be specified, but only membrane terms (MID1 on the derived PSHELL entry) are computed.
“BEND”	All plies must be specified, but only bending terms (MID2 on the derived PSHELL entry) are computed.

Field	Contents
"SMEAR"	All plies must be specified, stacking sequence is ignored (MID1=MID2 on the derived PSHELL entry and MID3, MID4 and TS/T and 12I/T**3 terms are set to zero).
"SMCORE"	All plies must be specified, with the last ply specifying core properties and the previous plies specifying face sheet properties. The stiffness matrix is computed by placing half the face sheet thicknesses above the core and the other half below with the result that the laminate is symmetric about the mid-plane of the core. Stacking sequence is ignored in calculating the face sheet stiffness.

Remarks

- The SYM option for the LAM option computes the complete stiffness properties while specifying half the plies. The MEM, BEND, SMEAR and SMCORE options provide special purpose stiffness calculations. SMEAR ignores stacking sequence and is intended for cases where this sequence is not yet known, stiffness properties are smeared. SMCORE allows simplified modeling of a sandwich panel with equal face sheets and a central core.
- Element output for the SMEAR and SMCORE options is produced using the PARAM NOCOMPS -1 methodology that suppresses ply stress/strain results and prints results for the equivalent homogeneous element.

Output

Standard element response data are recovered for elements that employ the new LAM options with the key provision that the SMEAR and SMCORE options results in element results that correspond to the equivalent homogeneous element and not the ply-by-ply results. This is equivalent to using PARAM NOCOMPS -1 and only the homogeneous results are produced regardless of the user input specification of the NOCOMPS parameter.

Guidelines for Design Optimization

Special procedures must be employed when using the PCOMP with the SMEAR and SMCORE options in an automated design task in SOL 200. The FSD option in SOL 200 does not support the SMEAR or SMCORE options since the resizing of a ply layer in FSD is driven by a response in that layer. The SMEAR/SMCORE options do not produce layer responses, therefore, FSD resizing is precluded. For the standard mathematical programming of SOL 200, composite stress, strain and force results are

not available for the PCOMP with SMEAR or SMCORE. Instead, the homogeneous element results must be invoked. This is done by using the standard STRESS, STRAIN, FORCE response types (as opposed to the CSTRESS, CSTRAIN, CFAILURE response types) and the PTYPE can be set to either PCOMP or PSHELL. The PID for the PSHELL is identical to that of the PCOMP. The rules for the design of properties are not changed by these new options in that the user can invoke a lamina thickness or orientation on the PCOMP entry from a DVPREL1 or DVPREL2. Invoking the thickness of the equivalent PSHELL that is derived from a PCOMP is not allowed and produces a fatal error.

Example PSTIFF1, PSOFT1, PMEM1, PSMEAR and PSMCORE

13.2.8 Temperature-Dependent Properties

Introduction

MSC.Nastran 2004 updates the smeared laminate properties with the temperature-dependent material properties of the individual plies, and calculates the thermal strain as an integral rather than a secant. In addition, temperature dependence has been added for shell orthotropic materials, and thermal strains may be input directly. These enhancements apply to nonlinear static analysis.

Benefits

Prior to MSC.Nastran 2004, the smeared laminate properties for a composite material were calculated only once, at the beginning of the analysis, using the reference temperature with no gradient through the thickness. The thermal strain was calculated using a secant approach with the coefficient of thermal expansion at the initial temperature and the load temperature. For a nonlinear analysis, where the temperature may change appreciably, and especially where thermal buckling may occur, the new smeared laminate properties calculations provide a more realistic simulation.

Theory

Temperature Distribution

The temperature through the shell thickness is assumed to be linear:

$$T(\xi, \eta, \zeta) = T(\xi, \eta) + \zeta \cdot T', \quad -\frac{t}{2} \leq \zeta \leq \frac{t}{2}$$

where $T(\xi, \eta)$ is the membrane temperature defined by the weighted average of the grid temperatures from the TEMP and/or TEMPD Bulk Data entries, or the element temperature from the TEMPP1 Bulk Data entry; and T' is the temperature gradient from the TEMPP1 Bulk Data entry.

Currently the smeared laminate properties are calculated using the reference temperature from the PCOMP Bulk Data entry, which has no gradient defined through the thickness. This implies that all the plies are at the same temperature, which is not a valid assumption in certain cases.

The temperature at the midplane of the i -th ply will be calculated as:

$$T_i = \bar{T} + \left(z_0 + \sum_{j=1}^{i-1} t_j + \frac{1}{2} t_i \right) \cdot T'$$

where \bar{T} is the reference plane temperature; z_0 is the bottom surface offset from the reference plane; and t_i are the ply thicknesses defined on the PCOMP Bulk Data entry. This temperature will be used to calculate the temperature-dependent material properties of each ply.

Coefficient of Thermal Expansion

Secant Approach

The coefficient of thermal expansion (in one dimension) is defined as the normalized change in length for a given change in temperature:

$$\alpha = \frac{1}{L} \frac{\Delta u}{\Delta T}$$

such that the thermal strain is given by:

$$\varepsilon = \alpha \Delta T$$

which is the secant approach. This can be computed in several ways, depending on the case control and material dependence.

For linear analysis using neither TEMP(MAT) nor TEMP(INIT):

$$\varepsilon = \alpha(T_{load} - T_{ref})$$

For linear analysis using TEMP(INIT) with no MATTi:

$$\varepsilon = \alpha(T_{load} - T_{init})$$

For linear analysis using TEMP(MAT) with MATTi:

$$\varepsilon = \alpha_{mat}(T_{load} - T_{ref})$$

For linear analysis using TEMP(INIT) with MATTi:

$$\varepsilon = \alpha_{init}(T_{load} - T_{init})$$

For nonlinear analysis using TEMP(INIT) with MATTi (TEMP(INIT) is required and TEMP(MAT) is not allowed), the expression is more complex, in an attempt to better approximate the temperature-dependent properties:

$$\varepsilon = \alpha_{load}(T_{load} - T_{ref}) - \alpha_{init}(T_{init} - T_{ref})$$

Integral Approach

This secant approximation is adequate for small changes in temperature; however, for temperature-dependent properties, a derivative approximation is more accurate:

$$\alpha(T) = \frac{1}{L} \frac{\partial u}{\partial T}$$

such that the thermal strain is given by:

$$\varepsilon = \int_{T_{init}}^{T_{load}} \alpha(T) dT$$

which is the integral approach.

Temperature dependence for each material property, such as the coefficient of thermal expansion, may be defined as a linear function or a power series, and may include offsets and scale factors. These are input on the TABLEMi Bulk Data entries, and may be integrated analytically for each interval of the table.

Input

For the temperature-dependent composites, two new parameters have been added. The COMPMATT parameter controls whether the smeared laminate properties are updated with the temperature-dependent material properties of the individual plies. It has a default value of NO and must be changed to YES to invoke the enhancement. The EPSILON_T parameter controls whether the thermal strain is computed using an integral or a secant method. It has a default value of SECANT and must be changed to INTEGRAL to invoke the enhancement.

A new MATT8 Bulk Data entry has been added for temperature-dependent shell orthotropic material properties, and thermal strains may be input directly by using a negative TABLEMi ID for the coefficients of thermal expansion on the MATTi Bulk Data entry. All of the other bulk data entries and case control remain the same.

Output

The output for the temperature-dependent composites has not changed; rather, the way in which it is calculated has changed.

For the strains, the thermal strains will be calculated using the smeared coefficients of thermal expansion at the current load temperature, in addition to using the appropriate secant or integral method. For the stresses and forces, the smeared material properties will be calculated at the proper temperature. For the ply strains and stresses, the thermal strain will be calculated using the coefficients of thermal expansion at the current load temperature and appropriate method for the specific ply, and the ply properties will be calculated at the proper temperature, rather than using the reference temperature on the PCOMP Bulk Data entry.

In addition, if the parameter COMPMATT is set, the composite elements will be included in the nonlinear stress/strain output.

Guidelines and Limitations

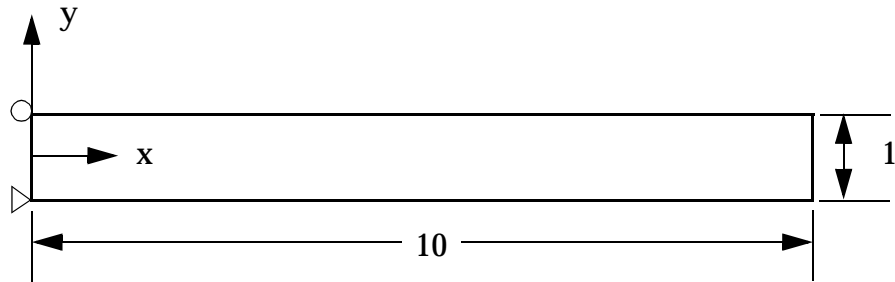
Composite capabilities are currently available with the QUAD4, QUAD8, TRIA3, and TRIA6 h-elements, but nonlinear capabilities are currently available with only the linear elements. Therefore only the QUAD4 and TRIA3 element types will be supported. (Note that although smeared laminate properties may be applied to other two-dimensional elements as well, ply-by-ply post-processing is not performed on those elements.)

The capabilities will be implemented for SOL 106, Nonlinear Statics. They will not be implemented for SOL 101, Linear Statics, nor SOL 129, Nonlinear Transient Response.

Examples

(TDCRN_*.DAT)

In order to illustrate the new enhancements, a very simple problem will be solved for each permutation of the parameters. The problem consists of a simply-supported cantilever beam, ten units long by one unit wide, undergoing a uniform free expansion from $T_{init} = 100$ to $T_{load} = 200$ with $T_{ref} = 50$, as shown in the following figure:



Two independent versions of the model will comprise each test file: one will reference a PSHELL, and the other will reference a PCOMP with a single ply of the same thickness and orientation as the shell. The material is isotropic, although a MAT8/MATT8 will be used to demonstrate the new MATT8.

The coefficient of thermal expansion is temperature-dependent, as shown in the following table:

	Temperature (T)	CTE (α)
T_{ref}	50	0.0005
T_{init}	100	0.001
T_{load}	200	0.005

Most of the models will be run in SOL 106, Nonlinear Statics, though one will be run in SOL 101, Linear Statics, for comparison purposes. (Note again that the enhancements are not implemented in SOL 101.)

The six example problems are listed in the following table, along with the thermal strains computed using the equations given above. For the final problem, the thermal strains, marked with an asterisk, are input directly on the TABLEM4 entry as the quadratic integral of the linear property variation of the coefficient of thermal expansion.

	Solution Sequence	Parameters		Thermal Strain (ϵ_t)	
		COMPMATT	EPSILON T	PSHELL	PCOMP
TDCRN_L	101	n/a	n/a	0.10	0.05
TDCRN_NS	106	no	secant	0.70	0.05
TDCRN_YS	106	yes	secant	0.70	0.70
TDCRN_NI	106	no	integral	0.30	0.05

	Solution Sequence	Parameters		Thermal Strain (ϵ_t)	
		COMPMATT	EPSILONT	PSHELL	PCOMP
TDCRN_YI	106	yes	integral	0.30	0.30
TDCRN_YIE	106	yes	integral	0.30*	0.30*

Results from the six example problems are listed in the following table, along with the axial displacement at the final load step.

	Solution Sequence	Parameters		Axial Displacement (u_x)	
		COMPMATT	EPSILONT	PSHELL	PCOMP
TDCRN_L	101	n/a	n/a	1.0	0.5
TDCRN_NS	106	no	secant	7.0	0.5
TDCRN_YS	106	yes	secant	7.0	7.0
TDCRN_NI	106	no	integral	3.0	0.5
TDCRN_YI	106	yes	integral	3.0	3.0
TDCRN_YIE	106	yes	integral	3.0*	3.0*

The axial displacements consist of the thermal strains multiplied by the length, as expected; the strains are those given above; and the stresses are numerically zero.

13.3 Nonlinear Analysis

General Description of Nonlinear Capabilities

Nonlinear effects in structures occur mainly due to nonlinear materials and large deformations (rotations and/or strains). Contact problems exhibit nonlinear effects due to changes in boundary conditions. All of these attributes may be represented by MSC.Nastran nonlinear elements.

The nonlinear element library consists of:

- CBUSH and CBUSH1D for generalized nonlinear springs and dampers.
- CROD, CONROD, and CTUBE for unidirectional truss members.
- CBEAM for axially and laterally deforming line members.
- CQUAD4 and CTRIA3 for membrane, plate and shell modeling and hyperelastic plane strain.
- CQUAD8, CQUAD, and CTRIA6 for hyperelastic plane strain.
- CQUADX and CTRIAX for hyperelastic axisymmetric modeling.
- CHEXA, CPENTA, and CTETRA for solid modeling.
- CGAP and slideline contact (BCONP, BLSEG, BFRIC, BWIDTH) for contact and friction modeling.

Nonlinear elements may be combined with linear elements for computational efficiency if the nonlinear effects can be localized. Primary operations for nonlinear elements are updating element coordinates and applying loads for large displacements, and the internal relaxation iteration for material nonlinearity. Refer to [Table 13-2](#) for a summary of the nonlinear element properties.

The geometric nonlinearity becomes discernible when the structure is subjected to large displacement and rotation. Geometric nonlinear effects are prominent in two different aspects: geometric stiffening due to initial displacements and stresses, and follower forces due to a change in loads as a function of displacements. These effects are included, but the large deformation effect resulting in large strains is implemented only for hyperelastic materials. Large strain effects are not available for elastic-plastic, creep, or any other material types.

Material nonlinearity is an inherent property of any engineering material. Material nonlinear effects may be classified into many categories and those included in MSC.Nastran are plasticity, nonlinear elasticity, hyperelasticity, creep, and

viscoelasticity. Creep and viscoelasticity, implemented as a generalized nonlinear viscoelastic capability, may be coupled with plasticity. Many options are available for yield criteria and hardening behavior in plasticity.

The primary solution operations are gradual load or time increments, iterations with convergence tests for acceptable equilibrium error, and stiffness matrix updates. The iterative process is based on the modified-Newton's method combined with optional expeditious methods such as the quasi-Newton (BFGS) update and the line search. The stiffness matrix updates are performed occasionally to improve the computational efficiency, and may be overridden at the user's discretion. A number of options for arc-length methods are also available for snap-through or post-buckling analysis of static problems. For transient response analysis, a number of options are available for implicit direct time integration, combined with adaptive and expedient iteration strategies similar to those implemented for static analysis.

Solution Sequences 106, 129, 153, and 159 consolidate all the nonlinear features described above. SOLs 106 and 153 are applicable to static, quasi-static, and nonlinear buckling analyses. SOLs 129 and 159 are primarily applicable to dynamic transient response analysis with some limited static analysis capability. SOLs 153 and 159 may also be used to perform a nonlinear heat transfer analysis. See the [*MSC.Nastran Thermal Analysis User's Guide*](#). These solution sequences can accommodate superelements and provide easy restarts from the database.

There are three other nonlinear solution sequences—SOLs 66, 99, and 64. SOL 66 is the unstructured solution sequence corresponding to 106. SOL 99 is the unstructured solution sequence corresponding to SOL 129. However, Version 68 nonlinear enhancements are in the structured solution sequences only, SOLs 106 and SOL 129. For example, the slideline contact capability is only added to SOL 106 and SOL 129. Therefore, it is strongly recommended that the users select the structured solution sequences, SOLs 106 and 129. SOL 64, the forerunner of these solution sequences, was developed only to analyze geometric nonlinear problems. No new capabilities have been added to SOL 64 since Version 63. SOL 64 will be eventually removed because SOL 106 can be substituted for any capability in SOL 64.

Table 13-2 Summary of Properties of the Nonlinear Elements

Connectivity	Properties	No. of GRID Points	No. of Gauss Points	Geometric Nonlinearity	Material Property			Static Loads			DATA Recovery				
					Nonlinear Isotropic	Anisotropic	Orthotropic	Thermal	Pressure	Gravity	Stress	Grid Point Stress	Force	Structure Plot	Contour Plot
BCONP BLSEG	BWIDTH BFRIC	variable		X							X		X		
CBEAM	PBEAM	3	16*	X	1†			X		X	X		X	X	
	PBCOMP														
CBUSH	PBUSH	1 or 2			X	X					X		X	X	X
	PBUSHT														
CBUSH1D	PBUSH1D	1 or 2		X	X						X		X	X	X
CGAP	PGAP	3	0								X		X		
CHEXA	PSOLID	8	8	X	1	9		X	X	X	X	X		X	
	PLSOLID	8 20‡	8 27	X	MATHP			X	X	X	X			X	
CONROD	---	2	1	X	1			X		X	X	X		X	X
CPENTA	PSOLID	6	6	X	1	9		X	X	X	X	X		X	
	PLSOLID	6 15‡	6 21	X	MATHP			X	X	X	X			X	
CQUAD4	PSHELL	4	5*	X**	1	2	8	X	X	X	X	X	X	X	X
	PCOMP														
	PLPLANE	4	4	X	MATHP			X		X	X			X	
CQUAD8	PLPLANE	8‡	9	X	MATHP			X		X	X			X	
CQUAD	PLPLANE	9‡	9	X	MATHP			X		X	X			X	
CROD	PROD	2	1	X	1			X		X	X		X	X	
CTETRA	PSOLID	4,10	1	X	1	9		X	X	X	X	X		X	
	PLSOLID	4 10‡	1 5	X	MATHP			X	X	X	X			X	
CTRIA3	PSHELL	3	5*	X**	1	2	8	X	X	X	X	X	X	X	X
	PCOMP														
	PLPLANE	3	1	X	MATHP			X		X	X			X	
CTRIA6	PLPLANE	6‡	3	X	MATHP			X		X	X			X	
CTUBE	PTUBE	2	1	X	1			X		X	X		X	X	

Table 13-2 Summary of Properties of the Nonlinear Elements (continued)

Connectivity	Properties	No. of GRID Points	No. of Gauss Points	Geometric Nonlinearity	Material Property			Static Loads			DATA Recovery				
					Nonlinear Isotropic	Anisotropic	Orthotropic	Thermal	Pressure	Gravity	Stress	Grid Point Stress	Force	Structure Plot	Contour Plot
CQUADX	PLPLANE	4 9 [‡]	4 9	X	MATHP			X	X	X	X			X	
CTRIAX	PLPLANE	3 6 [‡]	1 3	X	MATHP			X	X	X	X			X	

* May be changed by the user.

† For the beam element, only elastic-perfectly plastic material is available.

‡ Center grid, if applicable, and midside grids may be omitted. For hyperelastic elements it is recommended that either all or no midside grids be omitted.

** Always in equilibrium in updated system.

Table Notes:

1. Integers listed under Material properties identify (i) on MATi Bulk Data entries.
2. MATS1 and CREEP may be attached to MAT1, MAT2, and MAT9.
3. Edge nodes are not applicable to nonlinear elements, except hyperelastic elements and the TET10 element.

Rudiments of User Interface

The input file consists of an optional NASTRAN statement, Executive Control Section, Case Control Section, and Bulk Data Section. All the features and principles for the user interface are common in MSC.Nastran; thus, all the features of the user interface for nonlinear analysis are compatible with those for the linear analysis. Any exceptions for nonlinear analysis will be explained in the relevant sections that follow.

Mechanical design is dictated by the strength, dynamic, and stability characteristics of the structure. MSC.Nastran provides the analysis capabilities of these characteristics with solution sequences, each of which is designed for specific applications. The type of desired analysis is specified in the Executive Control Section by using a solution sequence identification. SOLs 106 and 153 are designed for static, quasi-static, and buckling analyses. SOLs 129 and 159 provide nonlinear transient response analysis.

The basic input data required for a finite element analysis may be classified as follows:

- Geometric data
- Element data
- Material data
- Boundary conditions and constraints
- Loads and enforced motions
- Solution methods

The first three classes of data may not be changed during the course of an analysis whereas the last three classes of data may be changed during the analysis as defined in the Case Control Section. Examples of input data are given in [Listing 13-1](#) and [Listing 13-2](#) for nonlinear static and dynamic analysis, respectively.

Listing 13-1 Example Input Data for Nonlinear Static Analysis

```

SOL 106                $ NONLINEAR STATIC ANALYSIS
DIAG 8,50              $ DIAGNOSTIC PRINTOUT
CEND                   $ END OF EXECUTIVE CONTROL DATA
TITLE = TEST OF CTETRA ELEMENT (CUBE SUBJECT TO UNIAXIAL LOADING)
    DISP = ALL
    STRESS = ALL
    SPC = 100
SUBCASE 1
    SUBTITLE = ELASTIC -- LOAD TO 850. PSI
    LABEL = LOAD TO YIELD
    LOAD = 50
    NLPARM = 50
SUBCASE 2
    SUBTITLE = PLASTIC -- LOAD TO 1000. PSI
    LABEL = LOAD BEYOND YIELD
    LOAD = 100
    NLPARM = 100
SUBCASE 4
    SUBTITLE = ELASTIC -- UNLOAD COMPLETELY TO 0. PSI
    LABEL = FULL UNLOAD
    LOAD = 200
    NLPARM = 200
$OUTPUT(PLOT) $ PLOT DATA ADDED FOLLOWING THIS LINE IF NEEDED
$ END OF CASE CONTROL DATA
BEGIN BULK
$ PARAMETERS FOR NONLINEAR ITERATION
NLPARM    50    1    AUTO    UPW    NO
NLPARM    100   8    SEMI    UPW    NO
NLPARM    200   2    AUTO    UPW    NO
.
.
.
ENDDATA

```

Listing 13-2 Example Input Data for Nonlinear Dynamic Analysis

```

SOL 129                $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50              $ DIAGNOSTIC PRINTOUT
CEND                   $ END OF EXECUTIVE CONTROL DATA
TITLE = GAP VERIFICATION BY NONLINEAR TRANSIENT ANALYSIS SEALL = ALL $ FOR ALL MATRICIES
  SET 1 = 201          $ ,202,200
  SET 3 = 22           $ ,23
  SET 4 = 20           $ CONROD
    DISP = 1
    STRESS = 3         $ FOR GAP FORCES
    ELFORCE = 4
  LOADSET = 20
  `SPC = 100
SUBCASE 1              $ UP TO 0.5 SECONDS
  DLOAD = 210
  TSTEPNL = 22
SUBCASE 2              $ UP TO 1 SECOND
  DLOAD.= 210
  TSTEPNL = 21
OUTPUT(XY PLOT)
  CSCALE = 1.3
  XTITLE = TIME IN SECONDS
  YTITLE = DISPLACEMENT IN INCHES
  YMIN = 0.
  YMAX = 15.
  XY PLOT DISP RESP/201(T1),202(T1),200(T1)
$ END OF CASE CONTROL DATA
BEGIN BULK
TSTEPNL    21      200      .0025      2      AUTO      10
.
.
.
ENDDATA

```

Case Control Section

The Case Control Section is used to define input data sets, output features, and subcases. The subcase structure in MSC.Nastran provides a unique means of changing loads, boundary conditions, and solution methods by making selections from the Bulk Data. Confining the discussion to SOL 106 and SOL 129, loads and solution methods may change from subcase to subcase on an incremental basis. However, constraints can be changed from subcase to subcase only in the static solution sequence. As a result, the subcase structure determines a sequence of loading and constraint paths in a nonlinear analysis. The subcase structure also allows the user to select and change output requests for printout, plot, etc., by specifying set numbers with keywords. Any selections made above the subcase specifications are applicable to all the subcases. Selections made in an individual subcase supersede the selections made above the subcases. [Table 13-3](#) summarizes the Case Control commands for nonlinear analysis.

Table 13-3 Summary of Nonlinear Case Control Commands

Load Selection	
LOAD	Selects static loading condition.
CLOAD	Selects static load combination for superelements.
DLOAD	Selects dynamic loading conditions.
LOADSET	Selects static load sets defined on the Bulk data LSEQ.
NONLINEAR	Selects nonlinear loading (NOLINi) for transient response.
IC	Selects initial conditions for transient response.
Solution Method Selection	
METHOD	Selects methods for eigenvalue analysis.
NLPARM	Selects iteration methods for nonlinear static analysis.
TSTEPNL	Selects iteration methods for nonlinear transient analysis.
Output Requests	
DISPLACEMENT	Requests output for displacements of physical points.
VELOCITY	Requests output for velocities of physical points.
ACCELERATION	Requests output for acceleration of physical points.
ELFORCE	Requests output for element forces.
STRESS	Requests output for element stresses.
BOUTPUT	Requests output for slideline contact.
SPCFORCES	Requests output for constraint forces of SPC points.
NLLOAD	Requests output for NOLINi in transient response.
OUTPUT(PLOT)	Requests the beginning of the plotter output.
Superelement Control	
SUPER	Specifies the superelement identification number and the load sequence number.
SEALL	Combines the functions of SEMG, SELG, SEKR, SEMR, and SELR.
SEKR	Specifies the superelement identification numbers for which stiffness matrices will be assembled and reduced.
SELG	Specifies the superelement identification numbers for which load vectors will be generated.

Table 13-3 Summary of Nonlinear Case Control Commands (continued)

SELR	Specifies the superelement identification numbers for which the static load matrices will be assembled and reduced.
SEMG	Specifies the superelement identification numbers for which stiffness, mass, and damping matrices will be generated.
SEMR	Specifies the superelement identification numbers for which the mass and damping matrices will be assembled and reduced.

Bulk Data Section

All details of the input data are specified in the Bulk Data Section. Constraint, load, and solution method Bulk Data entries are not used unless they are selected in the Case Control Section. The nonlinear Bulk Data entries are summarized in [Table 13-4](#). All the input data designed specially for nonlinear analysis will be described in the following sections.

Table 13-4 Summary of Nonlinear Bulk Data Entries

Element Connectivity	
CBEAM	Defines connection for beam element.
CBUSH	Defines connections for generalized spring and damper.
CBUSH1D	Defines connections for rod type spring and damper.
CGAP	Defines connection for gap or frictional element.
CHEXA	Defines connection for six-sided solid element.
CONROD	Defines connection and properties for rod.
CPENTA	Defines connection for five-sided solid element.
CQUAD4	Defines connection for quadrilateral element with plane strain or bending and membrane stiffness.
CQUAD8 and CQUAD	Defines connection for plane strain hyperelastic quadrilateral element.
CROD	Defines connection for rod with axial and torsional stiffness.
CTETRA	Defines connection for four-sided solid element.
CTRIA3	Defines connection for triangular element with plane strain or bending and membrane stiffness.
CTRIA6	Defines connection for hyperelastic plane strain triangular element.

Table 13-4 Summary of Nonlinear Bulk Data Entries (continued)

CQUADX	Defines connection for axisymmetric hyperelastic quadrilateral element.
CTRIAX	Defines connection for axisymmetric hyperelastic triangular element.
CTUBE	Defines connection for a tube.
Element Properties	
PBCOMP	Defines properties for composite CBEAM.
PBEAM	Defines properties for CBEAM.
PBUSH and PBUSHT	Defines properties for CBUSH.
PBUSH1D	Defines properties for CBUSH1D.
PCOMP	Defines properties for composite material laminate.
PGAP	Defines properties for CGAP.
PROD	Defines properties for CROD.
PSHELL	Defines properties for CTRIA3 and CQUAD4.
PSOLID	Defines properties for CHEXA, CPENTA and CTETRA.
PTUBE	Defines properties for CTUBE.
PLPLANE	Defines properties for hyperelastic plane elements CTRIA3, CTRIA6, CQUAD4, CQUAD8, CQUAD, CQUADX and CTRIA6.
PLSOLID	Defines properties for hyperelastic solid elements CHEXA, CPENTA and CTETRA.
Material Properties	
CREEP	Defines creep material properties.
MAT2	Defines anisotropic material properties for shell elements.
MAT8	Defines orthotropic material properties for shell elements.
MAT9	Defines anisotropic material properties for solid elements.
MATS1	Defines properties for plastic and nonlinear elastic material.
MATHP	Defines properties for hyperelastic material.
TABLEDi	Defines a function for internal load versus disp or velocity for bush element.
TABLES1	Defines a function for stress-dependent material properties or experimental data for hyperelastic material.
TABLEST	Combines many TABLES1 entries for temperature dependent material properties.

Table 13-4 Summary of Nonlinear Bulk Data Entries (continued)

Slideline Contact	
BCONP	Defines contact parameters.
BLSEG	Defines boundary line segments.
BFRIC	Defines frictional properties for slideline contact.
BWIDTH	Defines boundary line segments width/thickness.
BOUTPUT	Defines grid points where contact output is requested.
Constraints	
SPC	Defines single-point constraints and enforced displacements.
SPC1	Defines single-point constraints.
MPC	Defines a linear relationship for two or more degrees-of-freedom.
Loads	
CLOAD	Defines a static load combination for superelement loads.
FORCEi	Defines concentrated load at grid point.
LSEQ	Defines static load sets for dynamic analysis.
MOMENTi	Defines moment at a grid point.
NOLINi	Defines nonlinear transient load.
PLOAD	Defines pressure loads on CQUAD4, CTRIA3, CHEXA, CPENTA, and CTETRA. Should not be used for hyperelastic plane elements CQUAD4, CQUAD8, CQUAD, CTRIA3, and CTRIA6 or for hyperelastic CHEXA, CPENTA, CTETRA with midside nodes.
PLOAD2	Defines pressure loads on shell elements, CQUAD4 and CTRIA3. Not available for hyperelastic elements.
PLOAD4	Defines pressure loads on surfaces of CHEXA, CPENTA, CTETRA, CTRIA3 and CQUAD4 elements. Not available for hyperelastic plane elements CQUAD4, CQUAD8, CQUAD, CTRIA3, and CTRIA6.
PLOADX1	Defines pressure loads on axisymmetric elements CQUADX and CTRIAX.
RFORCE	Defines load due to centrifugal force field.
TEMP	Defines temperature at grid points.
TEMPPi	Defines temperature field for surface elements.
TEMPRB	Defines temperature field for line elements.
TIC	Specifies initial values for displacement and velocity.

Table 13-4 Summary of Nonlinear Bulk Data Entries (continued)

TLOADi	Defines loads as a function of time.
Solution Methods	
NLPARM	Defines iteration methods for nonlinear static analysis.
NLPCI	Defines arc-length methods for nonlinear static analysis.
TSTEPNL	Specifies integration and iteration methods for nonlinear transient analysis.
EIGB or EIGRL	Defines eigenvalue extraction method for buckling analysis.

Parameters

Parameters constitute an important part of the input data because they maintain communications among the user, DMAP, and modules. Parameters are single-valued input data used for requesting special features or specifying miscellaneous data.

Users can change the initial values of parameters by specifying the PARAM entry in the Bulk Data Section or in the Case Control Section. Many options available in MSC.Nastran are controlled by user-specified parameters. [Table 13-5](#) lists the parameters used in nonlinear Solution Sequences 106, 129, 66, 99 and 64.

Table 13-5 PARAMeters Used in Nonlinear Solution Sequences

B = Must be specified in the Bulk Data Section only.

E = May be specified in either the Bulk Data or Case Control Section.

Parameter Name	Solution Sequence			Default	Description
	64	106 or 66	129 or 99		
ADPCON		E	E	1.0	Adjusts automatic penalty values for slideline contact regions.
AUTOSPC	B	E	E	NO	Specifies automatic single-point-constraints.
AUTOSPCR		E	B	NO	Specifies automatic SPC's for residual structure.
BAILOUT		E	E	0	Terminates superelement program with singularities.
BETA			B	0.33333	Specifies control factor for Newmark-Beta method.
BUCKLE	E			-1	Selects nonlinear buckling analysis for restarts.
CB1,2		E		1.0	Selects superelement damping matrix coefficients.
CK1,2,3	E	E		1.0	Selects superelement stiffness matrix coefficients.
CM1,2	E	E		1.0	Selects superelement mass matrix coefficients.
CNSTRT	B			1	Controls restarts for constraint change.
COUPMASS	B	E	E	-1	Generates coupled rather than lumped mass matrices.

Table 13-5 PARAMeters Used in Nonlinear Solution Sequences (continued)

B = Must be specified in the Bulk Data Section only.

E = May be specified in either the Bulk Data or Case Control Section.

Parameter Name	Solution Sequence			Default	Description
	64	106 or 66	129 or 99		
CURV	E			-1	Computes stress data in a material coordinate system.
CURVPLOT	E			-1	Requests x-y (curve) plots.
DBDICT	B	B	B	-1	Prints database dictionary at the end and beginning of the run.
DBDRNL		E	E	0	Prints database dictionary in the nonlinear loop (SOLs 66 and 99 only).
DLOAD	B	B	B	0	Indicates no other changes except for loads (SOLs 64 through 99 only).
DOPT	E			0	Controls x spacing of curves for CURVPLOT.
ERROR		E	E	-1	Terminates run for superelement errors.
EST	B	E	E	2	Requests element measure and volume computation.
FACTOR	B	B	B	10,000	Generates SEQID on SEQGP entry.
FKSYMFAC		B		1.0	Controls symmetrization of stiffness.
FOLLOWK		E	E	YES	Include follower force stiffness.
GPECT	B	E	E	-1	Prints all elements connected to each grid point.
GRDPNT	B	E	E	-1	Executes Grid Point Weight Generator.
G			E	0.0	Specifies uniform structural damping coefficient.
INRLM			E	0	Requests appending of inertia relief modes.
KDIAG	E			1.0	Applies value of a spring constant to nonlinear stiffness matrix diagonals.
K6ROT	E	E	E	100.0	Assigns stiffness to normal rotation of CQUAD4 and CTRIA3 elements.
LANGLE		E	E	1	Selects gimbal angle approach for large rotations if 1 and rotation vector approach if 2.
LGDISP		E	E	-1	Selects large displacement effects.
LOOPID		E	E	0	Specifies loop ID in the database for restarts.
MAXLP			E	5	Maximum number of iterations for internal loop.
MAXRATIO	E	E	E	1.0E + 7	Determines the singularity of the stiffness matrix.
MODEL	B	B	B	0	Stores several models in the database.
MPCX	B	B	B	0	Controls MPC and rigid element processing on restarts.
NDAMP			E	0.025	Specifies numerical damping in ADAPT method.
NINTPTS	E			10	Requests CURV to interpolate over all elements.
NLAYERS		B	B	5	Specifies number of integration points through thickness for CQUAD4 and CTRIA3.
NLTOL		B		2	Sets accuracy tolerances

Table 13-5 PARAMeters Used in Nonlinear Solution Sequences (continued)

B = Must be specified in the Bulk Data Section only.

E = May be specified in either the Bulk Data or Case Control Section.

Parameter Name	Solution Sequence			Default	Description
	64	106 or 66	129 or 99		
NMLOOP		E		0	Saves tangent stiffness in SOL 106 for normal modes and other linear solution sequences.
NOCOMPS		E		1	Suppresses composite stress output.
NODATA	B	B	B	0	Indicates no changes in Bulk Data (SOLs 64 through 99 only).
NOTRED			E	0	Controls t-set reduction (SOL 99 only).
OG	E			0	Calculates stress/strain data at grid points for CURV.
OLDELM	B			1	Computes element forces using old method.
OUTOPT	E			0	Sets format for output quantities computed by CURV.
OUTPUT	B			0	Jumps to data recovery.
PDRMSG			E	1	Suppresses data recovery messages.
PLOT	B	E	E	1	Jumps to undeformed plot.
PLOTSUP		E	E	0	Makes undeformed plots for superelements.
PLTMSG		E	E	1	Suppresses undeformed plot messages.
POST		E	E	1	Stores post-processing data blocks in database.
PRGPST	B	E	E	YES	Suppresses singularity printout.
PROUT	B	E	E	-1	Suppresses execution and printout from ELTPRT.
PRPHIVZ			E	1.0E + 37	Prints eigenvector matrix for general dynamic reduction.
RESDUAL		E	E	1	Skips to nonlinear operations (Phase 2) (SOLs 66 through 99 only).
SDATA			E	1	Obtains additional solution set output (SOL 99 only).
SEFINAL		B	B	-1	Indicates no data changes if change SEFINAL.
SEMAP		B	B	SEMAP	Determines partitioning of superelements.
SEMAPPRT		B	B	3	Prints SEMAP table.
SENAME		E	E	SENAME	Identifies superelement by name.
SEQOUT	B	B	B	0	Controls output options for OLDSEQ.
SKPLOAD	E			1	Skips load vector calculation.
SKPMTRX	E			1	Skips matrix reduction and decomposition.
SLOOPID			B	0	Specifies loop ID from SOL66 database for restart.
SOLID	B	B	B	0	Stores several solutions in same database.
SPCGEN	B	E	E	0	Places automatic SPC's in SPC1 format.
SSG3	B			0	Skips operations prior to FBS.
START	B	B	B	0	Defines number of grid points for OLDSEQ.
STIME			E	0.0	Specifies time step for restarts.

Table 13-5 PARAMETERS Used in Nonlinear Solution Sequences (continued)

B = Must be specified in the Bulk Data Section only.

E = May be specified in either the Bulk Data or Case Control Section.

Parameter Name	Solution Sequence			Default	Description
	64	106 or 66	129 or 99		
SUBID	B	E		1	Specifies subcase ID for restarts.
SUBSKP	B			0	Skips a number of subcases for restarts.
SUPER	B	B	B	0	Deletes ungrouped grid points.
TABID			E	2	Controls punch for response spectra.
TABS		E		0.0	Converts units of temperature input to absolute.
TESTNEG	E	E		-1/-2/1	Tests for negative terms on factor diagonal of matrix.
TESTSE	E			1.0E + 36	Tests for strain energy for iteration control.
TSTATIC			E	-1	Selects static solution with TSTATIC=1 using ADAPT method in TSTEPNL.
USETPRT	B	E	E	-1	Selects output type for degrees-of-freedom.
USETSEL	B	E	E	0	Controls sets for row list output option.
WTMASS	B	E	E	1.0	Multiplies terms of structural mass matrix by value.
W3,W4			E	0.0	Selects frequency for conversion of damping.

Nonlinear Element Output Codes for Plotting

Nonlinear elements (which reference nonlinear geometry and/or material) have different plot codes than linear elements. The plot codes be found in “[Item Codes](#)” on page 777 of the *MSC.Nastran Quick Reference Guide*.

Nonlinear Characteristics and General Recommendations

Modeling for nonlinear analysis should follow the guidelines for good modeling practice pertaining to linear analysis, which are summarized as follows:

- The analyst should have some insight into the behavior of the structure to be modeled; otherwise, a simple model should be the starting point.
- Substructuring should be considered for the modularity of the model and/or synergism between projects and agencies involved.
- The size of the model should be determined based on the purpose of the analysis, the trade-offs between accuracy and efficiency, and the scheduled deadline.

- Prior contemplation of the geometric modeling will increase efficiency in the long run. Factors to be considered include selection of coordinate systems, symmetric considerations for simplification, and systematic numbering of nodal points and elements for easy classification of locality.
- Discretization should be based on the anticipated stress gradient, i.e., a finer mesh in the area of stress concentrations.
- Element types and the mesh size should be carefully chosen. For example, avoid highly distorted and/or stretched elements (with high aspect ratio); use CTRIA3 and CTETRA only for geometric or topological reasons.
- The model should be verified prior to the analysis by some visual means, such as plots and graphic displays.

Nonlinear analysis requires better insight into structural behavior. First of all, the type of nonlinearities involved must be determined. If there is a change in constraints due to contact during loading, the problem may be classified as a boundary nonlinear problem and would require CGAP elements or slide line contact algorithm. The material nonlinearity is characterized by material properties. However, the material nonlinear effects may or may not be significant depending on the magnitude and duration of the loading, and occasionally on environmental conditions. The anticipated stress level would be a key to this issue. The geometric nonlinearity is characterized by large rotations which usually cause large displacements. Intuitively, geometric nonlinear effects should be significant if the deformed shape of the structure appears distinctive from the original geometry without amplifying the displacements. There is no distinct limit for large displacements because geometric nonlinear effects are related to the dimensions of the structure and the boundary conditions. The key to this issue is to know where the loading point is in the load-deflection curve of the critical area.

SOL 106 or SOL 129 must be used if the model contains any of the following:

- A parameter LGDISP (for geometric nonlinearity).
- Any hyperelastic element (PLPLANE or PLSOLID property entry, pointing to a MATHP material).
- Any CGAP element (CGAP and PGAP).
- Slideline contact (BCONP, BLSEG, and/or BFRIC).
- Any active nonlinear material data (specified on MATS1 and/or CREEP).
- Any combination of the above.

The model may consist of superelements, but only the residual structure (superelement 0) may consist of nonlinear elements mixed with any type of linear elements. All the gaps, slideline contact regions, and hyperelastic elements are always actively nonlinear if included in the residual structure. However, other potentially nonlinear elements in the residual structure become actively nonlinear only if a parameter, LGDISP, is used and/or if they use the nonlinear material data specified on the MATS1 and/or CREEP data entries.

The subcase structure of MSC.Nastran may be used in SOLs 106 and 129 to characterize a nonproportional loading path. Cyclic loading is one example. Some nonlinear problems, such as plastic deformation, creep deformation, and contact problems with frictional forces, are path-dependent; others, such as hyperelastic or purely geometric nonlinear, are path-independent. An incremental process may not be required for a purely geometric nonlinear problem. For computational efficiency, however, the analysis of such a problem does frequently need incremental solutions.

With these points in mind, additional recommendations are imperative for nonlinear analysis:

- Identify the type of nonlinearity and localize the nonlinear region for computational efficiency. If unsure, perform a linear analysis by SOL 101 (or 1) or SOL 109 (or 27) prior to the nonlinear analysis by SOL 106 or SOL 129, respectively.
- Segregate the linear region by using superelements and/or linear elements if possible. Notice that the potentially nonlinear elements can be used as linear elements.
- The nonlinear region usually requires a finer mesh. Use a finer mesh if severe element distortions or stress concentrations are anticipated.
- Be prepared for restarts with the database properly stored in cases of divergence and changing constraints or loading paths via subcases.
- The subcase structure should be utilized properly to divide the load or time history for convenience in restarts, data recovery, and database storage control, not to mention changing constraints and loading paths.
- The load or time for the subcase should then be further divided into increments, not to exceed 20 load steps (for SOL 106) or 200 time steps (for SOL 129) in each subcase.
- Many options are available in solution methods to be specified on NLPARM (for SOL 106) or TSTEPNL (for SOL 129) data entries. The defaults should be used on all options before gaining experience.

- Caution should be exercised in specifying CGAP element properties. In particular, the closed gap stiffness should not exceed the stiffness of the adjacent degree-of-freedom by 1000 times.
- Normal rotation for CQUAD4 and CTRIA3 elements should be restrained by a parameter K6ROT when the geometric nonlinearity is involved. Default value for parameter K6ROT is 100.
- Understand the basic theory of plasticity or creep before using these capabilities.
- Caution should be exercised in preparing input data for creep, because it is unit dependent.
- For the transient analysis by SOL 129, some damping is desirable and the massless degree-of-freedom should be avoided.
- The time step size for a transient response analysis should be carefully determined based on the highest natural frequency of interest because it has significant effects on the efficiency as well as the accuracy.

For any anomalies, refer to the MSC.Nastran error list and general limitations list.

13.3.1 Material Nonlinearity

Material nonlinearity becomes an issue in the analysis if the stress-strain relationship is nonlinear. Constitutive equations characterize the macroscopic behavior of the material, which stems from its microscopic constitution. All engineering materials are inherently nonlinear. It is, however, not feasible to characterize a nonlinear material by a single set of equations for the entire range of environmental conditions, such as loading, temperature, and rate of deformation. Depending on the application or the situation, the material behavior may be idealized or simplified to account for certain effects, which are important in the analysis.

The linear elastic (Hookean) material is the simplest case of all. As the strain exceeds the infinitesimal range, the material exhibits nonlinear behavior: nonlinear elastic if it is recoverable and plastic if it is irrecoverable. If the temperature effects on the material properties become an important issue, coupling between thermal and mechanical behavior should be properly taken into consideration, which is the realm of thermo-elasticity and thermo-plasticity. If the strain-rate has significant effects on the material response, we have to resort to the theories of visco-elasticity and visco-plasticity. Anisotropic effects add complexity to the material model.

Development of material constitutive models has been prolific in recent years. Detailed discussion in this section will be confined to the capabilities provided in MSC.Nastran. The following nonlinear materials are available:

- Nonlinear elastic
- Hyperelastic
- Elastic-plastic
- Visco-elastic
- Temperature dependent

13.3.2 Nonlinear Elastic Material

Typical stress-strain relationship for a nonlinear elastic material is shown in [Figure 13-2](#). Notice that both the loading and unloading occur along the same path.

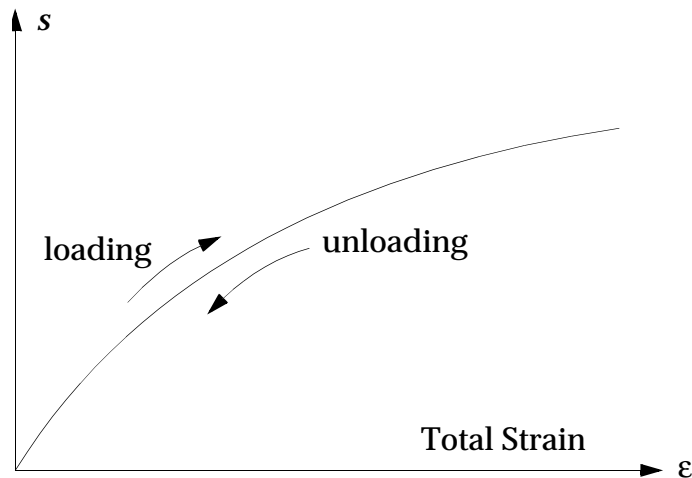


Figure 13-2 Nonlinear Elastic Material

The nonlinear elastic theory implemented is useful to predict multiaxial stress state for materials for which only the tension and compression stress-strain data is available. The stress-strain data in tension may be different from that in compression. The theory implemented in MSC.Nastran is adequate only for small strains (less than 20%) in multiaxial stress state as it is not based on the classical theory of finite elasticity. For large strains in multiaxial state, use of hyperelastic material is strongly recommended.

13.3.3 Hyperelastic Material

Descriptions

The hyperelastic model is used to analyze elastomers up to a large strain. Elastomers are natural or synthetic rubberlike materials that have the ability to undergo deformation under the influence of a force and regain their original shape once the force is removed (unless damage occurs). They are used extensively in many industries, because of their wide availability and low cost. Some examples of “real world” analyses in the automotive industry include o-rings, bushings, gaskets, seals, rubber boots, and tires.

The hyperelastic constitutive model available in MSC.Nastran is based on a generalized Rivlin strain energy function expressed as follows:

$$U = \sum_{i+j=1}^{N_a} A_{ij}(\bar{I}_1 - 3)(\bar{I}_2 - 3)^j + \sum_{i=1}^{N_d} D_i(J - 1 - \alpha_v(T - T_0))^{2i} \quad \text{Eq. 13-8}$$

where A_{ij} are the material constants related to shear deformation (distortion) and D_i are the material constants related to volumetric deformation; The strain energy function is a potential for the stresses. For example, the Cauchy stresses; i.e., components of the stress vector $t = dP/dA$ where dA is an infinitesimal area in the deformed configuration and P is the force, are obtained as follows:

$$\sigma = \frac{2}{J} \mathbf{F} \frac{\partial U}{\partial \mathbf{C}} \mathbf{F}^T \quad \text{Eq. 13-9}$$

where $\mathbf{F} = dx/dX$ is the deformation gradient; x and X are the positions of a material point in the deformed and the undeformed configurations, respectively; $\mathbf{C} = \mathbf{F}^T \mathbf{F}$ is the right Cauchy-Green deformation tensor; $J = \det \mathbf{F}$ is the volume ratio dV/dV_0 ; and \bar{I}_1 and \bar{I}_2 are the first and second distortional strain invariants,

$$\bar{I}_1 = \bar{\lambda}_1^2 + \bar{\lambda}_2^2 + \bar{\lambda}_3^2 \quad \bar{I}_2 = \bar{\lambda}_1^{-2} + \bar{\lambda}_2^{-2} + \bar{\lambda}_3^{-2} \quad \text{Eq. 13-10}$$

where $\bar{\lambda}_l$ are principal stretches l/l_0 of the distortional deformation; l is the current and l_0 is the original length along a principal direction; $\bar{\lambda}_l^2$ are principal values of $\bar{\mathbf{C}} = J^{-2/3} \mathbf{C}$; α_v is the volumetric coefficient of thermal expansion; $\alpha_v = 3\alpha$, where α is the coefficient of linear thermal expansion of an isotropic material; T is the current and T_0 is the initial temperature.

The use of a strain energy function implies isotropy in the undeformed configuration. The model can have up to 25 material constants.

Special cases of the material law are:

Mooney-Rivlin Material, $N_a = 1$. Two-parameter model that results in a constant shear modulus G .

Neo-Hookean Material, $N_a = 1$, $A_{01} = 0.0$. One-parameter model that results in a constant shear modulus G .

A curve-fitting algorithm based on least squares fitting is available to determine the material constants A_{ij} and D_i based on experimental data up to the fifth-order polynomial form (N_a and/or N_d up to 5) for the following deformations of the material:

1. Simple tension/compression

The simple tension/compression test is characterized by a prescribed axial force or displacement. X values in the TABLES1 entry which contains experimental data must be stretch ratios l/l_0 . Y values must be values of the engineering stress F/A_0 .

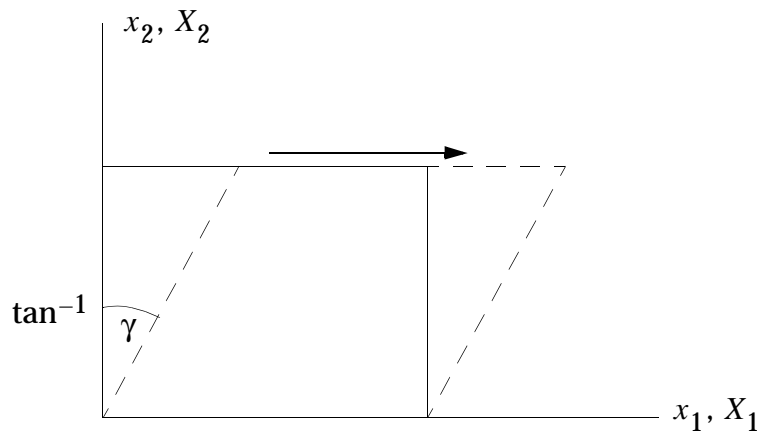
2. Equibiaxial tension

The equibiaxial tension test is characterized by prescribed equal tensile stresses or elongations in two directions and is typically produced by the pressure of a spherical membrane. X values in the TABLES1 entry which contains experimental data must be stretch ratios l/l_0 . Y values must be values of the engineering stress F/A_0 , l = current length, F = current value of the force, l_0 and A_0 = initial length and cross-sectional area, respectively. In the case of pressure of a spherical membrane, the engineering stress is given by $Pr_0\lambda^2/2t_0$ where P = current value of the pressure and r_0 , t_0 = initial radius and thickness.

3. Simple shear

The simple shear test is obtained by a prescribed shearing force or displacement of a cube. This is not a homogeneous deformation. It is approximated by one that is homogeneous isochoric as follows:

$$x_1 = X_1 + \gamma X_2 \quad x_2 = X_2 \quad x_3 = X_3$$



where x_1, x_2, x_3 are the coordinates of the deformed body and X_1, X_2, X_3 are the coordinates of the body in its undeformed configuration. The user should be aware of this approximation. X values in the TABLES1 entry which contains experimental data must be values of the shear tangent γ . Y values must be values of the engineering shear stress F/A_0 . It is recommended that simple shear be used along with another test case; otherwise, the coefficients cannot be determined uniquely.

4. Pure shear

The pure shear test is a more direct way of imposing a state of shear deformation and is obtained by a prescribed uniform stretch λ_1 of a specimen, which is long in the 2-direction leaving the 3-direction stress free, or by holding the 2-direction fixed while applying strains in the 1-direction of a thin rectangular sheet, as in the classical experiments of Rivlin and Saunders. If the material is incompressible, this results in $\lambda_3 = 1/\lambda_1$, which is characteristic of a state of shear deformation. It is recommended that pure shear be used along with another test case; otherwise, the coefficients cannot be determined uniquely. X and Y values, respectively, in the TABLES1 entry which contains experimental data must be stretch ratios $\lambda_1 = l/l_0$ and values of the nominal stress $F/A_0 l = \text{current length}$, $F = \text{current value of the force}$, l_0 and $A_0 = \text{initial length and cross-sectional area}$, respectively in the 1-direction.

5. Volumetric compression

Pure volumetric compression data may be used to obtain fitted values of the material constants related to volumetric deformation. X values in the TABLES1 entry which contains experimental data must be values of the volume ratio $J = \lambda^3$, where $\lambda = l/l_0$ is the stretch ratio in all three directions; Y values must be values of the pressure, assumed positive in compression.

The first four test cases (experiments) are used to obtain the material constants A_{ij} . The last test case (volumetric compression) is used to obtain the material constants D_i . In obtaining the material constants, compressibility is taken into account, in the case of simple tension/compression and equibiaxial tension, when the volumetric part of the strain energy function is of the order of one.

User Interface

The MATHP material entry defines the hyperelastic material. The MATHP entry consists of five continuation entries that define the A_{ij} and D_i material constants of the strain energy function. The continuation entries are not required if a Neo-Hookean or a Mooney-Rivlin material is modeled.

If the material constants are not known, the curve-fitting algorithm is invoked if the sixth continuation entry of the MATHP entry contains nonblank fields, TAB1, TAB2, TAB3, TAB4, and TABD. The TABi fields reference experimental data that is contained in TABLES1 entries. The order of the strain energy polynomial function is specified using the NA and ND fields of the MATHP entry for the distortional and volumetric parts, respectively. The curve-fitting algorithm computes the material constants from the supplied data, and the analysis continues using the produced constants.

Output Description

The curve-fitting algorithm computes the material constants and prints the results in an updated MATHP entry image in the output file. The image is printed using a 16-character field and therefore is convenient to directly input the produced MATHP entry into a data file in large-field format. An example of the produced output is shown in [Listing 13-3](#).

Guidelines

If the strains are small, the results of the hyperelastic elements should compare with those of the linear elastic elements where the shear modulus and the bulk modulus $G \approx 2(A_{10} + A_{01})$ and the bulk modulus $K \approx 2D_1$.

Conventional Mooney-Rivlin and Neo-Hookean materials are fully incompressible. Full incompressibility is not presently available but may be simulated with a large enough value of D_1 . However, a value of D_1 higher than $10^3 \cdot (A_{10} + A_{01})$ is not recommended.

When determining the material constants from experimental data, it is best to use data from more than one test involving different kinds of deformation. A rank deficiency warning may occur in the least squares fitting algorithm if there is insufficient experimental data. It is best to try to avoid rank deficiency by providing more experimental data or by lowering the order of the strain energy polynomial.

Limitations

It is recommended that Poisson's ratio for the hyperelastic material be no greater than .4995, or, equivalently, that $D_1/(A_{10} + A_{01})$ does not exceed 1000. As incompressibility increases, convergence difficulties and/or ill conditioning of the stiffness matrix occurs.

The hyperelastic material cannot be defined for a small strain element, i.e., it cannot be referenced by a PSHELL or PSOLID Bulk Data entry.

Temperature dependence of the hyperelastic material constants is not supported.

Example: Curve-Fitting Algorithm

An example created to solve a fully nonlinear (finite deformation) CQUAD4 element under pure shear with material constants obtained from the simultaneous fitting of experimental data in simple tension, equibiaxial tension, and pure shear. Annotated Bulk Data input is shown in the following listing that illustrates the MATHP and TABLES1 entries, which contain the experimental data for curve fitting.

Listing 13-3

```

BEGIN BULK
$ PARAMETERS
PARAM DBDROPT 0
PARAM LGDISP 1
$
NLPARM 10 40 AUTO 1
$
$ DEFINE GEOMETRY
GRID 1 0. 0. 0. 123456
GRID 2 1. 0. 0. 23456
GRID 3 1. 0. 1. 2456
GRID 4 0. 0. 1. 12456
$ DEFINE ELEMENTS
CQUAD4 1 1 1 2 3 4
$ CONSTRAINTS AND LOADING
MPC 10 2 1 1. 3 1 -1.
MPC 10 4 3 1. 3 3 -1.
SPC 10 3 1
SPCD 100 3 1 5.
$ DEFINE PROPERTIES
$ LARGE STRAIN QUAD4 ELEMENT
PLPLANE 1 1 ISOP 2
$ HYPERELASTIC MATERIAL
$ D1 SPECIFIED
$ AIJ MATERIAL CONSTANTS TO BE DETERMINED FROM EXPERIMENTAL DATA
MATHP 1 1500. +MA101
+MA101 3 1 +MA102
+MA102 +MA103
+MA103 +MA104
+MA104 +MA105
+MA105 +MA106
$ TAB1 TAB2 TAB3 TAB4 TABD
+MA106 100 200 400
$
$ TAB1 DATA
$
$ Treloar[1944] data in simple tension. Nominal stresses in kgcm-2
$
TABLES1 100 +TA101
+TA101 1. 0. 1.125 1. 1.25 2. 1.5 3. +TA102
+TA102 1.525 4. 1.875 5. 2. 6. 2.25 7. +TA103
+TA103 2.5 8. 3. 9. 3.625 10. 4. 12.5 +TA104
+TA104 4.75 16. 5.25 20. 5.75 23. 6. 27. +TA105
+TA105 6.25 31. 6.5 34. 6.75 38.75 7. 42.5 +TA106
+TA106 ENDT
$
$ TAB2 DATA
$
$ Treloar [1944] data in equibiaxial tension. Nominal stresses in kg-cm-2
$
TABLES1 200 +TA201
+TA201 1.016 0.83 1.07 1.56 1.15 2.55 1.21 3.25 +TA202
+TA202 1.335 4.31 1.44 5.28 1.66 6.65 1.93 7.88 +TA203
+TA203 2.46 9.74 3. 12.64 3.4 14.61 3.77 17.33 +TA204
+TA204 4.1 20.11 4.32 22.40 4.54 24.41 ENDT
$
$ TAB4 DATA
$
$ Treloar* [1944] data in pure shear. Units of the nominal stress are kgcm-2

```

Listing 13-3 (continued)

```

$
TABLES1 400
+TA401 1.01 0.24 1.1 1.12 1.2 1.92 1.31 2.87 +TA401
+TA402 1.49 3.62 1.86 5.61 2.36 7.2 2.97 9.045 +TA402
+TA403 3.45 10.8 3.93 12.45 4.4 14.23 4.714 15.826 +TA403
+TA404 4.96 17.46 ENDT
ENDDATA

```

* See “References” on page 978.

The following listing illustrates the produced MATHP entry image that is printed in the output just after the ENDDATA entry image.

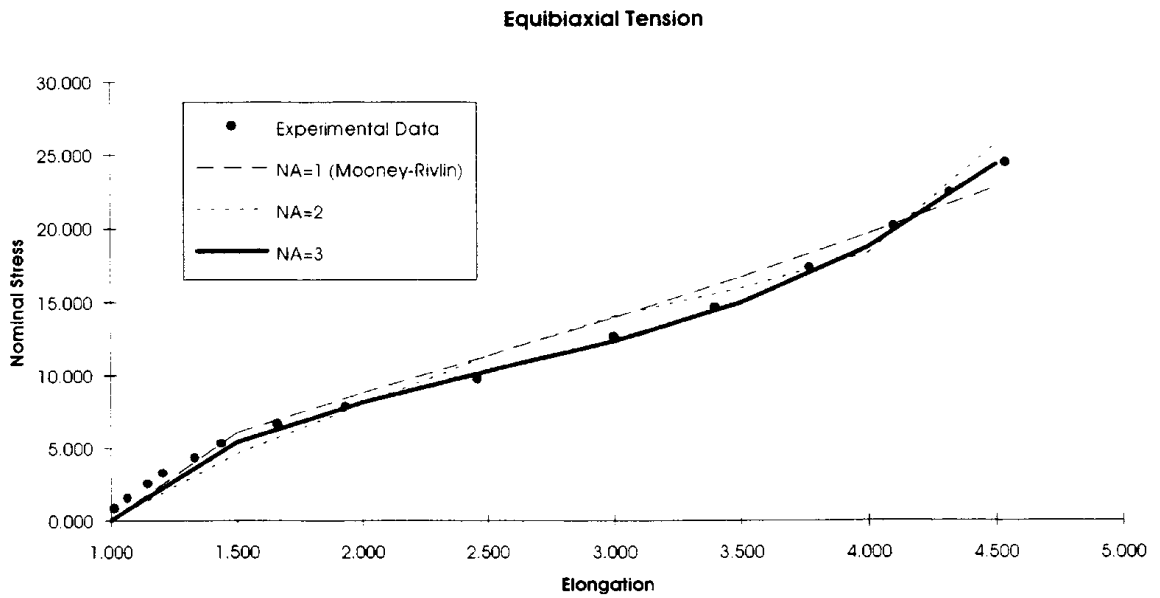
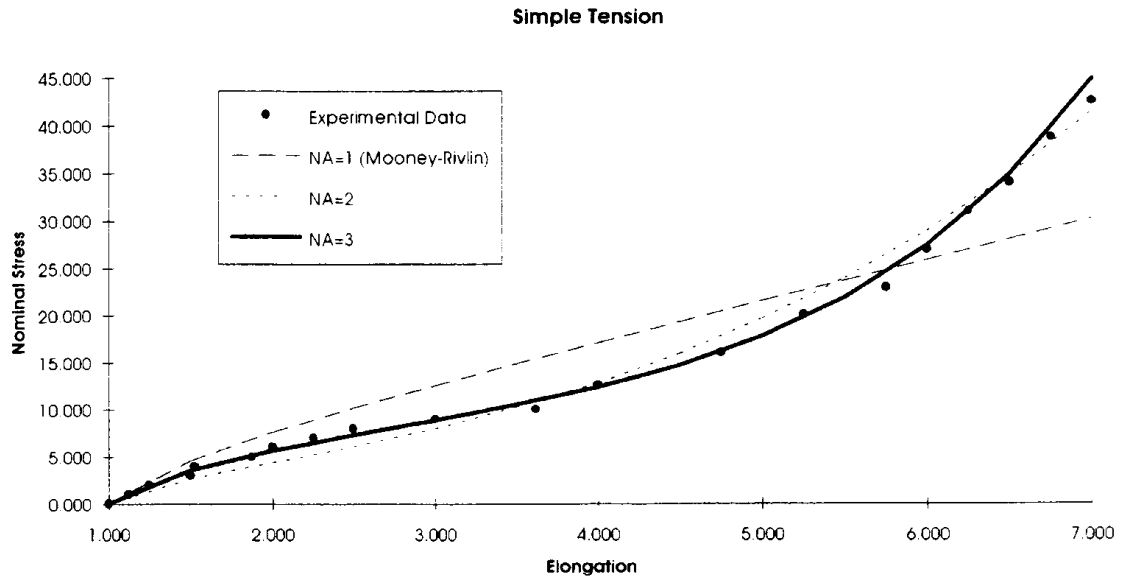
Listing 13-4

```

*** SYSTEM INFORMATION MESSAGE 6410 (IFP8). BEGIN PROCESSING OF MATHP ENTRY ID = 1
DISTORTIONAL PARAMETER FITTING.
SQUARE ROOT OF SUM OF THE ERRORS SQUARED = 0.25629E+01
UPDATED MATHP ENTRY
      1      1.54082      0.24046      1500.00000      0.00000      0.00000      0.00000      0.00000
      3      1
-0.00375      -0.01743      0.00104      0.00000
0.00029      0.00033      -0.00004      0.00000      0.00000
0.00000      0.00000      0.00000      0.00000      0.00000      0.00000
0.00000      0.00000      0.00000      0.00000      0.00000      0.00000      0.00000

```

Figure 13-3(a) through **Figure 13-3(c)** illustrate the curves obtained from this and similar files containing a large strain CHEXA in simple and equibiaxial tension with the material constants generated for the first-, second-, and third-order strain energy polynomial functions from simultaneous fitting of experimental data in simple tension, equibiaxial tension, and pure shear.



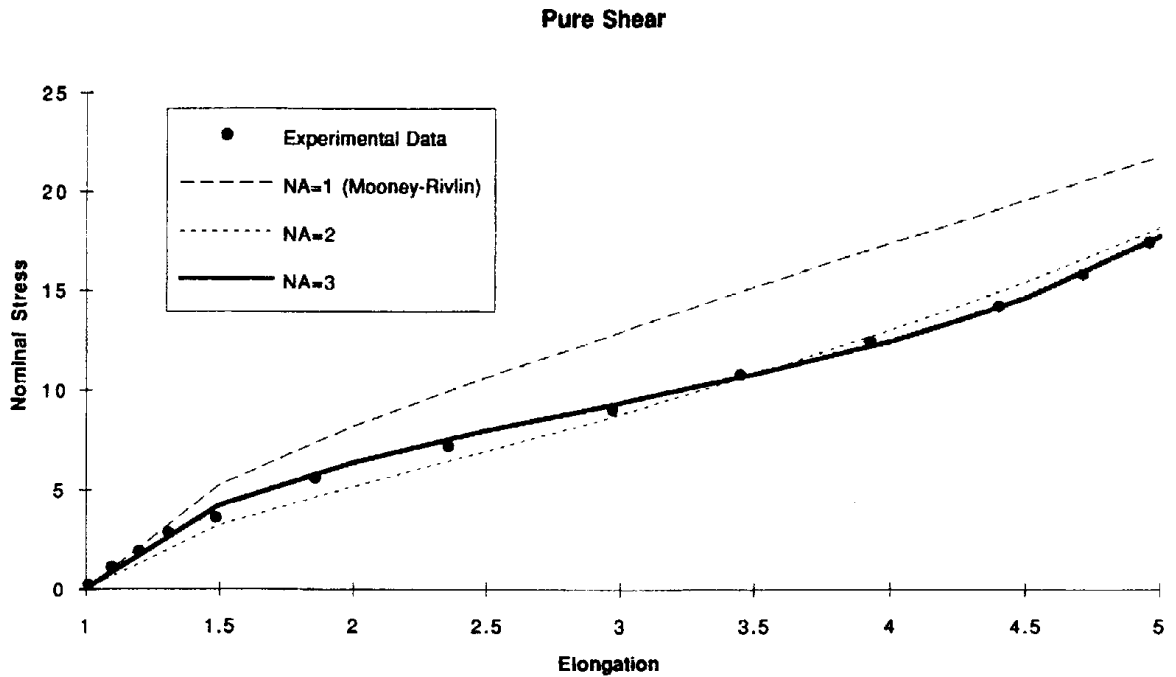


Figure 13-3 MSC.Nastran Curve-Fitting Algorithm for Hyperelastic Materials (c)

13.3.4 Elastic-Plastic Material

Before discussing the computational aspects associated with the various models, the definitions of some of the commonly used terms in plasticity are given below.

Yield Stress

Yield Stress is usually measured as the value of stress which produces the smallest measurable permanent strain.

Bauschinger Effect

When a plastically deformed specimen is unloaded, residual stresses on a microscopic scale remain and influence the plastic yielding for the different loading. If the previous strain was a uniform extension and the specimen is then reloaded in compression in the opposite direction, it is observed that yielding occurs at a much reduced stress. This is known as the Bauschinger Effect.

The phenomenon of Bauschinger effect can best be described with reference to **Figure 13-4** which shows an idealized stress-strain curve of a metal first deformed by uniform tension, the load removed and then reloaded in compression. According to one model (which is one extreme viewpoint), it is assumed that the elastic unloading range will be double the initial yield stress. If the initial yield stress in tension is σ_y and the specimen is loaded up to a stress σ_1 and unloaded, the plastic yielding will begin in compression at a stress σ_2 given by

$$\sigma_2 = \sigma_1 - 2\sigma_y \quad \text{Eq. 13-11}$$

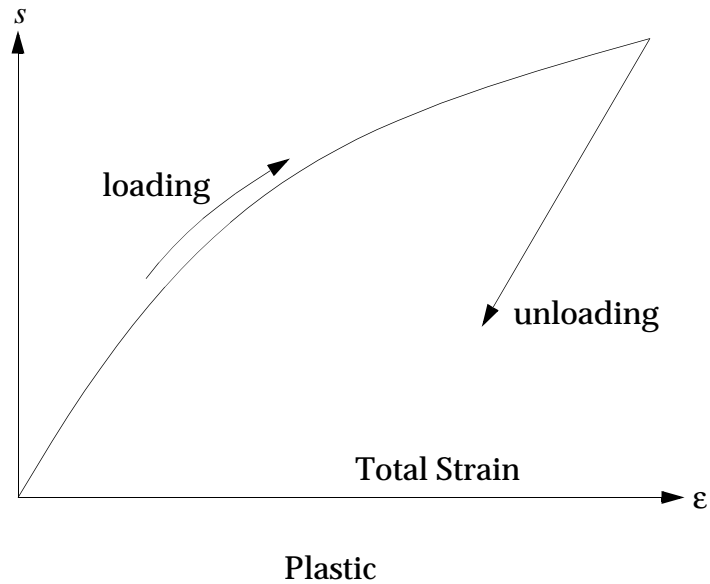


Figure 13-4 Material Nonlinear Models

Yield Function

Associated with the yield criteria, there exist yield functions in the form of

$$F(\sigma, \bar{\epsilon}^p) = f(\sigma) - Y(\bar{\epsilon}^p) = 0 \quad \text{Eq. 13-12}$$

where:

$f(\sigma)$ = an effective stress, a function of the stress state

$Y(\bar{\epsilon}^p)$ = a tensile yield stress, a function of the stress hardening.

Geometric representation of the yield function is a surface in the stress space ($\sigma_1 - \sigma_2 - \sigma_3$ coordinates) which is called a yield surface. The von Mises yield surface is a circular cylinder and Tresca's is a hexagonal cylinder. The yield surface is reduced to the yield locus in the case of plane stress, as shown in **Figure 13-5** and **Figure 13-6** for the von Mises and Tresca criteria, respectively. The Mohr-Coulomb and Drucker-Prager yield functions represent conical surfaces as shown in **Figure 13-7** and **Figure 13-8**.

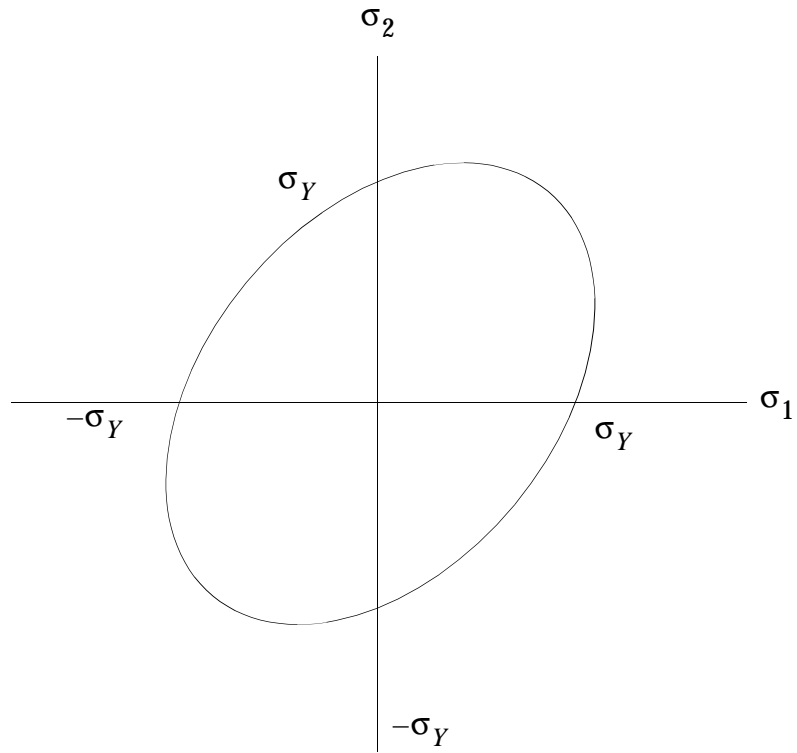


Figure 13-5 von Mises Yield Locus for Plane Stress

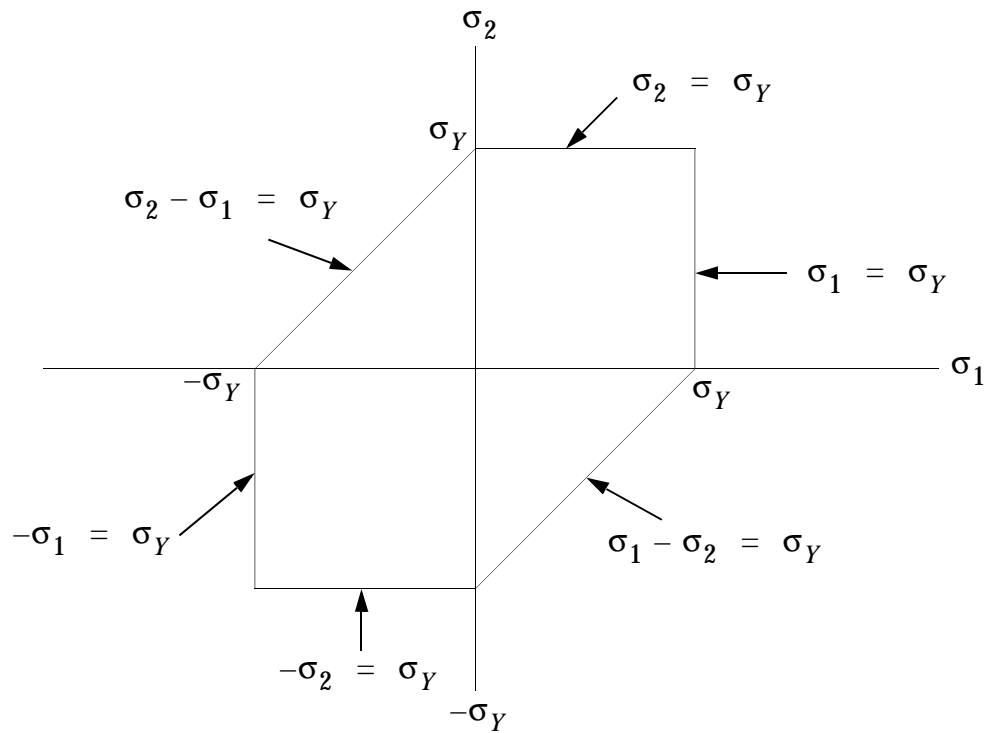


Figure 13-6 Tresca's Yield Locus for Plane Stress

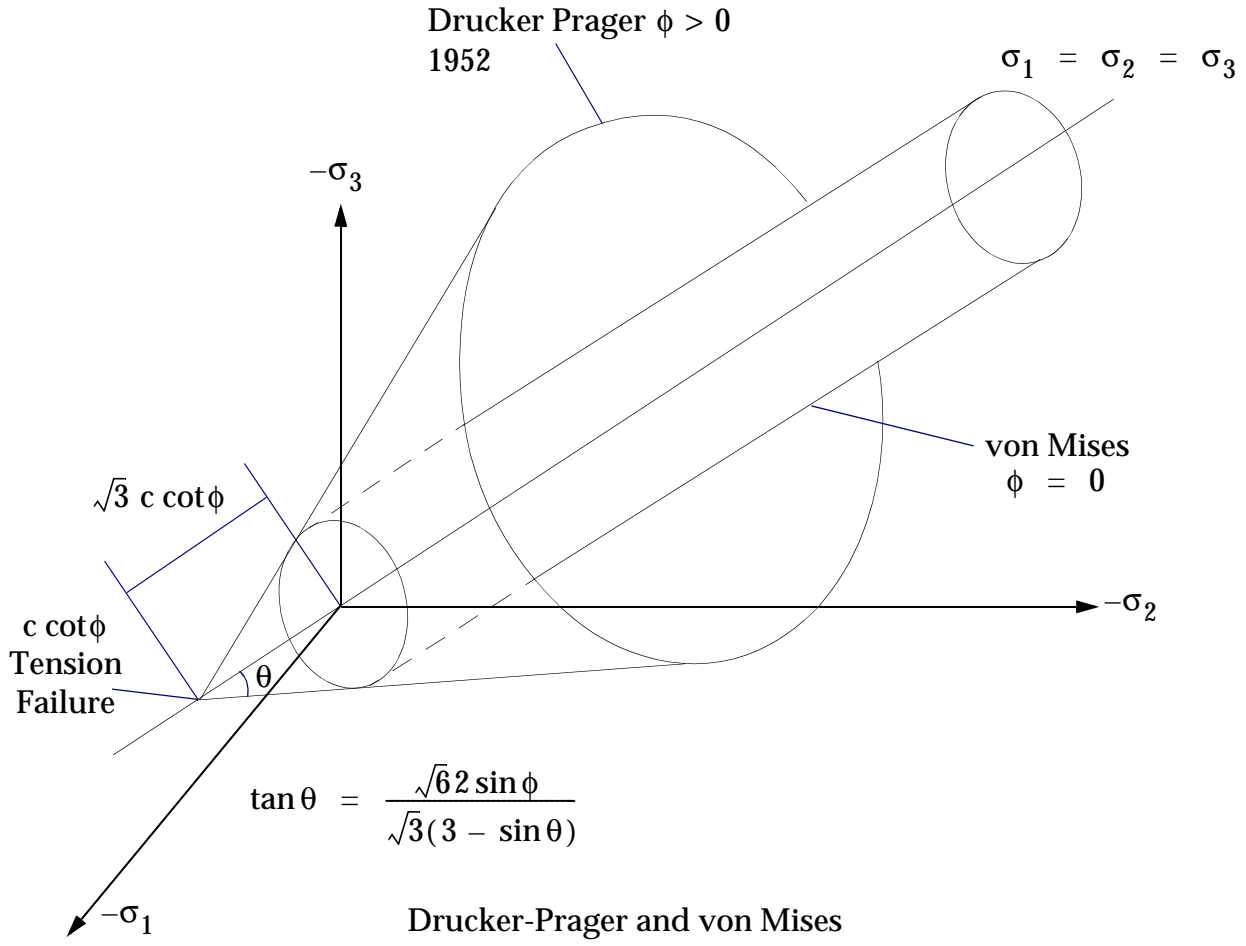
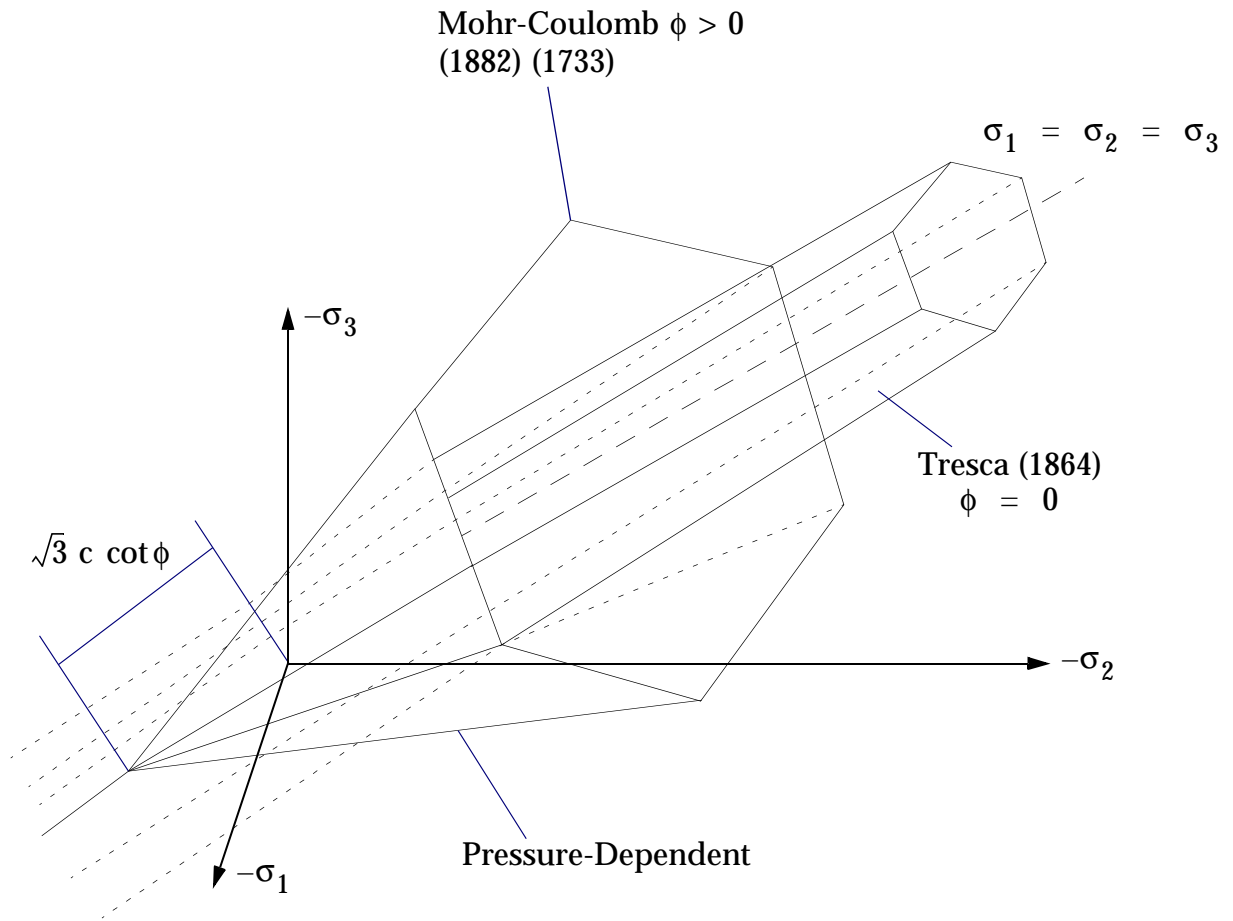


Figure 13-7 Some Isotropic Yield Surfaces in Principal Stress Space



Mohr-Coulomb and Tresca

Figure 13-8 Some Isotropic Yield Surfaces in Principal Stress Space

13.3.5 Viscoelastic Material in Nonlinear Analysis

The quasi-static behavior of the viscoelastic material may be analyzed using the creep capability of MSC.Nastran. Viscoelastic properties (linear or nonlinear) can be defined in terms of rheological model parameters (K_p , C_p , and C_s) of the Maxwell-Kelvin model in the Bulk Data entries CREEP and TABLES1. Then the quasi-static analysis of the viscoelastic materials can be performed by specifying time increment in the Bulk Data entry NLPARM (selected by the subcase) in SOL 106. For details see Section 6.4 of the *Handbook for Nonlinear Analysis*.

13.3.6 Temperature-Dependent Material

Currently only linear and nonlinear elastic materials can be temperature dependent. Temperature-dependent properties for elastic and nonlinear elastic materials are specified by using the MATTi, TABLEST, and TABLES1 Bulk Data entries. Elastic-

plastic and hyperelastic materials cannot be temperature dependent. A limited temperature dependence for creep materials is available based on the Arrhenius equation.

13.3.7 Geometric Nonlinearity

Geometric nonlinearities are manifested in problems involving large rotations and large deformation. The characteristics are follower forces due to large rotations, geometric stiffening due to initial stress effect (as a result of large rotations), and large strains due to large deformation.

Geometric nonlinear effects should be significant if the deformed shape of the structure appears distinctive from the original geometry by a visual inspection. A more rigorous and quantitative definition for the large displacements can be derived from the plate theory of Kirchhoff and Love: the small deflection theory is valid for a maximum deflection of less than 20% of the plate thickness or 2% of the small span length. However, this definition seems to be a little conservative for numerical analysis, and there is no distinct limit for large displacements because geometric nonlinear effects are related to the boundary conditions as well as the dimensions of the structure. If the load-deflection curve of the critical point can be estimated, the loading point should be in the nonlinear portion of the curve.

Geometric nonlinear effects in the structure involving large rotations, whether rigid body rotations or deformation induced rotations, are self-evident. Stiffening of a membrane, stiffness in a pendulum or snap-through of an arch belong to this category. The motion of a pendulum under gravity is caused by geometric (differential) stiffness. Follower forces are manifested when the applied loads are displacement dependent, such as pressure load and thermal load applied on the surface that rotates. Centrifugal force is another example of follower forces. Large strain effects are pronounced in metal forming, rubber and elastomer applications. In such applications the strains exceed 100%. Finite strain formulation is required to treat the problems in this category. MSC.Nastran currently supports the large strain capability for rubbers and elastomers. This formulation contains the effect of large strain as well as geometric nonlinearity. However, it does not support large strain formulation for metal forming applications. In most structural applications, however, moderately large strains (20 to 30%) appear in local areas if there is any large deformation

Other geometric nonlinear effects are treated by updated element coordinates, gimbal angles (or rotation vector), and the differential stiffness $[K^d]$.

13.3.8 Nonlinear Elements

The nonlinear elements can be divided into two categories: physical and contact elements. Physical elements have material properties associated with them. Contact elements do not have any material properties associated with them.

Physical Elements

The nonlinear physical elements are listed in [Table 13-6](#) and [Table 13-7](#).

The CBUSH element may have nonlinear force-deflection properties defined on the PBUSH and PBUSHT property entries. The CBUSH element is geometric linear; the orientation is not updated in case of large deformation. The CBUSH1D element is both geometric and material nonlinear.

The elements CROD, CONROD, and CTUBE may have material nonlinear extensional properties, with linear torsion. The user may supply plastic or nonlinear elastic material properties. Since the stress-strain curve for compression need not be the same as for tension, this element can, for example, be used to model cables which cannot carry tension.

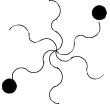
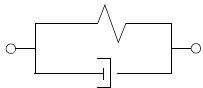
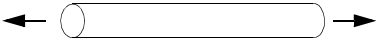
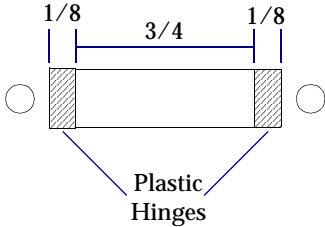
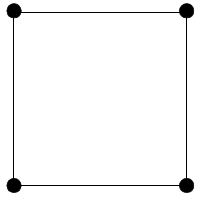
The CBEAM element has been modified to provide plastic hinges at the ends of an otherwise elastic element. This element is intended for collapse analysis in frameworks with loads at the joints, and materials with small work hardening. The user need not specify the cross-section axis about which the yielding occurs, since the implementation allows for combinations of bending moments in two directions plus an axial load. The flexibility of the plastic hinge is based upon eight idealized rods at each end, chosen to match the total area, center of gravity and moments of inertia of the crosssection. As is shown in [Table 13-8](#), the calculated ultimate moment agrees reasonably well with the theoretical value for many cross-sections. The material specified on a MATS1 Bulk Data entry should be elastic-perfectly plastic. Use of any nonlinear material other than elastic-perfectly plastic will give inaccurate results.

The CQUAD4 and CTRIA3 plate elements are available for combined bending and membrane strains found in shell structures as well as for plane strain applications. For shell type problems the state of stress is assumed to be constant in each of a variable number of equal layers. One layer should be used for plane strain. The user input is the one-dimensional stress-strain curve for tension. Higher order CQUAD8, CQUAD, and CTRIA6 elements exist for fully nonlinear (i.e., large strain and large rotation) hyperelastic plane strain analysis.

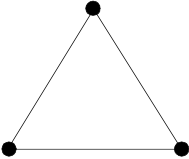
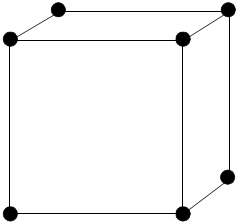
The CHEXA, CPENTA, and CTETRA elements are used to model three dimensional solid structures. No midside nodes are allowed for nonlinear analysis, except for the hyperelastic 20-noded CHEXA, 15-noded CPENTA, and 10-noded CTETRA elements.

Axisymmetric elements CQUADX and CTRIAX are available for fully nonlinear analysis with the hyperelastic material.

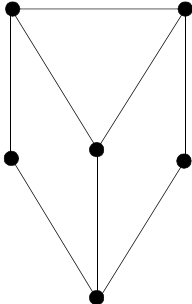
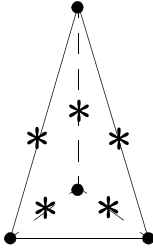
**Table 13-6 Physical Elements for Nonlinear Analysis --
Small Strain Elements**

Name	Characteristics	Nonlinear Elastic	Elasto-Plastic	Geometric Only Nonlinear
CBUSH 	Generalized 3D spring damper with 6 stiffnesses for 3 translations and 3 rotations.	Yes	No	No
CBUSH1D 	Rod-like spring and damper with generalized force function $f(u, \dot{u})$ for axial displacement u and velocity \dot{u} .	Yes	No	Yes
CROD, CONROD, CTUBE 	Extension may be plastic or nonlinear elastic. Torsion is elastic (linear).	Yes	Yes	Yes
CBEAM 	Plastic hinge at each end, which couples axial motion and rotations. Linear material for center section, traverse shear, and torsion.	No	Yes (only elastic-perfectly plastic)	Yes
CQUAD4 	Shell and Plate Two-dimensional plasticity with FIVE layers for membrane and bending. 12 layers maximum; 1 layer minimum. Transverse shear is linear.	Yes	Yes	Yes
	Plane Strain PSHELL with MID2=-1 and PARAM,NLAYER=1	Yes	Yes	Yes
	Plane Stress MID2=0 NLAYER=1	Yes	Yes	Yes

**Table 13-6 Physical Elements for Nonlinear Analysis --
Small Strain Elements (continued)**

Name	Characteristics	Nonlinear Elastic	Elasto-Plastic	Geometric Only Nonlinear
CTRIA3 	Shell and Plate Two-dimensional plasticity with FIVE layers for membrane and bending. 12 layers maximum; 1 layer minimum. Transverse shear is linear. Plane Strain PSHELL with MID2=-1 and PARAM,NLAYER=1 Plane Stress PSHELL with MID2=0 NLAYER=1	Yes Yes Yes	Yes Yes Yes	Yes Yes Yes
CHEXA 	Eight Gauss points for nonlinear analysis. Reduced shear integration. Strain (bubble) function. No mid side nodes allowed.	Yes	Yes	Yes

**Table 13-6 Physical Elements for Nonlinear Analysis --
Small Strain Elements (continued)**

Name	Characteristics	Nonlinear Elastic	Elasto-Plastic	Geometric Only Nonlinear
CPENTA 	Six Gauss points for nonlinear analysis. Reduced shear integration. Strain (bubble) function.	Yes No	Yes No	Yes No
CTETRA 	For four-noded tetrahedron a single Gauss point for nonlinear analysis. For 10-noded tetrahedron four Gauss points for nonlinear analysis.	Yes	Yes	Yes

- One-dimensional stress-strain curves use the MAT1 entry.
- All other elements may be used for nonlinear analysis as long as they remain linear.

Table 13-7 Physical Elements for Nonlinear Analysis -- Fully Nonlinear (Finite Deformation) Hyperelastic Elements

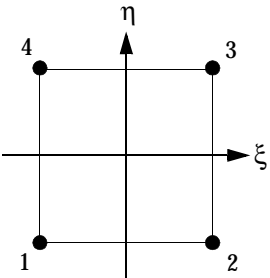
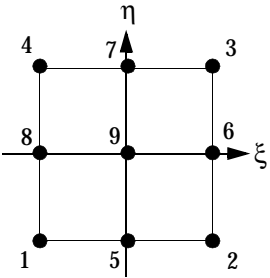
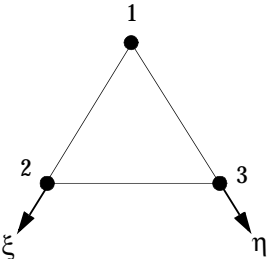
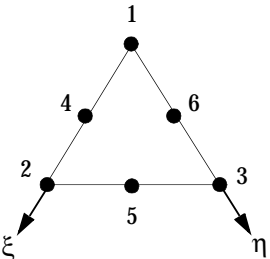
Name	Characteristics	Gauss Point	ξ	η	ζ
CQUAD, CQUAD4, CQUAD8, CQUADX 	Plane strain or axisymmetric analysis. Linear isoparametric quadrilateral with constant pressure. Compressible and nearly incompressible applications. Four Gauss points.	1 2 3 4 $\alpha = 1/\sqrt{3}$	$-\alpha$ α $-\alpha$ α	$-\alpha$ $-\alpha$ α α	--
CQUAD, CQAUD8, CQUADX 	Plane strain or axisymmetric analysis. Quadratic isoparametric quadrilateral with linear pressure. Compressible and nearly incompressible applications. Center node not available on the CQUAD8 connectivity. Nine Gauss points.	1 2 3 4 5 6 7 8 9 $\alpha = \sqrt{15}/5$	$-\alpha$ 0 α $-\alpha$ 0 0 α $-\alpha$ 0 α α	$-\alpha$ $-\alpha$ $-\alpha$ 0 0 0 0 α α	--
CTRIA3, CTRIA6, CTRIA3 	Plane strain or axisymmetric analysis. Linear isoparametric triangle with constant pressure. Compressible and nearly incompressible applications. One Gauss point.	1	$1/3$	$1/3$	--
CTRIA6, CTRIA3 	Plane strain or axisymmetric analysis. Quadratic isoparametric triangle with constant pressure. Compressible and nearly incompressible applications. Three Gauss points.	1 2 3	$1/6$ $2/3$ $1/6$	$1/6$ $1/6$ $2/3$	--

Table 13-7 Physical Elements for Nonlinear Analysis -- Fully Nonlinear (Finite Deformation) Hyperelastic Elements (continued)

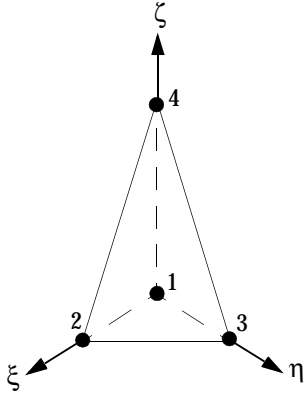
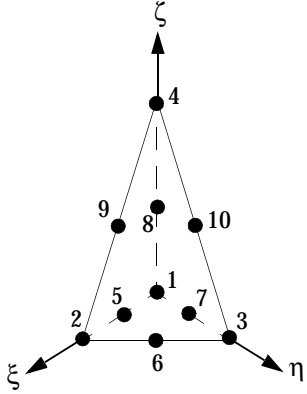
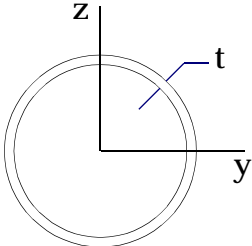
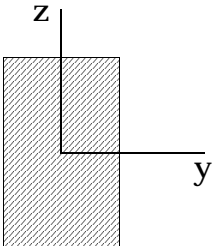
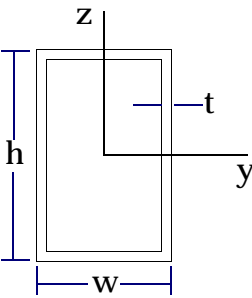
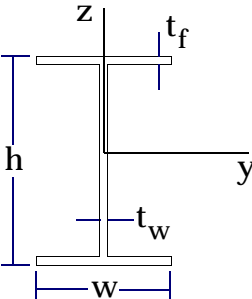
Name	Characteristics	Gauss Point	ξ	η	ζ
CTETRA 	Linear isoparametric tetrahedron with constant pressure. Compressible and nearly incompressible applications. One Gauss point.	1	1/4	1/4	1/4
CTETRA 	Quadratic isoparametric tetrahedron with linear pressure. Compressible and nearly incompressible applications. Five Gauss points.	1 2 3 4 5	1/6 1/2 1/6 1/6 1/4	1/6 1/6 1/2 1/6 1/4	1/6 1/6 1/6 1/2 1/4

Table 13-8 Accuracy in Calculation of Ultimate Moment in Yielded State

Section			
	Moment Axis	η	
	Any multiple of 45° from y-axis	.9481	
	Moment Axis	η	
	y or z	.9856	
 <p style="text-align: center;">$t \ll h, w$</p>	h/w	η for y-axis	η for z-axis
	0	.8536	.9856
	.5	.9031	.9543
	1.0	.9295	.9295
	2.0	.9543	.9031
	∞	.9856	.8536
 <p style="text-align: center;">$t_f, t_w \ll h, w$</p>	ht_w/wt_f	η for y-axis	η for z-axis
	0	.8536	.9856
	.5	.8826	1.102
	1.0	.9031	1.207
	2.0	.9295	1.394
	∞	.9856	

Contact Elements

The contact (interface) elements are listed in [Table 13-9](#).

The CGAP element is intended to model surfaces which may come into contact. When positive pressure exists, the gap can carry any transverse shear load which is less than the coefficient of friction times the normal load. The CGAP element connects two grid points which may be initially coincident. There is no geometric nonlinear behavior, which implies that the orientation of the contact plane does not change during deflection. (The physical shape of the two contact surface would have to be specified and it would require the solution of a difficult analytic problem to determine the location of the actual contact point or points.)

Due to the requirements of the solution algorithm, there must be a finite compression stiffness for the CGAP element. The value must be carefully chosen, since a very large value may lead to numerical problems. A finite extensional stiffness may be used to prevent the drift of structures that are supported only by CGAP elements. The user may also supply an initial opening of the gap. An orientation vector is required (see CGAP entry for details) to define the transverse axes, even if no shear is to be carried.

The slideline contact is capable of modeling nonlinear contact geometry and inelastic material behavior including large deformation. Slideline contact is useful for two dimensional geometries, for example plane stress, plane strain and axisymmetric. They can also be used for three dimensional geometries provided the contact between the two bodies can be defined in terms of parallel planes called the slideline planes. The sliding and separation of the two bodies is restricted to the slideline planes. The bodies can have large relative motions within the slideline plane. However, relative motions outside the slideline planes are ignored; therefore, they must be small compared to a typical slideline element.

Table 13-9 Contact Elements for Nonlinear Analysis

Name	Characteristics	Friction Law	Geometric Nonlinearity
CGAP	Contact between two points. Extension has different springs for compression and tension. Shear force is less than the friction coefficient times the compression. Transverse finite compression stiffness is required.	Coulomb's Law with: kinetic friction 0 static friction	No
Slideline	Contact between a point and a line.	Coulomb's Law with: kinetic friction = static friction	Yes

13.3.9 Small Strain Elements

Overview

A subset of MSC.Nastran's elements for linear analysis is available for nonlinear structural analysis (see [Table 13-10](#)). A more general overview for all types of elements is presented in “[Structural Elements](#)” on page 47. The nonlinear elements for small strain analysis can handle material or geometric nonlinearities or both. In geometric nonlinear analysis, the elements may undergo large total displacements and rotations but the net deformation of each element has to remain small, therefore these elements are called “small strain” elements. For linear material and infinitesimal deformation, a nonlinear analysis with small strain elements gives results which are identical to a linear analysis. The small strain elements may be combined with any other type of elements, e.g., hyperelastic elements and/or linear elements.

Table 13-10 Small Strain Elements in Nonlinear Structural Analysis

Element	Property	Material ⁴
Line Elements		
CROD	PROD	MAT1 (MATT1), MATS1
CONROD	--	MAT1 (MATT1)
CTUBE	PTUBE	MAT1 (MATT1)

Table 13-10 Small Strain Elements in Nonlinear Structural Analysis

Element	Property	Material ⁴
CBEAM	PBEAM PBCOMP ¹ (Composite Beam)	MAT1 (MATT1), MATS1 (only elastic-perfectly plastic)
2D Elements Plane Strain, Plane Stress, Shells		
CQUAD4	PSHELL	MAT1 (MATT1), MAT2 (MATT2), MAT8 ³ , MATS1, CREEP
CTRIA3	PCOMP (Composite Shell) ²	
Solid Elements		
CHEXA 8 Node only CPENTA 6 Node only CTETRA 4 & 10 Node	PSOLID	MAT1 (MATT1), MAT9 (MATT9), MATS1, CREEP

For all elements, geometric nonlinearity is turned on with PARAM, LGDISP,1.

1. PBCOMP may refer to MAT1 only (linear material).
2. PCOMP must refer to MAT1, MAT2, MAT8 (only linear material).
3. MAT8 must not be referred by a MATS1 entry (only linear orthotropic material).
4. MAT1 linear elastic isotropic material.
MAT2 linear elastic anisotropic material for plates.
MAT8 linear elastic orthotropic material for plates.
MAT9 linear elastic anisotropic material for solids.
MATTi temperature dependency for MAT1, MAT2, MAT9.
MATS1 nonlinear material (piecewise linear, elasto-plastic) referred by MAT1,
MAT2, MAT9.
CREEP creep characteristics, referred by MAT1, MAT2, MAT9.

User Interface

Element-related user input is summarized in **Table 13-10**. The nonlinear element calculations are activated if a nonlinear material exists (MATS1 or CREEP Bulk Data entries) or if a geometric nonlinear analysis of the whole model is requested with PARAM, LGDISP,1. The default is geometric linear analysis, PARAM, LGDISP, -1.

For elements with rotational degrees-of-freedom (CBEAM, CTRIA3, CQUAD4), two different definitions for large rotations are available, gimbal angles and rotation vector. The definitions are chosen with `PARAM,LANGLE,i`. The default is the gimbal angle definition, $i = 1$. The 3 gimbal angle components are the 3 subsequent rotations around the global axes. The second rotation component refers to the second global axis which has been rotated once by the first rotation. The third rotation refers to the third global axis which has been rotated twice by the first and second rotation. The rotation vector definition is chosen with $i = 2$. The components of the rotation vector refer to the fixed global axes. The direction of the rotation vector is the principal direction of the 3-D rotation. The magnitude of the rotation vector is the principal value of the 3-D rotation. The difference of the two angle definitions in the output of the elements appears only for large 3-D rotations.

Output

Element stress and strain output is requested with the `STRESS` Case Control command. The stress and strain output for all nonlinear elements is divided into a nonlinear and a linear format. The nonlinear format outputs stresses together with strains. Compared to the linear format, the nonlinear format provides more information with regard to nonlinear material laws (effective strain, equivalent stress etc.). In the linear format, strain output is not available. The stresses and strains of the small strain elements refer to the undeformed area and length, respectively. The components of the stresses and strains are in the deformed element coordinate system which corotates with the elements' rigid body deformation.

Averaged grid point stresses are requested with the `GPSTRESS` Case Control command. Averaged grid point stresses are available in nonlinear static analysis but not in nonlinear transient analysis. Element forces are requested with the `FORCE` or `ELFORCE` Case Control command. Element forces per unit length refer to the undeformed length. The components of the element forces are in the deformed element system. Element forces are available in nonlinear static analysis but not in nonlinear transient analysis.

Grid point forces are requested with the `GPFORCE` Case Control command. Grid point forces are not available in nonlinear structural analysis.

Elforces are not available for solid elements.

Deformations are requested with the `DISP` Case Control command. The output of displacements and rotations is available for all types of nonlinear structural analysis.

Corotational Formulation in Geometric Nonlinear Analysis

For small strain elements, a large deformation is split into element rigid body deformation and element net distortion. The rigid body rotations of the points in the element are approximated by the rotation of the local element triad which follows the element deformation, as shown in **Figure 13-9**. For the remaining element net distortions, a linear strain measure is used. The method is called corotational formulation. The formulation is capable to simulate large total deformations if a fine mesh is used and the stretches remain small.

In geometric nonlinear analysis with MSC.Nastran, the tangent stiffness matrix is calculated in two steps. First, the linear stiffness or nonlinear material stiffness is calculated. The linear stiffness or nonlinear material stiffness is constructed from the variation of the stresses in the virtual work. Second, the differential stiffness is calculated. The differential stiffness is constructed from the variation of the strains in the virtual work. The differential stiffness is also called geometric stiffness.

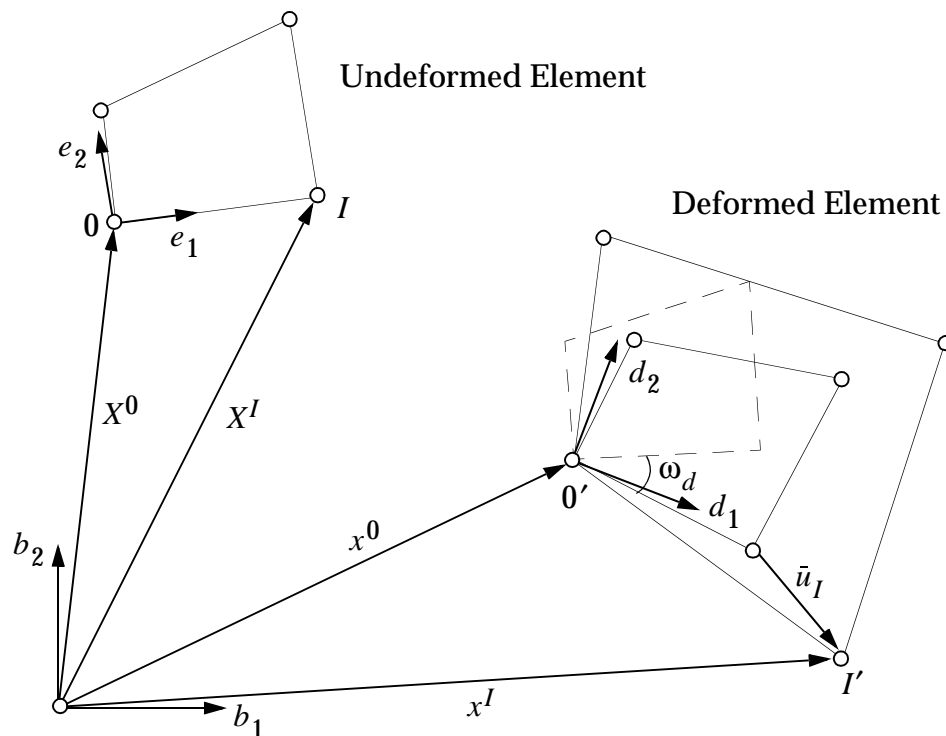


Figure 13-9 Corotational Concept for Small Strain Elements in Geometric Nonlinear Analysis

Guidelines

Large element net deformations should be avoided. In areas of the structure where large total deformations are expected, the mesh must be fine enough to keep the element net deformations small. The element net rotation should not exceed 20 degrees and the element should not be stretched by more than 10%. If stretches exceed 20%, it is recommended to use hyperelastic elements if applicable.

Limitations

- The element net distortions have to remain small.
- Rigid body elements (RBEi, RBAR, RROD entries, etc.) do not rotate in geometric nonlinear analysis.
- Multipoint constraints (MPCs) remain linear, the user defined constraint equations do not change automatically in geometric nonlinear analysis. In nonlinear static analysis, the user may change the MPCs from subcase to subcase. Then, the changes in the MPCs are accounted for incrementally.
- Offsets in the CBEAM, CTRIA3 and CQUAD4 elements are not allowed in combination with nonlinear material.
- The results in linear and nonlinear buckling with offsets may be incorrect.
- A temperature variation of the nonlinear material constants in plasticity is not available.
- For the CBEAM, the only nonlinear material available is elastic-perfectly plastic.

13.3.10 Fully Nonlinear Hyperelastic Elements

The hyperelastic elements are fully nonlinear finite deformation elements, including large strain and large rotation. The following elements are available: plane strain (CQUAD4, CQUAD8, CQUAD, CTRIA3, and CTRIA6), solid (CHEXA, CPENTA, and CTETRA) and axisymmetric elements (CQUADX and CTRIAX). See “**Nonlinear Elements**” on page 600, **Table 13-7** for more details. The element formulation is total Lagrangean in updated coordinates. See **Figure 13-10** for an illustration of the total Lagrangean concept. A nonlinear strain measure and pointwise rotation computed at each Gauss point are employed. Equilibrium is satisfied in the deformed configuration.

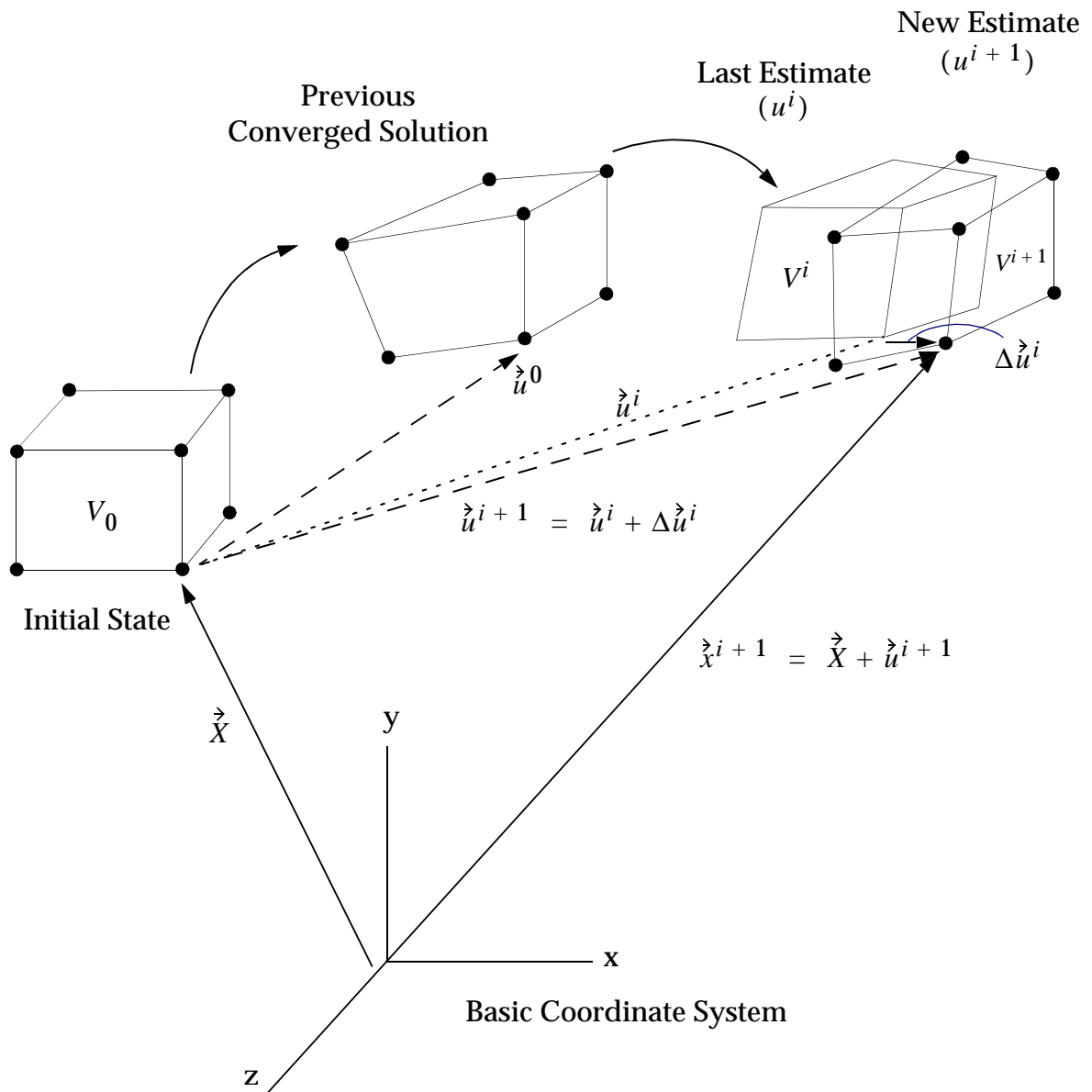


Figure 13-10 Total Lagrangian Concept: Deformations are Measured from an Initial Undeformed State

The elements are especially designed to address volumetric locking and other related problems such as ill-conditioning of the stiffness matrix, incorrect stresses etc., due to near incompressibility. To this end, a mixed formulation based on a three field variational principle in which the pressure and the volume ratio are interpolated independently from the displacements is employed. At the incompressible limit the bulk modulus becomes infinite and the volumetric strain $J - 1$ becomes zero so that the pressure may no longer be obtained from the displacements through the constitutive equation. The pressure and volume ratio interpolations are discontinuous between the elements.

The hyperelastic elements are supported in the nonlinear solutions, SOLs 106 and 129. They are not supported in the nonlinear solutions SOLs 66 and 99. They can be run through SOLs 101 and 105 for model checkout in linear analysis, the results, however, will generally not be correct. For buckling analysis, PARAM,BUCKLE in SOL 106 should be used instead.

User Interface

Because of no difference in geometry, the hyperelastic elements are defined using the current CQUAD4, CQUAD8, CTRIA3, CTRIA6 CHEXA, CPENTA, and CTETRA connectivity entries. Additional CQUAD, CQUADX, and CTRIAX connectivity entries are available for the hyperelastic elements. The PID field of the connectivity entries for the plane or axisymmetric and for the solid hyperelastic elements references the PLPLANE and PLSOLID property entries, respectively. The MID field references a MATHP material entry that defines the hyperelastic material. This is the only material model available for fully nonlinear, finite deformation analysis.

Output Description

The nonlinear stress output for the hyperelastic elements differs from other material nonlinear stress output. The hyperelastic stress output contains the following quantities, at each Gauss point, in the basic coordinate system for solid and axisymmetric elements and in the CID coordinate system, defined on a PLPLANE entry, for plane strain elements:

Cauchy stresses, i.e., components of the stress vector $t = dP/dA$ where dA is an infinitesimal area in the deformed configuration, and P is the force. See “[Hyperelastic Material](#)” on page 585 for a description of the constitutive relation.

- Pressure

$$p = \frac{1}{3}(\sigma_x + \sigma_y + \sigma_z)$$

- Logarithmic strains, i.e., the components of

$$\sum_{l=1}^3 \ln \lambda_l N_l N_l^T$$

where λ_l are principal stretches l/l_0 in the principal directions N_l . See [Figure 13-11](#). For compatibility with other MSC.Nastran elements, twice the shear components of the logarithmic strain tensor are printed in the output.

N_i = principal directions in undeformed configuration; $n_i = RN_i$ = principal directions in deformed configuration, obtained from N_i by a rigid body rotation. The deformation along N_i is a pure stretch, after the rigid body motion has been factored out.

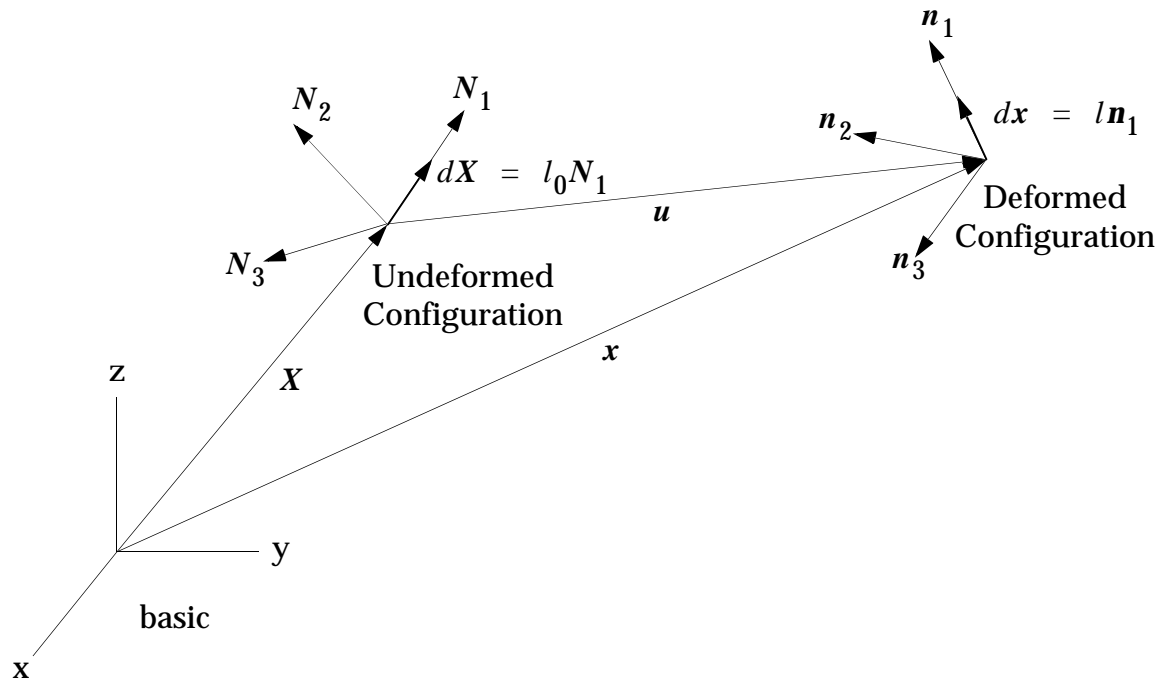


Figure 13-11

- Volumetric strain

$$J - 1 = \frac{dV - dV_0}{dV_0}$$

where J is an approximation to the determinant of the deformation gradient dx/dX .

This output is obtained with the STRESS (or ELSTRESS) Case Control command. Examples of the new output for the plane, axisymmetric and solid elements are shown in [Listing 13-5](#).

The linear output contains principal Cauchy stresses and directions with respect to the basic (or CID) coordinate system. See [Listing 13-6](#).

Listing 13-5

```

NONLINEAR STRESSES IN HYPERELASTIC QUADRILATERAL ELEMENTS (QUAFD)
ELEMENT GRID/ POINT
ID GAUSS ID CAUCHY STRESSES/ LOG STRAINS PRESSURE VOL. STRAIN
1 GAUS 1 1.678579E+02 -1.188351E-03 5.456201E+00 1.560323E-16 5.777097E+01 1.925699E-02
1.791759E+00 -1.772686E+00 .0 .0
2 1.678579E+02 -1.188351E-03 5.456201E+00 -1.191492E-15 5.777097E+01 1.925699E-02
1.791759E+00 -1.772686E+00 .0 .0
3 1.678579E+02 -1.188351E-03 5.456201E+00 -3.368891E-16 5.777097E+01 1.925699E-02
1.791759E+00 -1.772686E+00 .0 .0
4 1.678579E+02 -1.188351E-03 5.456201E+00 -7.083336E-16 5.777097E+01 1.925699E-02
1.791759E+00 -1.772686E+00 .0 .0
NONLINEAR STRESSES IN HYPERELASTIC AXISYMM. QUADRILATERAL ELEMENTS (QUAFXD)
ELEMENT GRID/ POINT
ID GAUSS ID RADIAL(X) AXIAL(Y) THETA(Z) XY PRESSURE VOL. STRAIN
1 GAUS 1 1.808360E-09 2.917973E+02 1.808360E-09 4.298727E-15 9.726578E+01 3.242192E-02
-9.420009E-01 1.930910E+00 -9.570014E-01 1.188717E-01
2 1.808132E-09 2.917973E+02 1.808132E-09 -2.820565E-17 9.726578E+01 3.242192E-02
-9.420009E-01 1.930910E+00 -9.570014E-01 1.188717E-01
3 1.808019E-09 2.917973E+02 1.808019E-09 -2.820565E-17 9.726578E+01 3.242192E-02
-9.420009E-01 1.930910E+00 -9.570014E-01 1.188717E-01
4 1.808218E-09 2.917973E+02 1.808218E-09 1.004509E-16 9.726578E+01 3.242192E-02
-9.420009E-01 1.930910E+00 -9.570014E-01 1.188717E-01
NONLINEAR STRESSES IN HYPERELASTIC HEXAHEDRON ELEMENTS (HEXAFD)
ELEMENT GRID/ POINT
ID GAUSS ID X Y Z XY YZ ZX PRESSURE VOL. STRAIN
1 GAUS 1 2.9180E+02 1.8047E-09 1.8047E-09 -7.8413E-15 -7.7424E-17 2.3330E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
2 2.9180E+02 1.8047E-09 1.8047E-09 -3.1028E-16 -1.4850E-16 9.6370E-16 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
3 2.9180E+02 1.8047E-09 1.8047E-09 2.8235E-15 6.2553E-17 -6.4668E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
4 2.9180E+02 1.8047E-09 1.8047E-09 -5.9973E-16 9.1812E-17 -2.1878E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
5 2.9180E+02 1.8047E-09 1.8047E-09 2.3338E-15 -1.2646E-16 2.8235E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
6 2.9180E+02 1.8047E-09 1.8047E-09 -2.3359E-16 -1.1296E-16 2.1388E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
7 2.9180E+02 1.8047E-09 1.8047E-09 2.1626E-15 2.1405E-17 -6.2480E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
8 2.9180E+02 1.8047E-09 1.8047E-09 -4.0477E-16 2.4964E-17 -1.9690E-15 9.7266E+01 3.2422E-02
1.9459E+00 -9.5700E-01 -9.5700E-01 .0 .0 .0
SIMPLE TENSION DECEMBER 1, 1993 MSC/NASTRAN 11/30/93 PAGE 108

```

Listing 13-6

```

                                NONLINEAR
LOAD STEP = 1.00000E+00
STRESSES IN HYPERELASTIC QUADRILATERAL ELEMENTS (QUADFD)
ELEMENT GRID/ POINT -----CAUCHY STRESSES----- PRINCIPAL STRESSES (ZERO SHEAR)
ID GAUSS ID NORMAL-X NORMAL-Y SHEAR-XY ANGLE MAJOR MINOR
1 1 1 1.678579E+02 -1.188351E-03 1.560323E-16 .0000 1.678579E+02 -1.188351E-03
  2 1.678579E+02 -1.188351E-03 -1.191492E-15 .0000 1.678579E+02 -1.188351E-03
  3 1.678579E+02 -1.188351E-03 -3.368891E-16 .0000 1.678579E+02 -1.188351E-03
  4 1.678579E+02 -1.188351E-03 -7.083336E-16 .0000 1.678579E+02 -1.188351E-03
STRESSES IN HYPERELASTIC AXISYMM. QUADRILATERAL ELEMENTS (QUADFXD)
ELEMENT GRID/ POINT -----CAUCHY STRESSES----- PRINCIPAL STRESSES (ZERO SHEAR)
ID GAUSS ID RADIAL(X) AXIAL(Y) SHEAR-XY ANGLE MAJOR MINOR
1 1 1 1.808360E-09 2.917973E+02 4.298727E-15 90.0000 2.917973E+02 1.808360E-09
  2 1.808132E-09 2.917973E+02 -2.820565E-17 -90.0000 2.917973E+02 1.808132E-09
  3 1.808019E-09 2.917973E+02 -2.820565E-17 -90.0000 2.917973E+02 1.808019E-09
  4 1.808218E-09 2.917973E+02 1.004509E-16 90.0000 2.917973E+02 1.808189E-09
                                NONLINEAR SUBCASE 100 $ UNIAXI
LOAD STEP = 1.00000E+00
STRESSES IN HYPERELASTIC HEXAHEDRON ELEMENTS (HEXA F D)
ELEMENT-ID GRID/ POINT -----CAUCHY STRESSES----- DIR. COSINES MEAN
1 GAUSS ID NORMAL SHEAR PRINCIPAL -A- -B- -C- PRESSURE
1 X 2.917973E+02 XY -7.841313E-15 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804707E-09 YZ -7.742389E-17 B 1.804693E-09 LY .00 .0 .0
  Z 1.804707E-09 ZX 2.332978E-15 C 1.804693E-09 LZ .00 .0 .0
2 X 2.917973E+02 XY -3.102808E-16 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804707E-09 YZ -1.485038E-16 B 1.804693E-09 LY .00 .0 .0
  Z 1.804707E-09 ZX 9.636988E-16 C 1.804693E-09 LZ .00 .0 .0
3 X 2.917973E+02 XY 2.823466E-15 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804707E-09 YZ 6.255328E-17 B 1.804707E-09 LY .00 .0 .0
  Z 1.804707E-09 ZX -6.466815E-15 C 1.804707E-09 LZ .00 .0 .0
4 X 2.917973E+02 XY -5.997311E-16 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804707E-09 YZ 9.181191E-17 B 1.804707E-09 LY .00 .0 .0
  Z 1.804707E-09 ZX -2.187819E-15 C 1.804707E-09 LZ .00 .0 .0
5 X 2.917973E+02 XY 2.333811E-15 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804707E-09 YZ -1.264558E-16 B 1.804707E-09 LY .00 .0 .0
  Z 1.804707E-09 ZX 2.823466E-15 C 1.804707E-09 LZ .00 .0 .0
6 X 2.917973E+02 XY -2.335867E-16 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804693E-09 YZ -1.129638E-16 B 1.804694E-09 LY .00 .0 .0
  Z 1.804693E-09 ZX 2.138826E-15 C 1.804694E-09 LZ .00 .0 .0
7 X 2.917973E+02 XY 2.162630E-15 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804665E-09 YZ 2.140466E-17 B 1.804636E-09 LY .00 .0 .0
  Z 1.804665E-09 ZX -6.248027E-15 C 1.804636E-09 LZ .00 .0 .0
8 X 2.917973E+02 XY -4.047673E-16 A 2.917973E+02 LX 1.00 .0 .0 9.726578E+01
  Y 1.804651E-09 YZ 2.496398E-17 B 1.804622E-09 LY .00 .0 .0
  Z 1.804651E-09 ZX -1.969031E-15 C 1.804622E-09 LZ .00 .0 .0
LOAD STEP = 1.00000E+00

```

Guidelines

MSC.Nastran performs a fully nonlinear analysis, including the effect of large strain and large rotation, if a PLPLANE and/or PLSOLID Bulk Data entry are encountered. The hyperelastic material is defined by using the MATHP Bulk Data entry. A nonlinear analysis is always performed except in linear solutions when a hyperelastic material is defined.

It is best that the differential stiffness PARAM,LGDISP,1 be used in all hyperelastic runs, but it is not the default. If the strains and the rotations are small, then the hyperelastic analysis should compare to the linear elastic analysis. However, for hyperelastic analysis, it is always best to use PARAM,LGDISP,1.

Due to the independent pressure interpolations, particularly at very high compressive values of the pressure (of the order of the bulk modulus), indefiniteness of the stiffness matrix may occur, caused by an apparent material instability. By default in such instances MSC.Nastran will ignore the so-called differential stiffness. Although

indefiniteness of the stiffness matrix is probably not associated with a geometric instability, this may indeed help, as many of the terms, which heavily depend on the pressure, are part of the differential stiffness.

Limitations

The fully nonlinear hyperelastic CQUAD, CQUAD4, CQUAD8, CTRIA3, and CTRIA6 are plane strain elements only; they do not contain bending properties, nor can they be used in plane stress analysis. Therefore, they cannot be subject to out-of-plane loading conditions and do not sustain a thickness. The elements assume a thickness of 1.0 for mass calculation in dynamic analyses.

The lower order fully nonlinear hyperelastic elements CQUAD4, CTRIA3, 8-noded CHEXA, 6-noded CPENTA and 4-noded CTETRA are not immune to shear locking and therefore are not suitable for bending type loading. The higher order elements CQUAD8, 9-noded CQUAD, CTRIA6, 20-noded CHEXA, 15-noded CPENTA, and 10-noded CTETRA should be used in such situations. See “[Example 1: Rubber Bushing Problem](#)” on page 621.

Even though the fully nonlinear elements are defined using the current CQUAD4, CQUAD8, CTRIA3, CTRIA6, CHEXA, CPENTA, and CTETRA Bulk Data entries, internally, MSC.Nastran designates them as the separate elements, CQUAD4FD, CHEXAFD (for Finite Deformation) etc., because of the different element formulations. This designation is printed in all the output in which the elements are referenced by name, such as PARAM,EST, PARAM,GPECT, the sequence processor, and the element stress output.

GPSTRESS and FORCE output is not available for the fully nonlinear hyperelastic elements.

The element types for the fully nonlinear hyperelastic elements are as follows:

Element	Element Number	Element Number in OESNL Data Block
CQUAD4	139	201
CQUAD8 (Reverts to CQUAD4 when all midside grids are omitted)	164	208
CQUAD (Reverts to CQUAD4 when center and all midside grids are omitted)	164	208
CTRIA3	162	206

Element	Element Number	Element Number in OESNL Data Block
CTRIA6 (Reverts to CTRIA3 when all midside grids are omitted)	167	211
CHEXA with 8 nodes	140	202
CHEXA (Reverts to 8-noded CHEXA when all midside grids are omitted)	163	207
CPENTA with 6 grids	160	204
CPENTA (Reverts to 6-noded CPENTA when all midside grids are omitted)	165	209
CTETRA	161	205
CTETRA (Reverts to 4-noded CTETRA when all midside grids are omitted)	166	210
CQUADX	170	214
CQUADX (Reverts to 4-noded CQUADX when all midside grids are omitted)	171	215
CTRIAX	168	212
CTRIAX (Reverts to 3-noded CTRIAX when all midside grids are omitted)	169	213

Example 1: Rubber Bushing Problem

The cross section of a rubber bushing is shown in [Figure 13-12](#). It is assumed that the frame and internal shaft are rigid and the rubber bushing is perfectly bonded to these components. The rubber material is assumed to be of the Mooney-Rivlin type with $A_{10} = 0.177 \text{ N/mm}^2$, $A_{01} = 0.045 \text{ N/mm}^2$, and $D_1 = 333 \text{ N/mm}^2$. Considering symmetric conditions, only one-half of the rubber bushing is modeled with 72 finite deformation 4-noded CQUAD elements. The same problem is also modeled with 72 hyperelastic 9-noded CQUAD elements. The grid points on the outer boundary are fully constrained to simulate the rubber/frame interface. For the grid points on the inner boundary, only the horizontal degrees-of-freedom were constrained, while the vertical degrees-of-freedom were tied together with MPCs. Force is applied to the top grid point on the inner wall in the vertical direction.

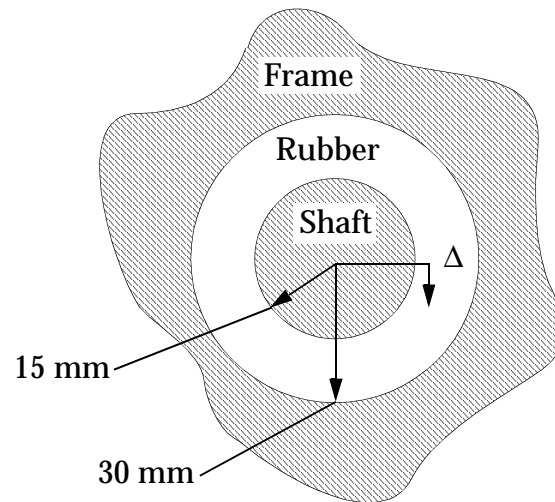


Figure 13-12 Rubber Bushing

The goal of the analysis is to determine the force-displacement curve of the unit and compare it with the solution in the reference. (Reference: Sussman, T., and Bathe, K. J., "A Finite Element Formulation for Nonlinear Incompressible Elastic and Inelastic Analysis," *Computers and Structures* (26), pages 357-409, 1987.)

Annotated Bulk Data input of interest is shown in [Listing 13-7](#).

Listing 13-7

```

ID MSC,BUSHING $
SOL 106
TIME 180
CEND
TITLE = RUBBER BUSHING
SUBTITLE = HYPERELASTIC MODEL TEST
ECHO=UNSORT
DISP(PLOT)=ALL
STRESS(PLOT)=ALL
SUBCASE 100
  MPC = 100
  NLPARM = 100
  LOAD = 100 $
  .
  .
SUBCASE 800
  MPC = 100
  NLPARM = 800
  LOAD = 800 $
BEGIN BULK
$ PARAMETERS
PARAM,DBDROPT,0
PARAM,POST,0
PARAM,LGDISP,1
$
NLPARM 100 5 ITER 1 NO +NL101
+NL101 -3 +NL102
.
.
NLPARM 800 5 ITER 1 NO +NL801
+NL801 -3 +NL802
$ DEFINE GEOMETRY
CORD2C 1 0. 0. 0. 0. 0. 1. +COR1
+CORD 1. 0. 0.
GRDSET 3456
GRID 101 1 15.0 -90. 0. 13456
GRID 102 1 17.5 -90. 0. 13456
GRID 103 1 20.0 -90. 0. 13456
.
.
$ DEFINE ELEMENTS
CQUAD4 101 1 101 102 202 201
CQUAD4 102 1 102 103 203 202
CQUAD4 103 1 103 104 204 203
.
.
$ PROPERTY ENTRY FOR 4-NODE PLANE STRAIN ELEMENT WITH MIXED ELEMENT FORMULATION
PLPLANE 1 1
$ HYPERELASTIC MATERIAL - MOONEY RIVLIN TYPE
MATHP 1 0.177 0.045 333.3
$ DEFINE LOADING CONDITIONS
FORCE 100 1301 -50. 0. 1. 0.
.
.
FORCE 800 1301 -400. 0. 1. 0.
$ DEFINE CONSTRAINTS
MPC 100 101 2 1. 1301 2 -1.
.
.
ENDDATA

```

The resulting force-displacement curve at grid point 1301 is shown in [Figure 13-13](#) and the original and deformed models at a force of 800N are shown in [Figure 13-14](#). The solution obtained with 4-noded CQUAD elements is overly stiff for a force greater than 200N (shear locking). The shear locking problem is addressed with the higher order 9-noded CQUAD elements.

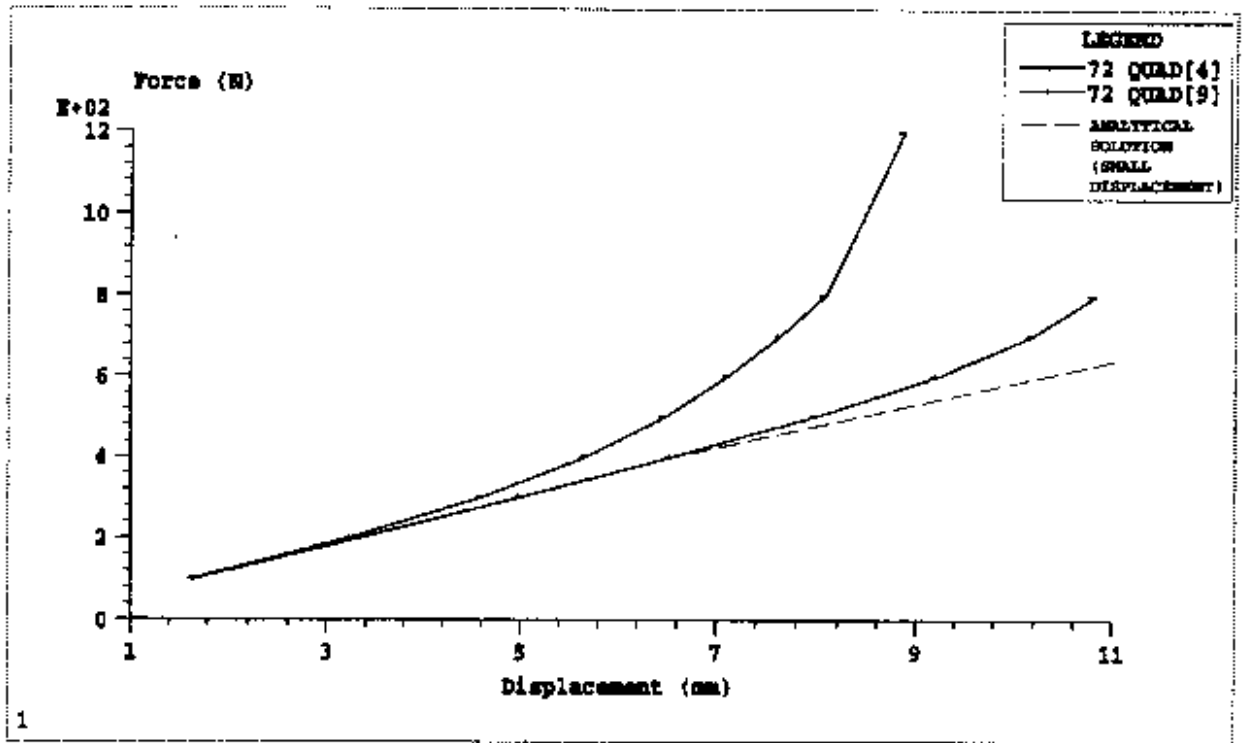


Figure 13-13 Force-Displacement Curve of a Rubber Bushing

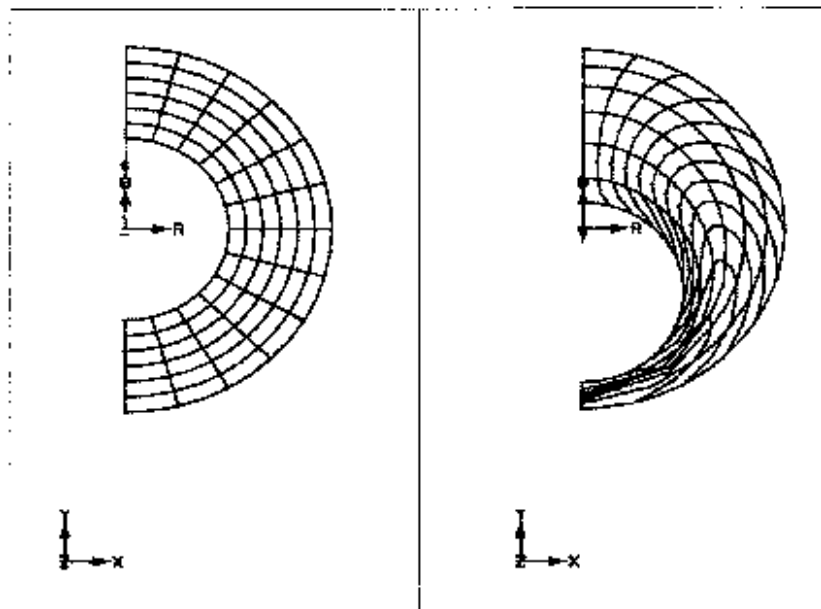


Figure 13-14 Initial and Deformed Shapes of a Rubber Bushing

The model file for this example is `bushing.dat`, which can be found in the examples directory (see the *MSC.Nastran 2004 Installation and Operations Guide*).

Example 2: Pressure of a Circular Plate

A circular plate is 15 in. in diameter and 0.5 in. thick. It is simply supported along the edge and is subjected to a uniform follower pressure. The plate is modeled with solid (CHEXA and CPENTA) and with axisymmetric elements. The solid model represents a 10° segment. Material constants are $A_{10} = 80$ psi, $A_{01} = 20$ psi, $D_1 = 50,000$ psi.

Annotated Bulk Data input of interest for the wedge model is shown in [Listing 13-8](#) and for the axisymmetric model is shown in [Listing 13-9](#).

Listing 13-8

```

BEGIN BULK
$ DEFINE PARAMETERS
PARAM,LGDISP,1
PARAM,DBDRNL,-1
PARAM,POST,0
NLPARM 10      10          ITER  1      100          +NL10
+NL10
-3
.
.
.
$ DEFINE GEOMETRY
CORD2C 1      0.      0.      0.      0.      0.      1.      +COR1
+COR1 1.      0.      0.
GRDSET
2456
GRID 101      1      0.      -5.      -0.25  1      12456
GRID 102      1      0.75     -5.      -0.25  1
GRID 103      1      1.50     -5.      -0.25  1
.
.
.
$ DEFINE ELEMENTS
CHEXA 102      1      102      103      1103      1102      202      203      +HX102
+HX102 1203      1202
CHEXA 103      1      103      104      1104      1103      203      204      +HX103
+HX103 1204      1203
CHEXA 104      1      104      105      1105      1104      204      205      +HX104
+HX104 1205      1204
.
.
.
CPENTA 101      1      101      102      1102      201      202      1202
CPENTA 201      1      201      202      1202      301      302      1302
CPENTA 301      1      301      302      1302      401      402      1402
CPENTA 401      1      401      402      1402      501      502      1502
.
.
.
$ DEFINE ELEMENT AND MATERIAL PROPERTIES
PLSOLID 1      1
MATHP 1      80.      20.      5.+4
$ DEFINE LOADING CONDITIONS
LOAD 3      1.      3.      1
LOAD 15      1.      15.      1
.
.
.
LOAD 45      1.      45.      1
PLOAD4 1      101      1.      101
PLOAD4 1      102      1.      102      1103
.
.
.
$ DEFINE CONSTRAINTS
MPC 10      1102      1      1.      102      1      -1.
MPC 10      1102      3      1.      102      3      -1.
.
.
.
ENDDATA

```

Listing 13-9

```

$ INFLATION OF A CIRCULAR PLATE
$ AXISYMMETRIC QUAD4 ELEMENTS
BEGIN BULK
$ DEFINE PARAMETERS
PARAM,LGDISP,1
PARAM,POST,0
NLPARM 10      10          AUTO1          +NL10
+NL10          -3
.
.
.
$ DEFINE GEOMETRY
GRDSET          3456
GRID  101      0.      -0.25  0.      13456
GRID  102      0.75   -0.25  0.
GRID  103      1.50   -0.25  0.
.
.
.
$ DEFINE ELEMENTS
CQUADX 101  1      101  102  202  201
CQUADX 102  1      102  103  203  202
CQUADX 103  1      103  104  204  203
.
.
.
$ DEFINE ELEMENT AND MATERIAL PROPERTIES
PLPLANE 1      1
MATHP  1      80.    20.    5.+4
$ DEFINE LOADING CONDITIONS
PLOADX1 3      101    3.          101  102
PLOADX1 3      102    3.          102  103
.
.
.
PLOADX1 45     110    45.          110  111
ENDDATA

```

Original and deformed models are shown in [Figure 13-15](#) and [Figure 13-16](#).

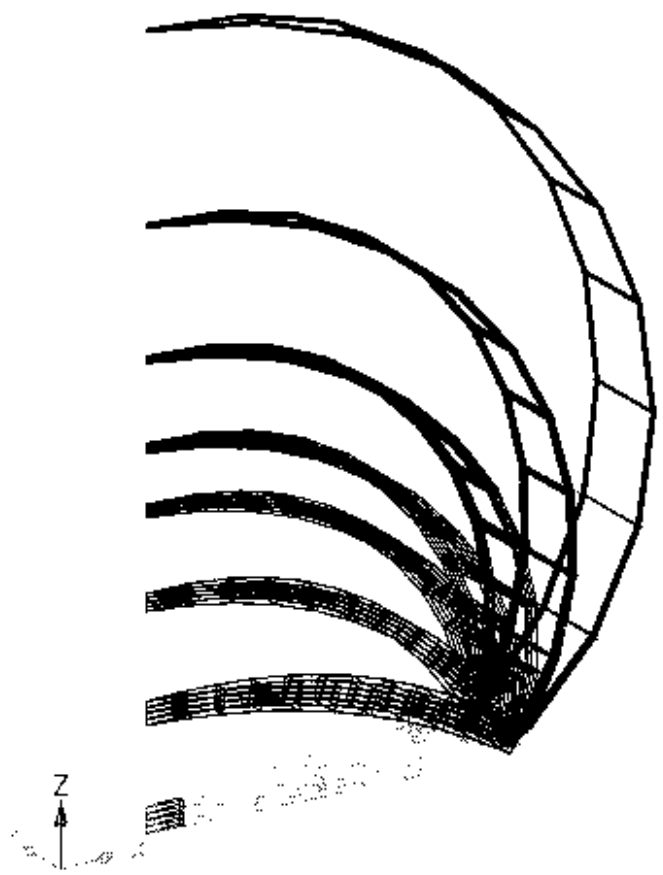


Figure 13-15 Deformed Shape of Circular Plate Under Pressure -- CHEXA and CPENTA Hyperelastic Elements

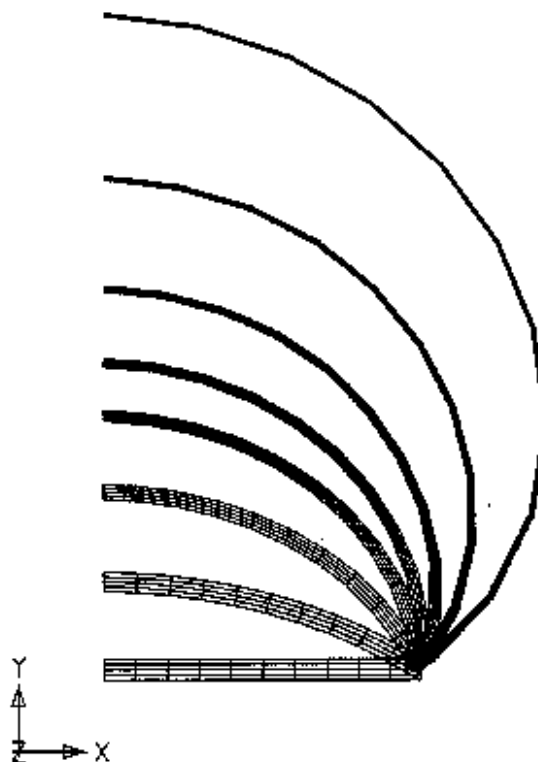


Figure 13-16 Deformed Shape of Circular Plate Under Pressure -- Axisymmetric Hyperelastic Elements

The model files for this example are prplth.dat and prpltg.dat, which can be found in the examples directory (see the [MSC.Nastran 2004 Installation and Operations Guide](#)).

13.3.11 Bushing Elements

The CBUSH element is a generalized spring and damper. Stiffness and damping can be specified for 6 degrees-of-freedom, 3 translations and 3 rotations. In SOL 106 and 129, the user can specify 6 nonlinear force-deflection curves instead of combined stiffness values. The properties of the CBUSH element are defined on the PBUSH and PBUSHi entries. The PBUSHT entry refers to TABLEDi entries for force-deflection curves. The CBUSH element is geometric linear only. The circulation is not updated.

The CBUSH1D element is a rod-type spring and damper, it is a 1D version of the CBUSH element. The CBUSH1D element can be geometric and material nonlinear. General nonlinear functions of the axial forces versus axial displacement and for velocity are defined with the PBUSH1D entry. The CBUSH1D element is the only element which can model rotation damping. The effect is demonstrated in the following example.

The CBUSH and CBUSH1D elements are recommended over the CELASi, CVISC, and CDAMP elements. The bushing elements always insure rigid body invariance and they offer nonlinear capabilities.

Swinging Pendulum

The solution to a swinging pendulum is calculated using SOL 129 of MSC.Nastran Version 70.5+. The pendulum has a length of 1.0 [m], and has a concentrated mass of $M = 1,000.0$ [kg] at the free end, see [Figure 13-17](#). We start the analysis with the pendulum at rest in horizontal position. The free end is loaded with a gravity induced load of $G = 10,000.0$ [N]. The leg of the pendulum is very stiff, $K = 1.e+7$ [N/m], so that the relative axial deformation is small compared to the overall motion of the pendulum. Large deformation effects are turned on with `PARAM,LGDISP,1`.

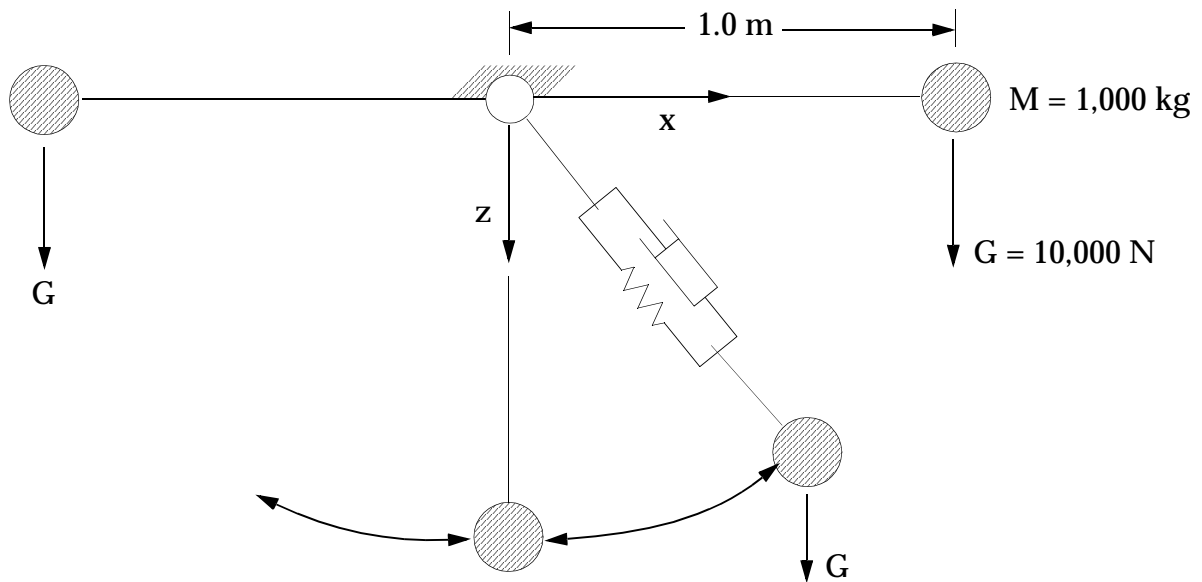


Figure 13-17 Swinging Pendulum

Listing 13-10 MSC.Nastran Input File for the Swinging Pendulum

```

$
$ Name: ar29src1
$
$ A swinging pendulum is modeled. The pendulum is loaded with
$ gravity load. The analysis goes through one and a half cycles,
$ starting in horizontal position. Large deformation is turned on.
$
$ The pendulum is modeled with two separate systems.
$ 1. CROD and CVISC, CROD rotates, CVISC stays in horizontal
$ orientation.
$ 2. CBUSH1D with SPRING and DAMPER, spring and damper rotate.
$
TIME 10 $
SOL 129 $
CEND
$
TITLE= ar29src1 swinging elastic pendulum
SUBTITLE= demonstrate rotating damping
.
.
.
$
BEGIN BULK
$
param, lgdisp, 1
$
$ GEOMETRY
$
.
.
$ CBUSH1D spring and rotating viscous damper
$
$CBUSH1D,EID,GA,GB,CID
cbush1d, 108, 208, 41, 42
$PBUSH1D,PID,K,C
pbush1d, 208, 1.e+7, 1000.
$
$          $          $          $          $          $          $          $
$ SOLUTION STRATEGY
$
.
.
.
$
ENDDATA

```

We investigate two different models of the pendulum in one run. In the first model, we use a Rod element for the stiffness of the leg, $K = 1.e+7$ [N/m], and a Visc element for 5% equivalent viscous damping, $C = 1,000.0$ [Ns/m]. The x- and z-displacements of the free end are shown in [Figure 13-18](#). The Rod rotates while the viscous damper Visc stays fixed in its initial horizontal position. The Visc element is linear and does not follow large deformations. The displacements are damped out because the viscous damper is acting in the x-direction during the whole motion.

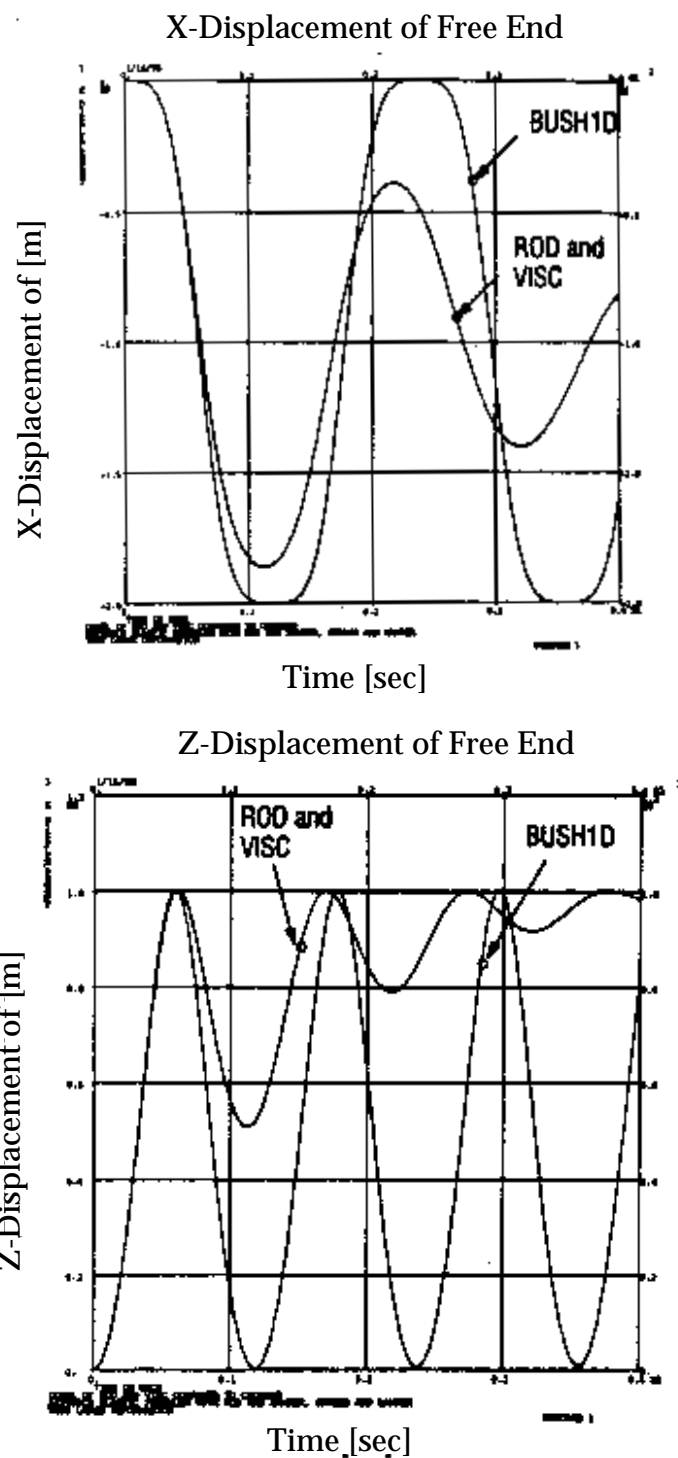


Figure 13-18 Displacement of Swinging Pendulum, Model with Rod/Visc Element Versus Model with BUSH1D Element

In the second model, we use a BUSH1D element. The element has a linear stiffness (K) and a viscous damper (C) with the same values as in the first model. In the BUSH1D element, the spring and damper rotate. The relative axial displacements and velocities

are small because of the high axial stiffness of the element. The rotating damper has no noticeable effect on the overall motion of the pendulum because it damps only the small relative deformations. The x- and z-displacements of the free end are not damped, see [Figure 13-18](#).

The example demonstrates how the answers can change if dampers rotate with the deformation compared to dampers which stay fixed in space.

13.3.12 Adaptive Gap Element

The gap element is used in MSC.Nastran to model surface contact problems. The adaptivity is based on the subincremental scheme with the capability to update the stiffness at the proper timing, bisect the increment when it is excessive, and adjust the penalty stiffness to the proper values. Generally, in a non-adaptive gap element, the axial stiffness has to be properly chosen in order to reach convergence. However, the main goal of the self-adaptive method is to relieve users from this trial and error method and to achieve convergence on the first try. Due to the adaptive nature of the algorithm, the adaptive gap method costs more in CPU time when compared to the non-adaptive gap element. However, the adaptive gap method allows a user to choose a wider latitude of axial stiffness and yet still achieve convergence. Therefore, with more experience, the user can decrease CPU time by fine-tuning this axial stiffness.

Usage

The gap element is used in nonlinear static analysis (SOL 106 or 66) and nonlinear transient analysis (SOL 129 or 99). The gap element provides point-to-point contact. When the gap element is open, there is no contact and no friction. When the gap element is closed, there are three different conditions. The first case is when the gap is sliding (no friction). The second case occurs when the gap element is sticking (static friction). The third case occurs when the gap element is slipping (kinetic friction). Sometimes, it may be difficult to converge because of the switching among these conditions.

Limitations

One of the contact surfaces should not rotate by a large angle because the gap element orientation is not updated for large rotations.

Input

How to Use the CGAP and PGAP Bulk Data Entries

Often a mistake is made in the orientation of the gap element, which is defined through G0 or X1, X2, and X3. If the grid points GA and GB are not coincident, then line AB is the element x-axis and the orientation vector lies in the X-Y plane. The use of coincident grid points is not recommended.

On the PGAP Bulk Data entry, U0 and F0 are the initial gap opening and the preload, respectively. U0 is the separation of the gap element. If you are unsure about what preload to use, use the default value of zero. KA, KB, and KT are called the penalty values. KA is the axial stiffness when the gap is closed, KB is the axial stiffness when the gap is open, and the KT is the transverse gap stiffness. All you need to input is the KA value. KB is defaulted to be zero, and KT is defaulted as a function of KA. The selection of KA or a penalty value is critical to the convergence of the problem. The non-adaptive gap does not update this KA value. The new adaptive gap updates this KA value if you specify a positive value for TMAX. The default option for the new adaptive gap is NO adaptive stiffness update. This is when the TMAX value is equal to 0.0. If TMAX is a negative number, then MSC.Nastran uses the non-adaptive gap formulation. A good initial KA selection is roughly 1000 times the Young's modulus of the material that is in the neighborhood of the gap element.

Fields 8 and 9 also require some explanation. For the new adaptive value this is MU1, which is the static coefficient of friction, and MU2, which is the dynamic coefficient of friction. The non-adaptive gap elements use MUY and MUZ in the same fields instead of MU1 and MU2 values. Therefore, to compare the results using adaptive and non-adaptive gap elements, set both of these entries to the same values. On most problems, you will not know how to estimate the penetration depth or field 2 on the continuation entry. Therefore, if you skip the continuation entry, the default new adaptive gap is obtained, which is everything adaptive except the KA update.

The recommended allowable penetration of TMAX is 10 percent of the thickness for the shell element, and for the solid element it is roughly 1/10,000 of the characteristic length for the problem. A small positive value will turn on the adaptive stiffness update, which will then converge at a faster rate.

The maximum adjustment ratio or MAR in field 3 of the continuation entry is used only for the penalty value adjustment of the adaptive gap element. The default value of 100 is sufficient for most problems.

Output

The gap element output is obtained by the STRESS output request in the Case Control. An example of the gap output format is shown in [Listing 13-11](#). The load step is 1.95 on the top left-hand corner. The digit on the left-hand side of the decimal indicates that subcase 1 is completed. The digits on the right-hand side of the decimal point indicate that 95 percent of subcase 2 is completed. The far left column lists the gap element numbers. The second column provides the forces in the axial direction of the gap element. The third and the fourth columns represent forces in the Y and Z direction. These forces should be zero if the gap is in the open position. Columns 5 to 9 represent the displacements of the gap in its element coordinate system. The last column indicates the status of the gap. This status can be “open”, “stick”, or “slip”. Sticking is when the ratio of (shear force)/coefficient of friction * normal force) = SR is such that $0 < SR < 1$. Slip occurs when $SR = 1.0$, open is when the gap is not in contact; hence $SR = 0$.

Listing 13-11 Sample of CGAP Output

```

NONLINEAR STATIC CONTACT OF A SPHERE ON A RIGID          NG6603          NOVEMBER 10, 1992 MSC/NASTRAN 11/ 9/92  PAGE 177
  USE TMAX=0.0016,FULLY ADAPTIVE FEATURE
  KA=1.0E6
  LOAD STEP = 1.92500E+00
                S T R E S S E S   ( F O R C E S )   I N   G A P   E L E M E N T S   ( C G A P )

ELEMENT  - F O R C E S   I N   E L E M   S Y S T -   - D I S P L A C E M E N T S   I N   E L E M   S Y S T -
  ID      COMP-X      SHEAR-Y      SHEAR-Z      AXIAL-U      TOTAL-V      TOTAL-W      SLIP-V      SLIP-W      STATUS
2001     3.60382E+00  -4.47466E-01  0.0          2.23604E-02  -8.30369E-04  0.0          -6.81213E-04  0.0          STICK
2003     5.04829E+00  -1.10082E+00  0.0          8.85048E-02  -7.80763E-03  0.0          -7.44069E-03  0.0          STICK
2005     0.0          0.0          0.0          1.95640E-01  -1.70540E-02  0.0          -1.70540E-02  0.0          OPEN
2007     0.0          0.0          0.0          2.23109E-01  -9.91112E-03  0.0          -9.91112E-03  0.0          OPEN
2009     0.0          0.0          0.0          2.40853E-01  -4.95583E-03  0.0          -4.95583E-03  0.0          OPEN
NONLINEAR STATIC CONTACT OF A SPHERE ON A RIGID          NG6603          NOVEMBER 10, 1992 MSC/NASTRAN 11/ 9/92  PAGE 178
  USE TMAX=0.0016,FULLY ADAPTIVE FEATURE
  KA=1.0E6
  LOAD STEP = 1.95000E+00
                S T R E S S E S   ( F O R C E S )   I N   G A P   E L E M E N T S   ( C G A P )

ELEMENT  - F O R C E S   I N   E L E M   S Y S T -   - D I S P L A C E M E N T S   I N   E L E M   S Y S T -
  ID      COMP-X      SHEAR-Y      SHEAR-Z      AXIAL-U      TOTAL-V      TOTAL-W      SLIP-V      SLIP-W      STATUS
2001     3.64258E+00  -4.47838E-01  0.0          2.23643E-02  -8.30493E-04  0.0          -6.81213E-04  0.0          STICK
2003     5.15982E+00  -1.12016E+00  0.0          8.85160E-02  -7.81408E-03  0.0          -7.44069E-03  0.0          STICK
2005     8.83100E-02  -5.82848E-03  0.0          1.97000E-01  -1.73532E-02  0.0          -1.73532E-02  0.0          STICK
2007     0.0          0.0          0.0          2.25831E-01  -1.01834E-02  0.0          -1.01834E-02  0.0          OPEN
2009     0.0          0.0          0.0          2.44078E-01  -5.09924E-03  0.0          -5.09924E-03  0.0          OPEN
NONLINEAR STATIC CONTACT OF A SPHERE ON A RIGID          NG6603          NOVEMBER 10, 1992 MSC/NASTRAN 11/ 9/92  PAGE 179
  USE TMAX=0.0016,FULLY ADAPTIVE FEATURE
  KA=1.0E6
  LOAD STEP = 1.97500E+00
                S T R E S S E S   ( F O R C E S )   I N   G A P   E L E M E N T S   ( C G A P )

ELEMENT  - F O R C E S   I N   E L E M   S Y S T -   - D I S P L A C E M E N T S   I N   E L E M   S Y S T -
  ID      COMP-X      SHEAR-Y      SHEAR-Z      AXIAL-U      TOTAL-V      TOTAL-W      SLIP-V      SLIP-W      STATUS
2001     3.66578E+00  -4.47205E-01  0.0          2.23666E-02  -8.30282E-04  0.0          -6.81213E-04  0.0          STICK
2003     5.22790E+00  -1.12097E+00  0.0          8.85228E-02  -7.81435E-03  0.0          -7.44069E-03  0.0          STICK
2005     2.37324E-01  -3.59185E-02  0.0          1.97024E-01  -1.73652E-02  0.0          -1.73532E-02  0.0          STICK
2007     0.0          0.0          0.0          2.27630E-01  -1.03924E-02  0.0          -1.03924E-02  0.0          OPEN
2009     0.0          0.0          0.0          2.46395E-01  -5.21857E-03  0.0          -5.21857E-03  0.0          OPEN
NONLINEAR STATIC CONTACT OF A SPHERE ON A RIGID          NG6603          NOVEMBER 10, 1992 MSC/NASTRAN 11/ 9/92  PAGE 180
  USE TMAX=0.0016,FULLY ADAPTIVE FEATURE
  KA=1.0E6
  LOAD STEP = 2.00000E+00
                S T R E S S E S   ( F O R C E S )   I N   G A P   E L E M E N T S   ( C G A P )

ELEMENT  - F O R C E S   I N   E L E M   S Y S T -   - D I S P L A C E M E N T S   I N   E L E M   S Y S T -
  ID      COMP-X      SHEAR-Y      SHEAR-Z      AXIAL-U      TOTAL-V      TOTAL-W      SLIP-V      SLIP-W      STATUS
2001     3.68888E+00  -4.46565E-01  0.0          2.23689E-02  -8.30068E-04  0.0          -6.81213E-04  0.0          STICK
2003     5.29569E+00  -1.12167E+00  0.0          8.85296E-02  -7.81458E-03  0.0          -7.44069E-03  0.0          STICK
2005     3.86731E-01  -6.51561E-02  0.0          1.97039E-01  -1.73749E-02  0.0          -1.73532E-02  0.0          STICK
2007     0.0          0.0          0.0          2.29423E-01  -1.06010E-02  0.0          -1.06010E-02  0.0          OPEN
2009     0.0          0.0          0.0          2.48707E-01  -5.33770E-03  0.0          -5.33770E-03  0.0          OPEN

```

Example: Hertz Contact Stress

Hertz contact stress is a well known problem arising in the contact of a wheel and a rail. To guard against the possibility of surface failure, it is necessary to calculate the stress that results from loading one body against another. The gap element is used to model the point-to-point contact phenomenon. In this case, the gap element represents the contact surface. The Hertz contact stress problem considered here is a nonlinear static analysis using SOL 106.

Modeling Techniques

For the static analysis, a symmetric half of a 10-degree sector of the sphere is modeled by CHEXA and CPENTA elements with axisymmetric boundary conditions as shown in [Figure 13-19](#). Contact with a rigid plane is simulated by adaptive gap elements.

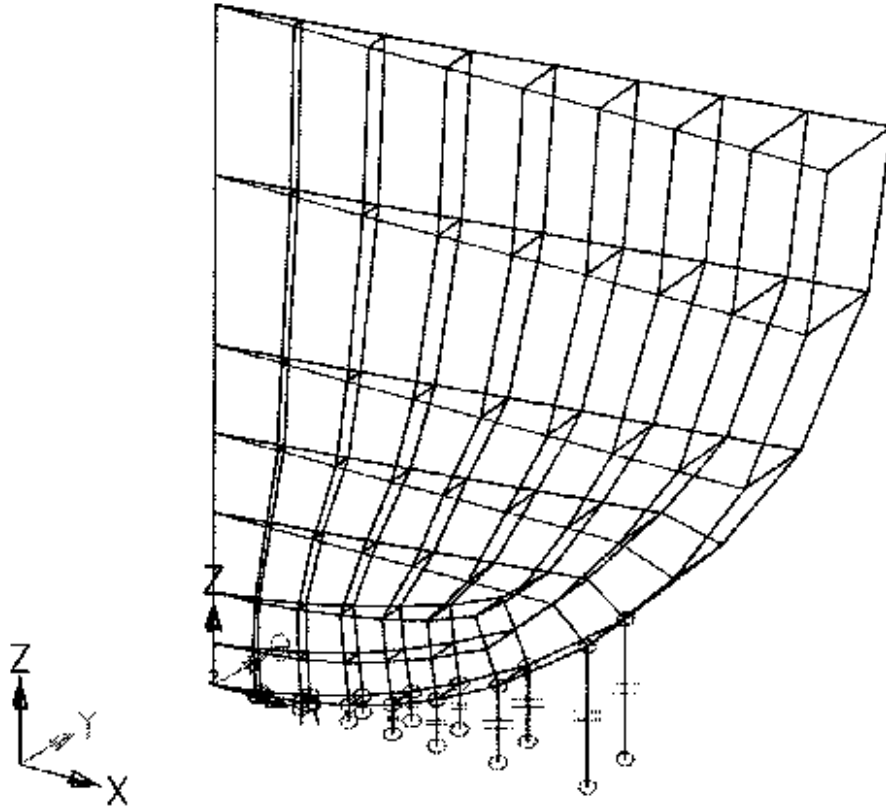


Figure 13-19 Hertz Contact Stress

Uniform vertical displacement of the symmetric plane is maintained by constraining the vertical movement of all the nodes on the top plane by MPC relations at the center line. The radius of the sphere is 8 inches, Young's modulus is 1000 psi, and Poisson's ratio is 0.3. The static analysis is performed in 17 increments up to the maximum loading. The characteristic length of the problem is the diameter of the sphere, i.e., 16 inches. Therefore, the TMAX value is $16/10000$, or 0.0016. The initial KA or axial gap stiffness is selected to be 1000 times Young's modulus. These values are inserted in the PGAP entry as shown in [Figure 13-20](#).

	1	2	3	4	5	6	7	8	9	10
\$	EID	PID	GA	GB	X1	X2	X3			
CGAP	2001	1	105	301	1.	0.	0.			

\$	PID	U0	F0	KA	KB	KT	MU1	MU2	
PGAP	1	.022		1.+6			.3	.3	+
\$	TMAX	MAR	TRMIN						
+	0.0016								

Figure 13-20 Partial Input Data

Discussion of Results

The objective of this example is to demonstrate that the adaptive features of the gap element allows the user to use a wider range of KA values, and still get good results. **Figure 13-21** shows that if the adaptive features of the gap element are used, good results are obtained for KA values ranging from 1.0E1 to 1.0E6. If adaptive features are not used, good results are obtained for KA values between 1.0E3 and 1.0E5. For KA values higher than 1.0E5, the non-adaptive gap element failed to converge, and below 1.0E3, the accuracy decreased.

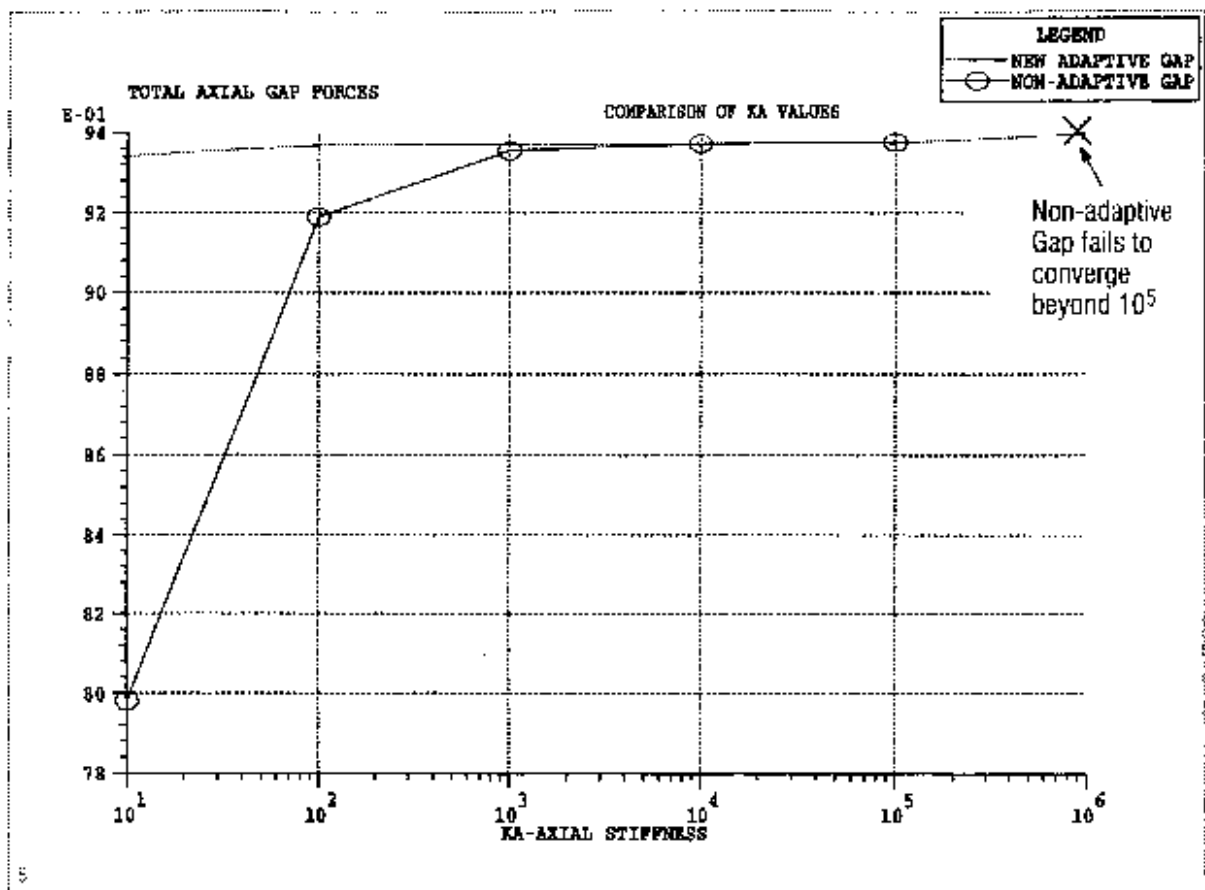


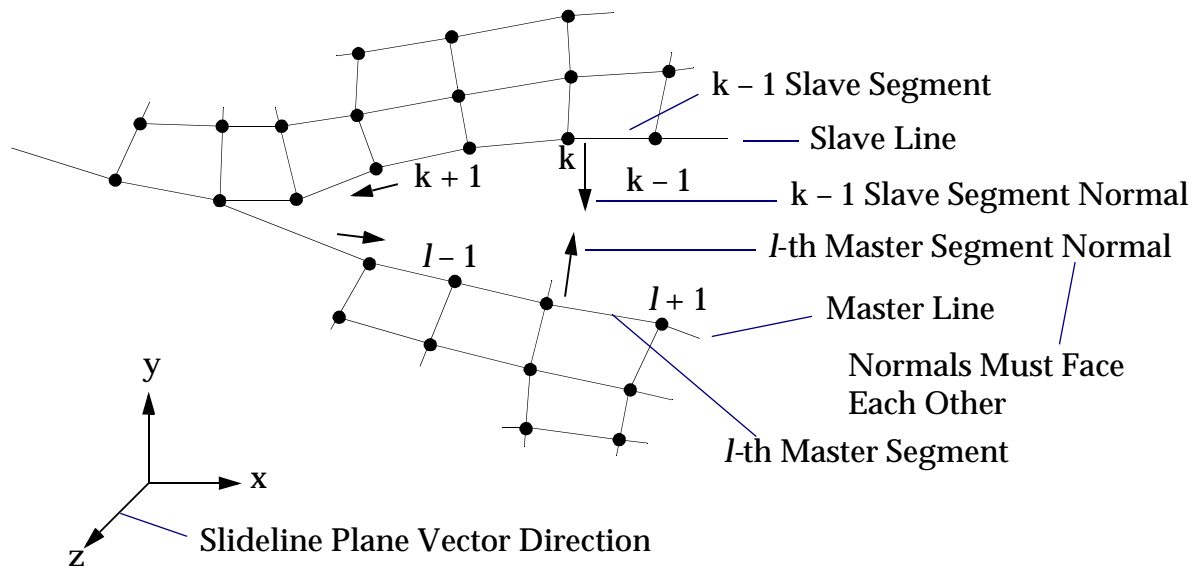
Figure 13-21 Comparison of Axial Stiffness Values

13.3.13 3-D Slideline Contact Capability

General Description

Slide line contact allows the user to model interactions between two deformable bodies. One of the deformable bodies is called the master and the other is called the slave. The modeling of interaction requires the user to define contact regions in terms of slide lines. The user may specify as many slide lines as desired.

A slide line contact region consists of a master line and a slave line as shown in **Figure 13-22**. A master line is a list of grid points in the topological order on the master body. A slave line is a list of grid points in the topological order on the slave body. The grid points on the master line are called the master nodes, and on the slave line are called the slave nodes. A line segment joining two consecutive master nodes is called a master segment. Thus a master line, in general, consists of number of master segments. A minimum of one master segment consisting of two master nodes are required for the master line. Similar to master line, a slave line, in general, consist of a number of slave segments. However, a slave line may not have any slave segments; it may have just one slave node.



- Arrows show positive direction for ordering nodes. Counterclockwise from master line to slave line.
- X-Y plane is the slideline plane. Unit normal in the Z-direction is the slideline plane vector.
- Slave and master segment normals must face each other; i.e., the normals must be in the opposite directions.

Figure 13-22 Typical Slideline Contact Region

The master and the slave nodes in a contact region must lie in a plane called the slide line plane. The slide line plane is defined by a vector normal to the plane called the slide line plane vector. The master and slave can have large relative motions within the slide line plane. However, relative motion outside the slide line plane are ignored, therefore, must be small compared to a typical master segment. Thus, slide line contact is ideal for modelling interactions between two bodies which may come in contact in specific planes. MSC.Nastran checks to make sure that in the initial geometry all the nodes of a contact region lie in the slide line plane. However, no such check is made during analysis.

In general, contact is determined only for slave nodes and the master line. This may result in master nodes penetrating the slave line. However, the error involved depends only on the mesh discretization. At the expense of increase in CPU cost, a user may wish to determine contact between master nodes and the slave line in addition to determining contact between slave nodes and the master line. This is called symmetric penetration. Thus, for symmetric penetration, no distinction is made between slave and master.

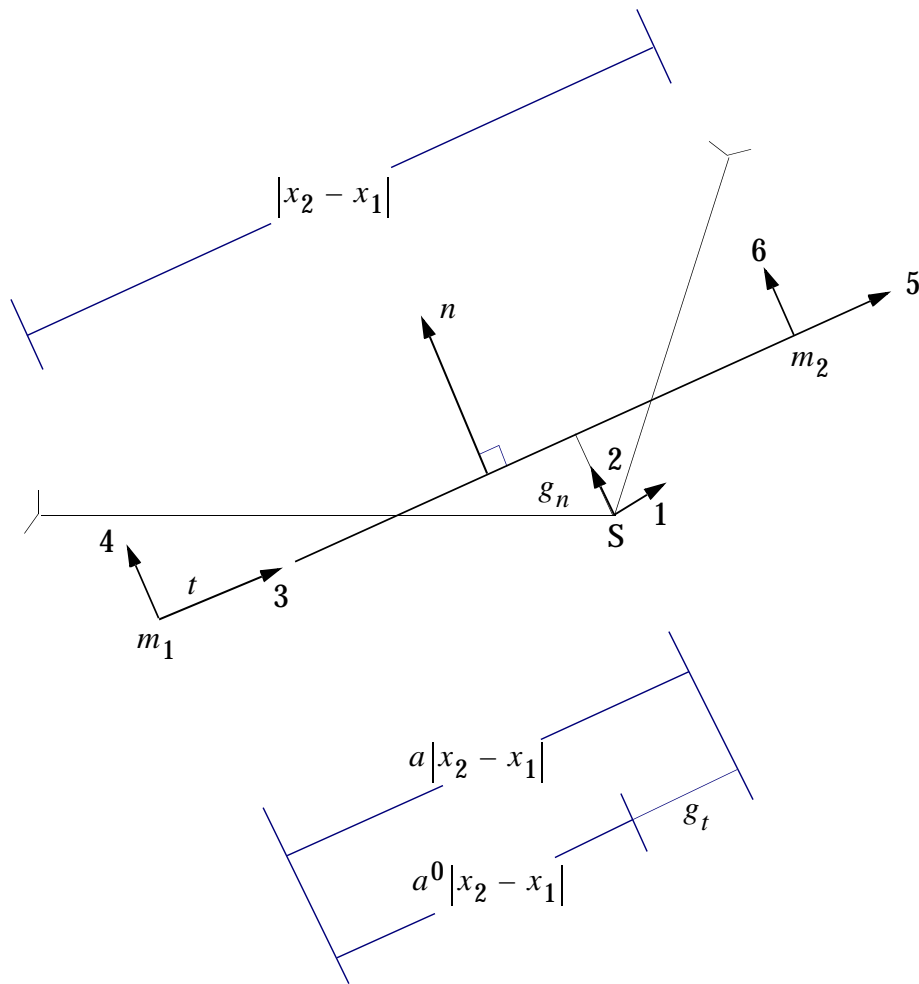
To determine contact between the slave nodes and the master line, a local coordinate system is created for each master segment. The tangent direction for the master segment is given by the vector from its first node to the second node. Remember that the master nodes are specified in the topological order. The normal direction to the master segment is determined by the cross product between the slide line plane vector and the tangent vector. This normal direction must point from master line to the slave line. This is because MSC.Nastran determines the contact between slave nodes and the master line by measuring how close the slave nodes are to the master segment in this normal direction.

For symmetric penetration, similar to the master segments, tangent and normal directions are also defined for slave segments. Now, the normals of master segments and slave segments must face each other as shown in [Figure 13-22](#). This is generally accomplished by traversing from master line to slave line in a counter-clockwise fashion, or clockwise fashion depending on whether the slideline plane vector forms a right hand or a left hand coordinate system with the slide line plane. It is the user's responsibility to make sure that the normals face each other. MSC.Nastran checks indirectly to make sure that the normals face each other by calculating the initial gap between the two bodies. However, this is not a foolproof check, especially when there is no initial clearance.

Initial penetration between the two bodies is not allowed. The initial gaps between the slave nodes and the master line are calculated based on the coordinates specified for the slave and the master nodes. Thus, if there is no initial gap, MSC.Nastran may

calculate initial penetration if the coordinates are not specified accurately. To avoid having the user to calculate the coordinates very accurately, MSC.Nastran adjusts the coordinates automatically and issues a warning message if the initial penetration of a slave node is less than ten percent of the length of the master segment to which it projects. However, if the initial penetration is larger than ten percent, the analysis is terminated and a fatal message is issued. Sometimes, the analysis may be terminated with this message even though there is a large initial gap between the two bodies. This implies that the normals of master or slave segments are not pointing in the correct direction.

Once a contact between a node and a body is detected, a three node slide line element as shown in **Figure 13-23** is automatically created. The first node of this element is the slave node and the last two nodes are the master nodes of the master segment with which the slave node is in contact. The slave node is forced to slide on the master line after contact and must remain on the master line until a tensile force develops.



- S, m_1 , , m_2 = slave, master node 1 and master node 2, respectively
- a, a_0 = current and previous surface coordinate
- g_n = penetration of slave node into the master segment
- g_t = sliding of the slave node on the master segment
- n = normal direction for the master segment

Figure 13-23 Geometry and Definition of a Typical Three Node Slide Line Element

In MSC.Nastran the contact and traction forces/stresses are always associated with slave nodes. In order to compute the contact and traction stresses, an area is associated with each slave node, based on the contributory length and the width/thickness from the adjacent slave segments. Therefore, the user needs to specify widths associated with slave line in a three dimensional model or thickness associated with slave line in a two dimensional model. A user can request to output

results for any number of slave nodes in any number of slide line contact regions. The output for a slave node consists of (a) the slide line contact region identification number, (b) the master segment to which it projects, (c) the parametric surface coordinate to identify the exact projection of slave node relative to the two master nodes of the master segment, (d) contact and tangential forces and stresses in the element coordinate system, and (e) a slip ratio to indicate whether the slave node is sticking, slipping (or sliding). The slip ratio is defined as the (shear force)/(coefficient of friction * normal force). Thus, a ratio of 0. indicates no shear or no contact. A slip ratio of 1. indicates slipping. A value between 0. and 1. is a normalized value which indicates how close the value of the shear force is to impending slipping.

The slide line contact formulation is based on the penalty method which uses an artificial stiffness value. However, the user need not provide penalty values. MSC.Nastran will automatically calculate the appropriate numbers. The user has the option to scale the calculated penalty values. It is possible that the automatically calculated penalty values, may sometimes, cause convergence difficulties. However, a user can overcome the convergence problems by restarting the analysis from any previous converged solution and scaling down the penalty values. Generally, it is recommended to scale down the penalty values by one order of magnitude at a time.

Diagnostic Output

Diagnostic output related to 3-D slide line contact can be obtained by setting the executive control statement DIAG to 35.

DIAG 35 Prints current coordinates of contact node and other information associated with slave nodes to debug slide line contact algorithm for each iteration for slide line contact.

Limitations

The slide line capability is only available for SOL 106 and SOL 129. For all other solution sequences all the Bulk Data entries and Case Control commands related to slide line contact are ignored.

Contact stress or force for any slave node could be plotted using XYPLOT BOUT RESP command. The grid ID is the slave node external grid ID. However, the slave node requested for plotting must also be requested for printout via the BOUTPUT Case Control command.

The user interface for slideline contact consists of one Case Control command, five Bulk Data entries and one PARAMeter. The Case Control command is BOUTPUT. The Bulk Data entries are BCONP, BLSEG, BWIDTH, BFRIC, and BOUTPUT. The parameter is ADPCON.

Example

The purpose of this example is to demonstrate how to use the features of slide line contact capability in MSC.Nastran.

An elastic sphere in contact with a rigid plane is pressed down with a uniform pressure at the top. No friction is assumed between the sphere and the rigid plane. The analytical solution to this problem were first published by Timoshenko, S. P., and Goodier, J. N., *Theory of Elasticity*, McGraw-Hill Book Company, 1970, pgs. 409-420. For demonstration purposes, a symmetric half of a ten degree sector of the sphere is modeled by CHEXA and CPENTA elements with axisymmetric boundary conditions (see [Figure 13-24](#)). Uniform vertical displacements of the symmetry plane is maintained by tying the vertical displacements of all the grid points on the top plane (loading plane) by MPC relations to the one at the center line (z-axis). A total load of 2160 is applied at the top of sphere. Because only a ten degree sector is modeled and because of the MPC relationships, a load of only 60 is applied at the top center grid point. This load is applied in three subcases with five load increments per subcase.

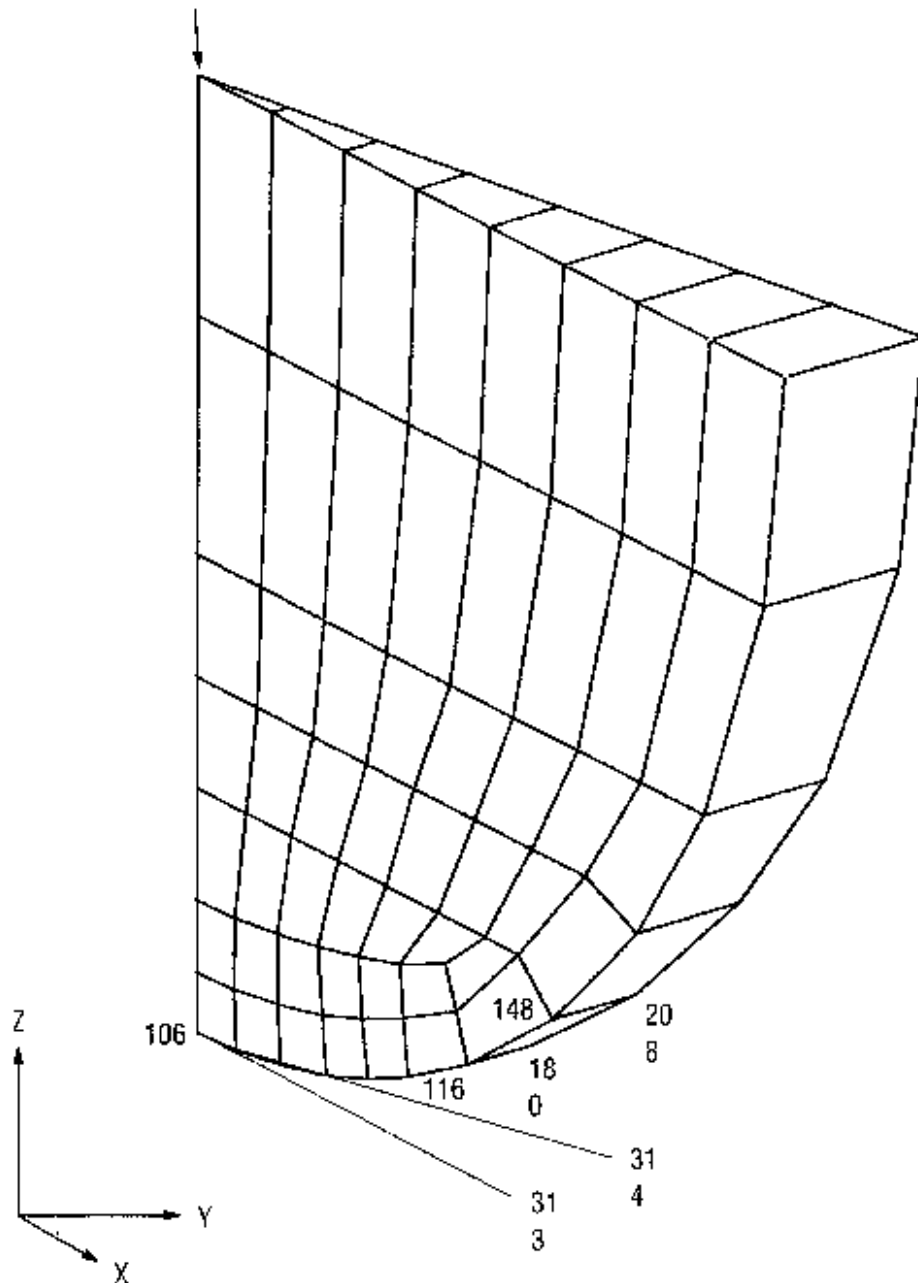


Figure 13-24 Model of a Sphere in Contact with Rigid Plane

To model the contact, two slide line regions are defined via the BCONP Bulk Data entry. The first slide line contact region models the contact between the outer grid points of the sphere at $\theta = 0^\circ$ and the rigid plane as shown in [Figure 13-25](#). The outer grid points of the sphere (148, 116, 114, 112, 110, 108, 105) are used to define the slave line. To define the master line a grid point 313 at $\theta = 0^\circ$ is defined on the rigid plane. The master line is defined by the lower most center grid point of the sphere (grid point 106) and the grid point 313. Notice that the grid points on the master and slave are defined in topological order traversing in counter-clockwise form the master line to

the slave line (see BCONP with ID = 10, BLSEG with ID's = 10 and 20 in the example input). This is because the slide line plane is the R-Z plane at $\theta = 0^\circ$ and the slide line plane vector forms a right hand system with the slide line plane (see CORD2R with ID = 1 in the example listing).

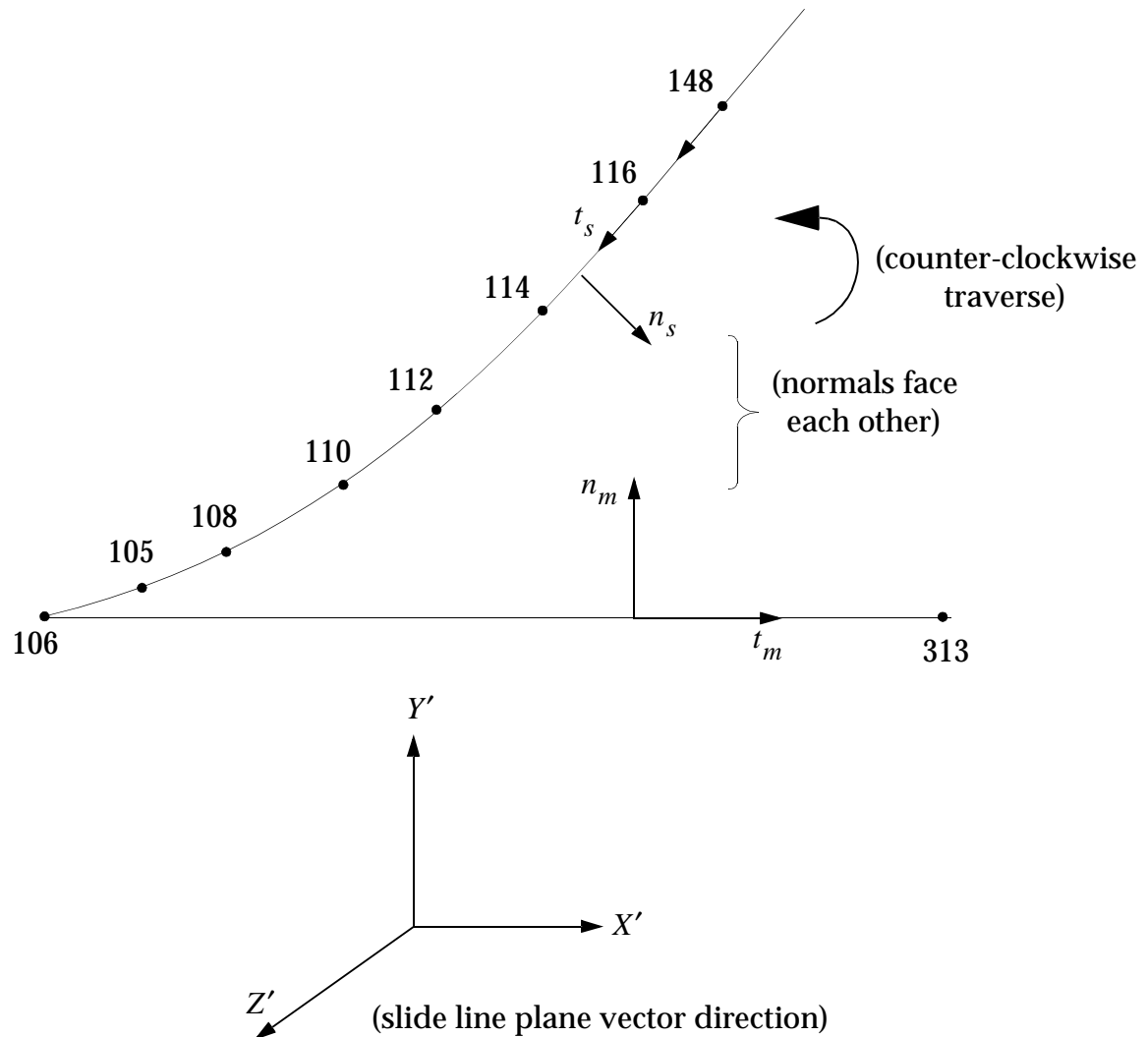


Figure 13-25 Slide Line Contact Region One

The second slide line contact region models the contact between the outer grid points of the sphere at $\theta = 10^\circ$ and the rigid plane as shown in [Figure 13-26](#). Another grid point 314 at $\theta = 10^\circ$ on the rigid plane is defined to model the master line. This time we traverse clockwise from the master line to the slave line in topological order (see BCONP with ID = 20 and BLSEG with ID's = 30 and 40 in the example input) because the slide line plane vector forms a left hand system with the slide line plane (see CORD2R with ID = 3 in the example input).

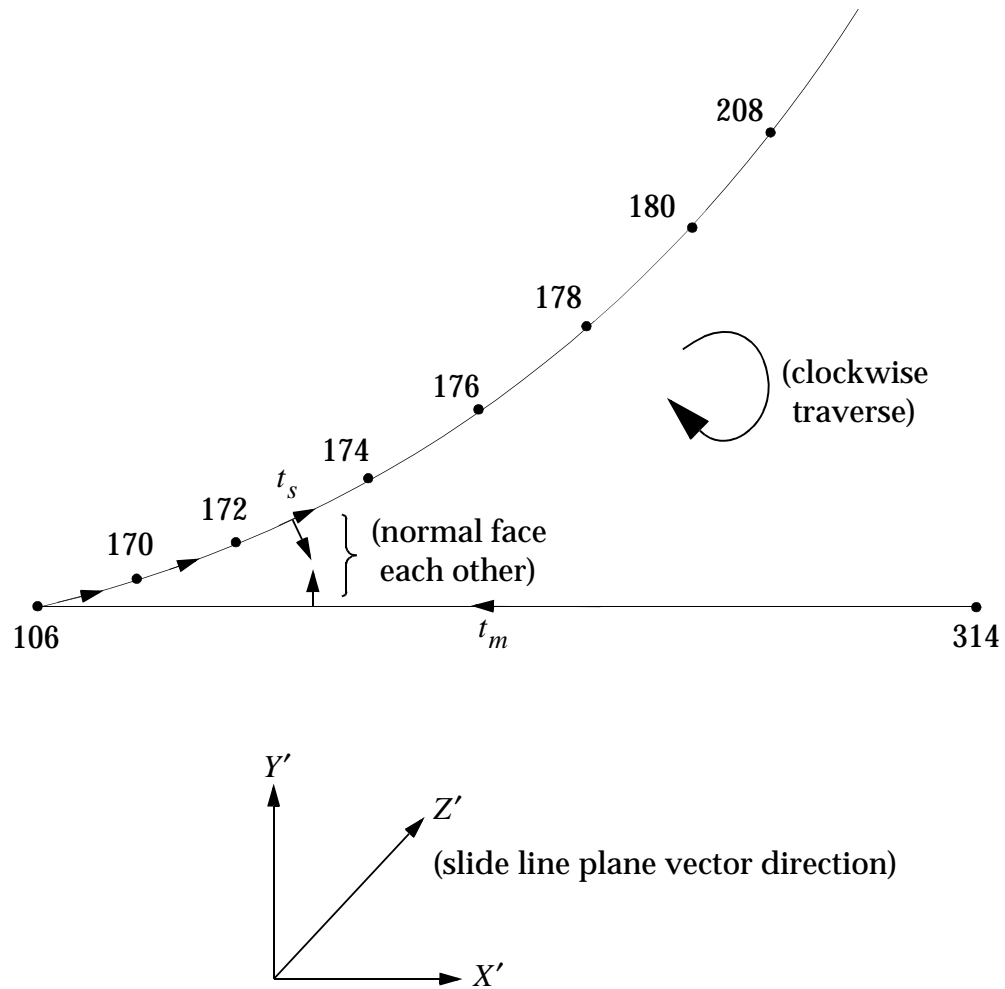


Figure 13-26 Slide Line Contact Region Two

The most important issue is to make sure that the normals of master and slave line point towards each other. The normal to a segment is obtained by $e_3 \times t$. e_3 is the unit vector in the z-direction of the coordinate system used in field CID of BCONP Bulk Data entry. t is the tangential vector from node 1 to node 2 of the line in the order specified in the BLSEG Bulk Data entry.

Output Formats

SORT1 Output

```

NONLINEAR STATIC CONTACT OF A SPHERE ON A RIGID PLANE WITHOUT FRIC      MARCH  1, 1993  MSC/NASTRAN  2/17/93  PAGE  109
                                SUBCASE 3
LOAD STEP =  3.00000E+00
                                RESULTS FOR SLIDE LINE ELEMENTS (IN ELEMENT SYSTEM)

```

SLAVE GRID	CONTACT ID	MASTER GRID1	MASTER GRID2	SURFACE COORDINATE	NORMAL FORCE	SHEAR FORCE	NORMAL STRESS	SHEAR STRESS	NORMAL GAP	SLIP	SLIP RATIO	SLIP CODE
148	1	106	313	9.7604E-01	.0	.0	.0	.0	-1.1240E+00	1.5218E-02	.0	OPEN
116	1	106	313	7.3917E-01	.0	.0	.0	.0	-4.1923E-01	-4.1647E-03	.0	OPEN
114	1	106	313	5.7432E-01	.0	.0	.0	.0	-1.1861E-01	-2.3407E-02	.0	OPEN
112	1	106	313	4.5907E-01	2.4318E+00	.0	4.1069E+00	.0	1.2792E-04	-3.9671E-02	.0	SLIDE
110	1	106	313	3.4298E-01	1.1404E+01	.0	1.8499E+01	.0	5.9989E-04	-4.8093E-02	.0	SLIDE
108	1	106	313	2.1905E-01	1.0069E+01	.0	1.6320E+01	.0	5.2968E-04	-3.5747E-02	.0	SLIDE
105	1	106	313	1.0478E-01	5.2337E+00	.0	1.7661E+01	.0	2.7531E-04	-1.8078E-02	.0	SLIDE
170	2	314	106	8.9522E-01	5.2335E+00	.0	1.7660E+01	.0	2.7530E-04	1.8078E-02	.0	SLIDE
172	2	314	106	7.8095E-01	1.0069E+01	.0	1.6320E+01	.0	5.2968E-04	3.5747E-02	.0	SLIDE
174	2	314	106	6.5702E-01	1.1404E+01	.0	1.8499E+01	.0	5.9989E-04	4.8093E-02	.0	SLIDE
176	2	314	106	5.4093E-01	2.4318E+00	.0	4.1069E+00	.0	1.2792E-04	3.9671E-02	.0	SLIDE
178	2	314	106	4.2568E-01	.0	.0	.0	.0	-1.1861E-01	2.3407E-02	.0	OPEN
180	2	314	106	2.6083E-01	.0	.0	.0	.0	-4.1923E-01	4.1647E-03	.0	OPEN
208	2	314	106	2.3956E-02	.0	.0	.0	.0	-1.1240E+00	-1.5218E-02	.0	OPEN

SORT2 Output

```

NONLINEAR STATIC CONTACT OF A SPHERE ON A RIGID PLANE WITHOUT FRIC      MARCH  1, 1993  MSC/NASTRAN  2/17/93  PAGE  99
                                SUBCASE 1
SLAVE GRID-ID =  110
                                RESULTS FOR SLIDE LINE ELEMENTS (IN ELEMENT SYSTEM)

```

STEP	CONTACT ID	MASTER GRID1	MASTER GRID2	SURFACE COORDINATE	NORMAL FORCE	SHEAR FORCE	NORMAL STRESS	SHEAR STRESS	NORMAL GAP	SLIP	SLIP RATIO	SLIP CODE
2.000E-01	1	106	313	3.5166E-01	.0	.0	.0	.0	-1.0336E-01	-4.6982E-03	.0	OPEN
4.000E-01	1	106	313	3.5067E-01	.0	.0	.0	.0	-7.1160E-02	-9.6436E-03	.0	OPEN
6.000E-01	1	106	313	3.4968E-01	.0	.0	.0	.0	-4.3237E-02	-1.4611E-02	.0	OPEN
8.000E-01	1	106	313	3.4868E-01	.0	.0	.0	.0	-1.5314E-02	-1.9579E-02	.0	OPEN
1.000E+00	1	106	313	3.4787E-01	5.6225E-01	.0	9.1206E-01	.0	2.9576E-05	-2.3653E-02	.0	SLIDE
1.200E+00	1	106	313	3.4727E-01	1.8074E+00	.0	2.9318E+00	.0	9.5072E-05	-2.6643E-02	.0	SLIDE
1.400E+00	1	106	313	3.4667E-01	3.0525E+00	.0	4.9516E+00	.0	1.6057E-04	-2.9632E-02	.0	SLIDE
1.600E+00	1	106	313	3.4608E-01	4.2976E+00	.0	6.9713E+00	.0	2.2606E-04	-3.2622E-02	.0	SLIDE
1.800E+00	1	106	313	3.4548E-01	5.5427E+00	.0	8.9911E+00	.0	2.9156E-04	-3.5612E-02	.0	SLIDE
2.000E+00	1	106	313	3.4488E-01	6.7878E+00	.0	1.1011E+01	.0	3.5706E-04	-3.8601E-02	.0	SLIDE
2.200E+00	1	106	313	3.4428E-01	8.0329E+00	.0	1.3031E+01	.0	4.2255E-04	-4.1591E-02	.0	SLIDE
2.400E+00	1	106	313	3.4368E-01	9.2780E+00	.0	1.5050E+01	.0	4.8805E-04	-4.4580E-02	.0	SLIDE
2.600E+00	1	106	313	3.4322E-01	1.0329E+01	.0	1.6756E+01	.0	5.4336E-04	-4.6913E-02	.0	SLIDE
2.800E+00	1	106	313	3.4310E-01	1.0867E+01	.0	1.7628E+01	.0	5.7162E-04	-4.7503E-02	.0	SLIDE
3.000E+00	1	106	313	3.4298E-01	1.1404E+01	.0	1.8499E+01	.0	5.9989E-04	-4.8093E-02	.0	SLIDE

13.3.14 Nonlinear Static Analysis

User Interface

Nonlinear static analysis can be performed using SOLs 66, 106, or 153. However, slideline contact is not available in SOL 66. Modeling options are compatible with linear analysis, i.e., a SOL 101 model can be converted to SOL 106 analysis with a few additional data entries relevant to the nonlinear analysis. The nonlinear properties and/or effects are defined by nonlinear material data (e.g., MATS1, MATHP, CREEP, and TABLES1), gap elements (GAP), slideline contact (BCONP, BLSEG and BFRIC) for nonlinear interface, and PARAMeter LGDISP for geometric nonlinearity.

Case Control

Each subcase defines a set of loading and boundary conditions which can be subdivided into a number of increments. Loading conditions are specified using the same loads as in linear static analysis, i.e., FORCE_i, MOMENT_i, PLOAD_i, GRAV, TEMP, etc. Constraints can be specified using SPCs and MPCs, which can be changed from subcase to subcase. The enforced motion can be applied using SPCD or nonzero value in SPC. The specified value for SPCD or SPC is a measure of displacement with reference to the undeformed position. The enforced motion is also subdivided in the subcase like the incremental loads. The unique data required for SOL 106 is supplied on the NLPARM entry, which controls the incremental and iterative solution processes. The snap-through analysis or post-buckling analysis can be performed using arc-length methods which are controlled by parameters in the NLPCI Bulk Data entry.

The load step is labeled by the cumulative load factor. The load factor varies from 0 to 1 in each subcase. Then the load step will end with 1, 2, 3, etc. for the first, the second and the third subcase, respectively. There are advantages to divide the entire loading history into many subcases so that each subcase does not have excessive number of increments, e.g., not to exceed 20 steps. The data blocks containing solutions can be generated at each increment or the end of each subcase depending on the intermediate output option specified in the INTOUT field of NLPARM data entry. Data blocks are stored in the database for output process and restarts. As such, the database size or the output quantities are better controlled with multiple subcases. Furthermore, the nonproportional loading can only be achieved by using multiple subcases.

Output requests for each subcase are processed independently. Requested output quantities for all the subcases are appended after the computational process for actual output operation. Available outputs are DISPLACEMENT, OLOAD, STRESS, FORCE, GPSTRESS, and SPCFORCE. However, FORCE and GPSTRESS output are available only for linear elements. It should be noted that the STRESS or STRESS(PLOT) command will cause the nonlinear stress to be printed. In other words, the STRESS(PLOT) command suppresses the combined linear/nonlinear stress output only.

Restarts are controlled by parameters (LOOPID and SUBID) which can be specified either in the Case Control Section or Bulk Data Section. Some optional parameters (ADPCON, BUCKLE, LANGLE, MAXLP, NMLOOP) are provided for additional control or capabilities. All the superelement model generation options and matrix reduction options are allowed for the linear portion of the structure.

Restriction and Limitations in SOL 106

1. Only CROD, CONROD, CTUBE, CBEAM, CGAP, CQUAD4, CQUAD8, CQUADX, CTRIA3, CTRIA6, CTRIA9, CHEXA, CPENTA, and CTETRA elements may be defined with material or geometric nonlinear properties. All other elements will be treated as having small displacements and linear materials.
2. The CBEAM is assumed to have its nonlinear region concentrated at the ends. Taper is allowed but pin joints are illegal for material nonlinearity. However, pin joints can be used if the only nonlinearity is geometric.
3. Constraints apply only to the nonrotated displacements at a grid point. In particular, multipoint constraints and rigid elements may cause problems if the connected grid points undergo large motions. However, also note that replacement of the constraints with overly stiff elements can result in convergence problems.
4. Large deformations of the elements may cause nonequilibrium loading effects. The elements are assumed to have constant length, area, and volume, except for hyperelastic elements.
5. All restarts with grid point or element changes must start over at the first subcase.
6. For large displacement analysis, there are two different approaches for the angular motions: gimbal angle and rotation vector approach. In Gimbal angle approach angular motions are treated as three ordered rotations about the z, y, and x axes. In rotation vector approach the three angular motions are treated as a vector. The rotation is about the rotation axis and the magnitude of rotation is equal to the amplitude of rotation vector.
7. The CGAP element contact planes do not rotate as a function of displacement. The actual shapes of the contacting surfaces would be required for such calculations.
8. Since the solution to a particular load involves a nonlinear search procedure, solution is not guaranteed. Care must be used in selecting the search procedures on the NLPARM data. Nearly all iteration control restrictions may be overridden by the user.
9. Follower force effects are calculated for loads which change direction with grid point motion.

10. All cold-start runs are assumed to start in a stress-free condition with zero displacements and rotations. Prestressed and preloaded cases must be simulated by running them as the initial load case, followed by the desired solution subcase.
11. Every subcase must define a new total load or enforced boundary displacement. This is necessary because error ratios are measured with respect to the load change.
12. All superelements, except for the residual ($SE = 0$), are assumed to be linear. All existing superelement restrictions also apply here.
13. Any changes to a nonresidual superelement requires the use of the appropriate SEMG, SEMA, SELA, etc., case control request for efficient restarts (see also “[Nonlinear Analysis With Superelements](#)” on page 715).
14. Only the residual superelement has automatic output data processing. Upstream superelements may be loaded only via LSEQ and CLOAD data.
15. In postbuckling analysis with arc-length methods (NLPCI Bulk Data entry), the iteration algorithm may jump between different buckling modes when multiple bifurcation points are present in the structure. It is recommended to check all possible buckling modes with the linear buckling solution sequence SOL 105 before making a run with SOL 106. The user has to introduce initial imperfections (loads or enforced displacements) to keep the structure in the intended buckling mode.

Iteration Control: NLPARM Entry

The most crucial data for successful nonlinear static solutions are contained in the NLPARM entry, which defines strategies for the incremental and iterative solution processes. It is difficult to choose the optimal combination of all the options for a specific problem. However, based on a considerable number of numerical experiments, the default option was intended to provide the best workable method for a general class of problems. Therefore, users with little insight or experience in a specific application should start with the default option. However, users with some experience and insight in a specific problem may change the default values based on the following observations:

- Computing cost for each line search is comparable to that of an iteration.
- The SEMI method usually provides better convergence than the AUTO method at the expense of higher computing cost.

- Default tolerances for the convergence criteria may be somewhat conservative. However, loose tolerances may cause difficulties in the subsequent steps.
- The quasi-Newton method is effective in most problems. However, it seems to have adverse effects in some problems, e.g., creep analysis.
- The line search method is effective to cope with difficulties in convergence in some problems. More extensive line searches may be exercised by a large value of MAXLS and/or a smaller value of LSTOL. On the other hand, line searches may have adverse effects in some problems, e.g., plane stress plasticity.
- The arc-length method should be used if the problem involves snap-through or postbuckling deformation. Then the Bulk Data entry NLPCI must be attached.

The NLPARM entry format is shown below with default values wherever applicable:

1	2	3	4	5	6	7	8	9	10
\$NLPARM	ID	NINC	DT	KMETHOD	KSTEP	MAXITER	CONV	INTOUT	
NLPARM			0.0	AUTO	5	25	PW	NO	
\$	EPSU	EPSP	EPSW	MAXDIV	MAXQN	MAXLS	FSTRESS	LSTOL	
	1.0E-3	1.0E-3	1.0E-7	3	MAXITER	4	0.2	0.5	
\$	MAXBIS				MAXR		RTOLB		
	5				20.		20.		

The ID field is referenced by the NLPARM Case Control command. The NINC field is an integer which specifies the number of increments to be processed in the subcase. The total load specified in the subcase minus the load specified in the preceding subcase is equally divided by this integer (NINC) to obtain the incremental load for the current subcase. Another subcase should be defined to change constraints or loading paths. However, multiple subcases may be required in the absence of any changes in constraints or loads to use a moderate value (e.g., not to exceed 20) for NINC. Use of a moderate value has advantages in controlling database size, output size, and restarts.

The DT field requires a real number specifying the time increment for each load step in the case of creep analysis. The time unit must be consistent with the unit used in the CREEP data to define the creep characteristics. The creep time increment should be determined properly for efficiency and accuracy based on the creep rate represented by the creep law to be used. The larger the creep rate, the smaller the time increment should be. No creep deformation is considered with the default value of 0. It is noted that a creep analysis should be preceded by a static analysis with at least one subcase, because the creep deformation requires stresses in the material.

Stiffness matrix update strategies are determined by a combination of the data specified in the two fields KMETHOD and KSTEP. Options for KMETHOD are AUTO, SEMI, or ITER. The KSTEP field, which is an auxiliary to the KMETHOD field, should have an integer equal to or greater than 1. With the AUTO option, the program automatically determines when to reevaluate the stiffness matrix based on the rate of convergence. At each iteration, the computing time for convergence without the stiffness matrix update is estimated and compared with the computing time for the matrix update in order to determine whether the update is more efficient. This decision is deferred in the first two iterations after a new stiffness is obtained. If the solution tends to diverge, however, the update decision will be made effective immediately. The stiffness matrix will be updated upon convergence if the number of iterations required for convergence is greater than KSTEP. The SEMI option is identical to the AUTO option except for one additional stiffness update after the first iteration which always occurs unless the solution converges in a single iteration. With the ITER option, the stiffness matrix is updated at every KSTEP iterations. Thus, the full Newton-Raphson iteration is exercised if KSTEP is 1. If $KSTEP > MAXITER$, the stiffness will never be updated.

The MAXITER field is an integer representing the number of iterations allowed for each load increment. If the number of iterations reaches MAXITER without convergence, the load increment is bisected and the analysis is repeated. If the load increment cannot be bisected (i.e., MAXBIS is reached or $MAXBIS = 0$) and MAXDIV is positive, the best attainable solution is computed and the analysis is continued to the next load increment. If MAXDIV is negative, the analysis is terminated.

The convergence test is performed at every iteration with the criteria specified in the CONV field. Any combination of U (for displacement), P (for load), and W (for work) may be specified. All the specified criteria must be satisfied to achieve convergence, except for an absolute convergence condition, under which the solution is converged regardless of criteria. This subject will be elaborated in the following section. The convergence tolerances are specified in the fields EPSU, EPSP, and EPSW for U, P, and W criteria, respectively.

The defaults for the nonlinear convergence tolerances have been modified to improve run times for most large nonlinear static analyses without appreciably sacrificing accuracy. The new default tolerances also reflect values similar to those used in other popular nonlinear FEA programs such as MSC.Marc. Run times of large nonlinear static models with plasticity and/or large deformation models will usually improve significantly. Run times for models with gaps or line contact may also improve, but not as significantly. At this time, it appears that static nonlinear heat transfer models and most nonlinear dynamic analysis models continue to require the present tight default tolerances.

Starting in MSC.Nastran 2001, default nonlinear tolerances in Solutions 106 and 153 will vary according to the model, analysis type, and user-defined precision. Although default tolerances can still be overridden by entering values of CONV, EPSU, EPSP, and EPSW on the NLPARM entry, default tolerances can now be easily invoked with a single parameter, leaving the above fields on the NLPARM entry blank. Depending on the desired level of accuracy and run times, various default tolerance sets may be selected. These tolerances are designated as “very high,” “high,” “engineering design,” and “preliminary design” and can now be controlled with the new parameter

PARAM,NLTOL,ITOL

where ITOL is an integer value ranging from 0 (very high) to 3 (preliminary design). The default is 2 (engineering accuracy) for nonlinear static analysis and 0 (very high) for nonlinear heat transfer and dynamic analyses. See “[Parameter Descriptions](#)” on page 602 of the *MSC.Nastran Quick Reference Guide*, for a complete description.

[Table 13-11](#), [Table 13-12](#), and [Table 13-13](#), illustrate the tolerances that will be set by the program if the CONV, EPSU, EPSP, and EPSW fields on the NLPARM entry are left blank. See Remark [16](#). of the NLPARM entry for further details.

Table 13-11 Default Tolerances for Static Nonlinear SOL 106 Models Without Gaps, Contact or Heat Transfer

NLTOL	Designation	CONV	EPSU	EPSP	EPSW
0	Very high	PW	_____	1.0E-3	1.0E-7
1	High	PW	_____	1.0E-2	1.0E-3
2	Engineering	PW	_____	1.0E-2	1.0E-2
3	Prelim Design	PW	_____	1.0E-1	1.0E-1
none	Engineering	PW	_____	1.0E-2	1.0E-2

Table 13-12 Default Tolerances for Static Nonlinear SOL 106 Models With Gaps or Contact (Enter NLTOL Values of 0 or 2 Only or Omit the Parameter)

NLTOL	Designation	CONV	EPSU	EPSP	EPSW
0	Very high	PW	_____	1.0E-3	1.0E-7
2	Engineering	PW	_____	1.0E-3	1.0E-5
None	Engineering	PW	_____	1.0E-3	1.0E-5

Table 13-13 Default Tolerances for Static Nonlinear SOL 106 or 153 Models With Heat Transfer (Enter NLTOL Value of 0 Only or Omit the Parameter)

NLTOL	Designation	CONV	EPSU	EPSP	EPSW
0	Very high	PW	_____	1.0E-3	1.0E-7
None	Very high	PW	_____	1.0E-3	1.0E-7

To use the new default tolerances, the CONV, EPSU, EPSP, and EPSW fields of the NLPARM entry must be left blank. In addition, the NINC field must also either be blank or have a value of 10 or larger. There are two reasons for this: (1) Most cases studied to date show that approximately 10 load steps or more are required for proper accuracy and convergence with the new tolerances, and (2) some users may have already “calibrated” a model as to the number of load steps necessary for good results using pre-version 2001 default tolerances. Therefore, if NINC is set to a value between 1 and 9, the pre-version 2001 tolerances (i.e. “very high” accuracy) will be used. It should be noted that although the *MSC.Nastran Quick Reference Guide* did not specify a default for NINC, the default prior to 2001 was actually 1. For MSC.Nastran 2001, the default value for NINC has been changed to 10.

If PARAM,NLTOL is not specified, then the default convergence tolerances are the same as if PARAM,NLTOL,2 had been entered except for heat transfer problems, for which the default is PARAM,NLTOL,0. The type of analysis will be detected automatically by MSC.Nastran and the default tolerances adjusted accordingly.

Dynamic problems using SOL 129 and 159 retain the same pre-version 2001 default tolerances as described in “**TSTEPNL**” on page 2139 of the .

The defaults on the NLPCI entry (arc length incremental solution strategy) have not been changed.

As in the past, if the user has a good idea of what to use for the default tolerances and/or number of increments, then these values should be used. If the problem is nearly linear, a value of NINC=1 sometimes yields acceptable results. For small problems, it is recommended that the pre-version 2001 defaults (very high tolerances) be used because, in these cases, run times are generally not significant.

Because nonlinear tolerances now depend on NLTOL, NINC (on the NLPARM entry), and other possible NLPARM overrides, convergence tolerances actually used by the code are now echoed in the output file (right after the Bulk Data echo, if any) as shown in the following example:

```

DEFAULT NONLINEAR TOLERANCES
  NLPARM ID      10
  NINC          10
  EPSU          1.00000E-03 (not normally used)
  EPSP          1.00000E-03
  EPSW          1.00000E-05
BASED ON
MODEL HAS GAPS AND/OR LINE CONTACT
(VALUES YOU ENTER OVERRIDE THE ABOVE DEFAULTS)

```

13.3.15 Examples of Convergence Tolerance Changes

A very large nonlinear model consisting of approximately 950,000 DOFs was studied extensively to determine the effect of various tolerances on run time and accuracy. This particular model did not employ heat transfer, gaps, or contact. Results of the study showed that compared to the present “very high accuracy” solution, acceptable accuracy can be achieved for both the “high accuracy” and “engineering accuracy” designations. The “preliminary design” accuracy tolerances produces results with the same general trends as other reasonable tolerances, but the answers were considered too inexact for analyses other than preliminary design tradeoff studies. The following table summarizes the effect of changing the tolerances for this large model.

Accuracy Type	Tolerances		Run time CPU hours	Accuracy for the large model
	EPSP	EPSW		
Very high*	0.001	1.0E-7	163.2	Very Good
High	0.01	0.001	10.3	5% or less accurate
Engineering Design	0.01	0.01	7.0	10% or less accurate
Preliminary Design	0.1	0.1	2.9	10% to 20% less accurate

*Same as pre-version 2001 default tolerances.

Many test cases--including standard MSC nonlinear QA test cases--were also analyzed using the new default tolerances. For all models run to date, satisfactory results were obtained. For those models that invoked the new “engineering design” tolerances (because NINC, CONV, EPSU, EPSP and EPSW were blank) results changed as expected. Typically, such changes were largest, on a percentage basis, for the smallest responses in the model. Maximum response values experienced little change. For example, if the largest displacement was 0.01, then displacements with values around 0.01 changed by about 1% or less, but displacements that were originally 0.0001 might change by 20% or even more. If the maximum stress was 20,000, then stresses around 20,000 would change by 1% or less, but stresses around 1000 might change by 20% or more. If the problem produced primarily displacements in the Y direction, then using the “engineering tolerances” gives larger cross-axis response in the X and Z directions than do the “very high” accuracy tolerances. Stresses behave in a similar manner. However, deformed contour plots of most models using the two accuracy values show little or no difference. If cross-axis responses or if stresses away from the most highly stressed regions are important, or if the nonlinear material properties of the model are known with great precision, the “very high” accuracy tolerances should be used. However for most analyses, the “engineering design” tolerances will often suffice.

13.3.16 Changes to the NLPARM Entry

Note the following changes to the NLPARM Bulk Data entry starting with MSC.Nastran 2001.

The following field comments have been revised:

- NINC Number of increments. (0 < Integer < 1000; Default = 10; See Remark 16)

- EPSU Error tolerance for displacement (U) criterion. (Real > 0.0; Default = 1.0E-2; See Remark 16)
- EPSP Error tolerance for load (P) criterion. (Real > 0.0; Usual default = 1.0E-2; See Remark 16)
- EPSW Error tolerance for work (W) criterion. (Real > 0.0; Usual default = 1.0E-2; See Remark 16)

A new Remark, 16, has been added:

16. Default tolerance sets are determined based on model type and desired accuracy. Accuracy is under user control and can be specified on the PARAM,NLTOL entry. NLTOL's value is used only if the CONV, EPSU, EPSP and EPSW fields are blank, and if NINC is blank or set to a value of 10 or larger. Otherwise, the NLTOL selection will be overridden. The tables below list tolerances according to NLTOL selections:

Default Tolerances for Static Nonlinear SOL 106 Models Without Gaps, Contact or Heat Transfer

NLTOL	Designation	CONV	EPSU	EPSP	EPSW
0	Very high	PW	_____	1.0E-3	1.0E-7
1	High	PW	_____	1.0E-2	1.0E-3
2	Engineering	PW	_____	1.0E-2	1.0E-2
3	Prelim Design	PW	_____	1.0E-1	1.0E-1
None	Engineering	PW	_____	1.0E-2	1.0E-2

Default Tolerances for Static Nonlinear SOL 106 Models With Gaps or Contact (Enter NLTOL Values of 0 or 2 Only or Omit the Parameter)

NLTOL	Designation	CONV	EPSU	EPSP	EPSW
0	Very high	PW	_____	1.0E-3	1.0E-7
2	Engineering	PW	_____	1.0E-3	1.0E-5
None	Engineering	PW	_____	1.0E-3	1.0E-5

Default Tolerances for Static Nonlinear SOL 106 or 153 Models With Heat Transfer (Enter NLTOL Value of 0 Only or Omit the Parameter)

NLTOL	Designation	CONV	EPSU	EPSP	EPSW
0	Very high	PW	_____	1.0E-3	1.0E-7
None	Very high	PW	_____	1.0E-3	1.0E-7

The INTOUT field requires a specification of YES, NO or ALL. If YES is specified, the output requests (DISPLACEMENT, FORCE, STRESS, etc.), specified in the Case Control Data, will be processed for every computed converged solution in the subcase. If NO is specified, the output requests will be processed only for the last load step of the subcase. If ALL is specified, the output requests are processed for every computed and user specified load increment. For Newton's iteration methods (i.e., without NLPCI), the option ALL is equivalent to the option YES, since the computed load increment is equal to the user specified load increment. For arc-length methods (i.e., NLPCI command is specified), the computed load increment in general is not the same as the user specified load increment, and is not known in advance. The option ALL allows the user to obtain solutions at the desired intermediate load increments. With a small database option, the database will retain only those data blocks corresponding to the incremental load steps for which output is requested by an INTOUT field. Then, the restarts will be restricted to those starting from these steps. However, if the job is terminated with an incomplete subcase for some reason (e.g., diverging, time expiration, etc.), data blocks for all the interim steps of that subcase are stored in the database to allow flexible restarts.

The MAXDIV field requires an integer to specify a limit on the probable divergence conditions allowed for each iteration to continue. There are two classifications in the divergence condition: probable and absolute. The absolute divergence is treated as two occurrences of the probable divergence. When the probable divergence occurs MAXDIV times or more and if MAXDIV is positive, the current solution base (displacements) is retracted and the stiffness matrix is updated in reference to the preceding iteration step. If the solution diverges again in the same load increment while MAXDIV is positive, the best attainable solution is computed and the analysis is continued to the next load increment. If MAXDIV is negative, the analysis is terminated on second divergence.

The MAXQN field requires an integer to specify the maximum number of quasi-Newton vectors to be saved in the database. Quasi-Newton vectors are accumulated, if desired, until MAXQN is reached. If the number of accumulated QN vectors is greater than or close to MAXQN upon convergence, the stiffness matrix will be updated before the next increment is processed. All the QN vectors already

accumulated are purged upon a stiffness update and the accumulation is reinitiated. The quasi-Newton update may be suppressed with a value of 0 for MAXQN. Details of the quasi-Newton method will be presented in the following section.

The line search is controlled by the fields MAXLS and LSTOL. The maximum number of line searches allowed for each iteration is specified in the MAXLS field. The line search process may be suppressed with a value of 0 for MAXLS. The LSTOL field requires a real number between 0.01 and 0.9 to specify a tolerance for the line search operation. The line search operation will be conducted if the error defining the divergence rate is greater than LSTOL. If the line search convergence is not achieved, the line search continues until the number of line searches reaches MAXLS.

The FSTRESS field requires a real number ($0.0 < \text{FSTRESS} < 1.0$), representing a fraction of the effective stress ($\bar{\sigma}$) used to limit the subincrement size in the material routines. The number of subincrements in the material routines for elasto-plastic or creep deformation processes is determined such that a subincrement in the effective stress is approximately $\text{FSTRESS} \cdot \bar{\sigma}$. This data is also used to establish an error tolerance in the yield function to alleviate accumulation and propagation of the error. If the error in the yield function exceeds $\text{FSTRESS} \cdot \bar{\sigma}$ at the converging state, the program will terminate the job with a fatal error message “ERROR EXCEEDED 20% OF CURRENT YIELD STRESS.” If the bisection option is selected, bisection will be activated under this condition.

The MAXBIS field requires an integer to specify the number of bisections allowed for a load or arc-length increment. When the solution diverges, different actions are taken depending on the sign of MAXBIS. If MAXBIS is positive, the stiffness matrix is updated on the first divergence and the load increment is bisected on the second divergence. If MAXBIS is negative, the load increment is bisected every time the solution diverges until the limit on bisection is reached. If the solution does not converge after $|\text{MAXBIS}|$ bisections, the analysis is continued or terminated depending on the sign of MAXDIV.

The MAXR field requires a real number to specify the bounds on the ratio of the adjusted arc-length increment relative to the initial value. In the adaptive load/arc-length increment, the overall upper and lower bounds on the load/arc-length increment in the subcase are defined as

$$\frac{1}{\text{MAXR}} \leq \frac{\Delta l_n}{\Delta l_0} \leq \text{MAXR}$$

where Δl_n is the arc-length at step n and Δl_0 is the original arc-length. The arc-length method for load increments is selected by the NLPCI Bulk Data entry, which must have the same ID as the NLPARM Bulk Data entry.

The RTOLB field requires a real number to specify the incremental rotation (in degrees) allowed per iteration. The bisection is activated if the incremental rotation for any degree-of-freedom ($\Delta\theta_x$, $\Delta\theta_y$ or $\Delta\theta_z$) exceeds the value specified for RTOLB. This bisection strategy based on the incremental rotation is controlled by the MAXBIS field.

User Interface for Arc-Length Methods: NLPCI Entry

Most of the parameters in the NLPARM Bulk Data entry are used to control the iteration strategy for arc-length methods. However, MAXLS is not applicable because the line search procedure is not coupled with arc-length methods currently. The parameters which are applicable only to the arc-length methods are specified in the Bulk Data entry NLPCI, in connection with the NLPARM data by the same ID. The NLPCI entry is shown below with default values, followed by brief description of the fields:

1	2	3	4	5	6	7	8	9	10
NLPCI	ID	TYPE	MINALR	MAXALR	SCALE		DESITER	MXINC	
NLPCI		CRIS	0.25	4.	0.0		12		

ID Must be identical to ID of NLPARM to be active.

TYPE Constraint type (CRIS for Crisfield, RIKS for Riks, or MRIKS for modified Riks method)

MINALR, MAXALR For variable arc-length method, the arc-length is bounded by

$$\text{MINALR} \leq \frac{\Delta l_{new}}{\Delta l_{old}} \leq \text{MAXALR}$$

where $\text{MINALR} \leq 1$. and $\text{MAXALR} \geq 1$. The automatic adjustment is not activated if $\text{MAXALR} = \text{MINALR} = 1$.

SCALE Scaling factor for the magnitude of the load term in arc length. Real > 0 Default = 0. The constraint equation has a disparity in the dimension by mixing the displacements with the load factor. The scaling factor (w) is introduced as user input so that the user can make constraint equation unit-independent by a proper scaling of the load factor, μ . As the value of w is increased, the constraint equation is gradually dominated by the load term. In the limiting case of infinite w , the arc-length method would be degenerated to the conventional Newton's method.

DESITER Desired number of iterations for convergence to be used for adaptive arc-length adjustment.

$$\Delta l_{new} = \Delta l_{old} \sqrt{\frac{I_d}{I_{max}}}$$

MXINC Maximum number of controlled increments allowed within the specified load step. There can be cases where the structure never snaps back and collapses to failure. In these cases, the analysis cannot be completed without bounding the maximum number of increments. For such cases, this value should be confined to a reasonable limit, e.g., less than 100, to prevent inadvertent depletion of the computing resources.

Iteration Related Output Data

At the end of every iteration, the relevant data from the iteration process are printed under the following heading. The section numbers in parentheses refer to sections in the *MSC.Nastran Handbook for Nonlinear Analysis*.

ITERATION	iteration count <i>i</i> .
EUI	relative error in terms of displacements (see Section 3.6.3).
EPI	relative error in terms of loads (see Section 3.6.3).
EWI	relative error in terms of work (see Section 3.6.3).
LAMBDA	rate of convergence (defined in Section 3.3.1).
DLMAG	absolute norm of the load error vector.
FACTOR	final value of the line search parameter (see Section 3.4).
E-FIRST	divergence rate, initial error before line search (see Section 3.4.3).
E-FINAL	error at the end of line search (see Section 3.4.3).
N-QNV	number of quasi-Newton vectors appended (see Section 3.5.3).
N-LS	number of line searches performed during the iteration.
ENIC	expected number of iterations for convergence (see Section 3.3.1).

NDV	number of occurrences of probable divergence during the iteration (see Section 3.3.2).
MDV	number of occurrences of bisection conditions in the material routine (excessive stress increment) or in the rotation angle (excessive rotation) during the iteration.

Loads Overview

The static loads in nonlinear analysis consist of a subset of the total MSC.Nastran static load set. Most of the relevant loads data applicable to the linear static analysis are also applicable to nonlinear static analysis except for the Bulk Data entry DEFORM. The enforced displacements can be applied as loads using the Bulk Data entry SPCD, which defines an enforced motion of a grid in a specific fixed direction. Each grid point with an enforced displacement must also appear on an SPC or SPC1 entry.

The loads are selected in each subcase by a Case Control command LOAD. The load for a subcase is subdivided into the number of increments specified for the subcase. The solution strategy in nonlinear is to apply the loads in an incremental fashion until the desired load level is reached. The algorithms “remember” the loads from one subcase to the next. If the load reaches the desired level in a subcase and if the load description is left out in the subsequent subcase inadvertently, the nonlinear solution algorithm will begin in an incremental fashion to remove the load. The loads described below make up the valid subset applicable for nonlinear static analysis.

Concentrated Loads

The concentrated loads consist of two categories: those which are stationary in direction through out the analysis and those which follow the grid motion. The first category consists of the following: FORCE, MOMENT, and SLOAD.

The second category consists of the following: FORCE1, FORCE2, MOMENT1, and MOMENT2. The three FORCE entries differ only in the manner in which the direction of the force is specified. FORCE uses the components of a vector. FORCE1 uses two grid points, not necessarily the same as the loaded grid points. FORCE2 defines the direction of the force as a vector which is the vector product of two other vectors. The distinctions between the three MOMENT entries are similar. These loads are follower loads in nonlinear and produce follower stiffness terms.

Distributed Loads

The distributed loads consist of two categories: those which are stationary in direction through out the analysis and those which follow the element motion. The first category consists of the PLOAD1.

The second category consists of the following: PLOAD, PLOAD2, and PLOAD4.

Loads may be applied to the interiors of finite elements. The PLOAD1 entry defines concentrated and linearly distributed forces and moments to the CBEAM element. However, the stress calculation for nonlinear CBEAM element ignores the distributed loads, but lumps them onto the nodal points. The PLOAD entry is exceptional in that it references three or four arbitrarily located grids rather than an element. The PLOAD2 entry defines a uniform static pressure load applied to the CQUAD4 or CTRIA3 elements. The PLOAD4 defines a pressure load on surfaces of CHEXA, CPENTA, CTRIA3, and CQUAD4 elements which need not be normal to the surface. When normal to the surface, the PLOAD4 is a follower load.

Mass Related Loads

The mass related loads are gravity and centrifugal loads, which require specification of mass data. The gravity loads which can be specified in the GRAV Bulk Data entry are stationary in direction through out the analysis. It is used to define the direction and magnitude of the gravity vector in some identified coordinate system. The components of the gravity vector are multiplied by the mass matrix to obtain the components of the gravity force at each grid point.

The centrifugal loads to be specified in the RFORCE Bulk Data entry is used to define the components of a spin vector which is used internally to compute centrifugal forces. Each component of the spin vector is multiplied by a scale factor.

Thermal Loads

Thermal loads are selected in each subcase by the Case Control command TEMP(LOAD). Thermal loading which is a follower loading is accomplished through the following entries: TEMP, TEMPD, TEMPP1, TEMPP3, and TEMPRB. The TEMP and TEMPD provide for grid point temperature specification which are then interpolated to points within elements. For the nonlinear CQUAD4 and CTRIA3 elements the TEMPP1 provides for an average temperature and thermal gradient in the thickness direction and the TEMPP3 provides for temperature at points through the thickness. The TEMPRB provides for average temperature at ends A and B for the CROD and CBEAM element. Average temperature specified directly for an element will take precedence over the temperature interpolated from the elements connected grid points.

The Case Control TEMP(INIT) is required above the subcase level. For nonlinear static analysis TEMP(MAT) or TEMP(BOTH) are not applicable. The use of TEMP(MAT) or TEMP(BOTH) will cause a fatal error. Subcases that do not contain a TEMP(LOAD) will default to TEMP(INIT), which implies no thermal load. Thermal loads are often used to provide initial strains.

Loads Combination

Various load specifications can be combined using LOAD and (CLOAD, LSEQ).

The LOAD provides for the linear combination of static load sets. The static loads (LOAD, FORCE, etc.) applied to the upstream superelements cannot be referenced by a Case Control command LOAD. The CLOAD entry is designed to apply static loads to upstream superelements by combining loads defined in LSEQ entries. The (CLOAD, LSEQ) combination allows the nonlinear algorithms to apply in an incremental fashion on the upstream superelement loads to the boundary of the residual structure.

The CLOAD defines a static load combination for superelement loads acting on residual boundary and the LSEQ defines a static load combination for superelement upstream loads. The LSEQ assigns load vectors to the superelements and numerically labels them via the DAREA field entry value. The LID and TID field entries point to the appropriate load and temperature distribution entries. The CLOAD defines a static load as a linear combination of previously calculated superelement loads defined by the LSEQ. The IDVi fields correspond directly to the numeric labels defined via the DAREA fields of the LSEQ entries.

The CLOAD must be selected in the residual solution subcase via the case control CLOAD entry. Only one LOADSET may appear in Case Control. The hierarchy of the loads data is shown schematically in [Figure 13-27](#). An example Case Control Section is shown below:

```

      :
SEALL=ALL $ Required in SOL 66 only

LOADSET = 1000 $ Selects LSEQ 1000 for upstream loads
SUPER = ALL $ Identify SEs to process

      :
      DISP = ALL
      :
      :
      :
      :
$ NONLINEAR SOLUTIONS FOR RESIDUAL SUPERELEMENTS
SUBCASE 10
      CLOAD = 1001 $ Refers to CLOAD Bulk Data
      NLPARM = 12 $ Iteration control
      :

```

```

SUBCASE 20
  CLOAD = 1002
  LOAD = 10 $ Residual SE load
  NLPARM = 22
  :

```

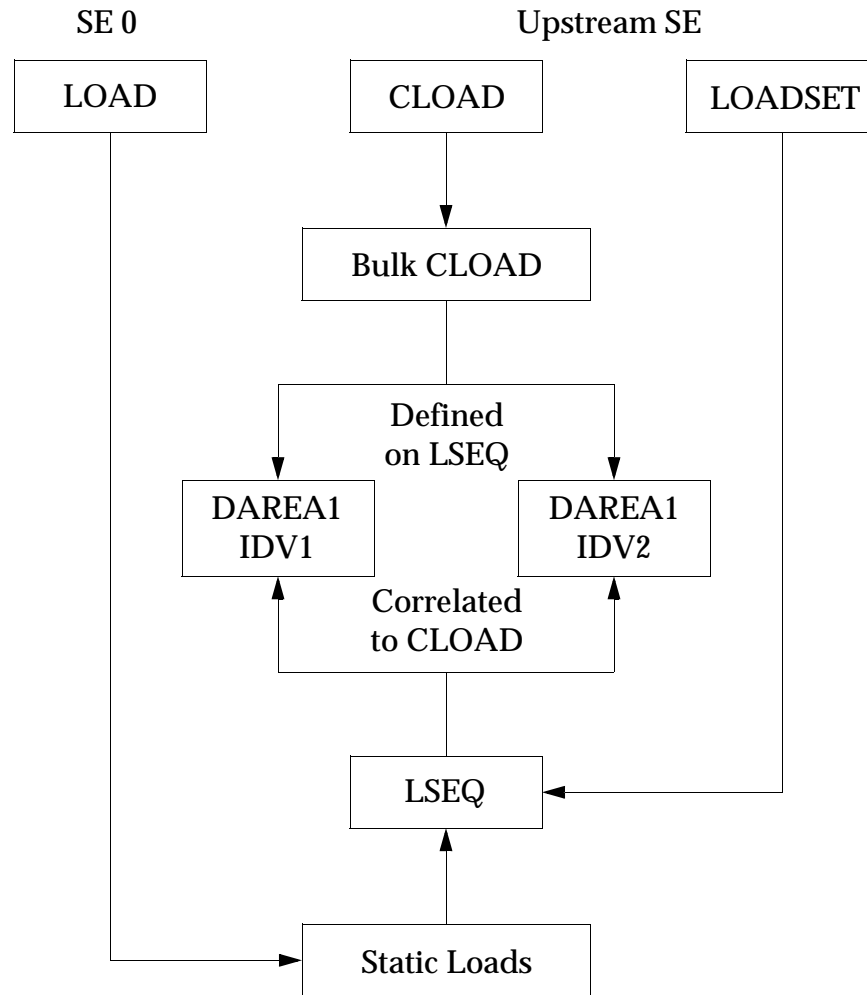


Figure 13-27 Hierarchy of Load Data

Restarts for Nonlinear Static Analysis

Any restart in SOL 106 must have the FMS command **RESTART** and the two parameters **PARAM,SUBID,value** and **PARAM,LOOPID,value**.

The **SUBID** is the sequential number of a subcase. It is recommended that the **SUBID** value always be incremented by one and point to a new subcase. The new Subcase should contain any standard subcase commands required.

The LOOPID is the identifier of the converged solution from which the restart is to take place.

Any converged and saved solution always gives the messages:

```
*** USER      INFORMATION MESSAGE 6186,  
    *** SOLUTION HAS CONVERGED ***  
  
    SUBID 7  LOADINC 5  LOOPID 67  LOAD STEP  7.000  LOAD FACTOR  1.00000  
  
^^^ DMAP INFORMATION MESSAGE 9005 (NLSTATIC) - THE SOLUTION FOR LOOPID= 67  
    IS SAVED FOR RESTART
```

To restart from the above run, the restart might look like:

```

RESTART
TIME 10000
SOL 106
DIAG 50
CEND
TITLE = SAMPLE RESTART
PARAM,SUBID,8
PARAM,LOOPID,67
SPC = 200
MPC = 250
SET 1 = 1118
DISPLACEMENT = 1
SPCF=1
SUBCASE 1
LOAD = 101
NLPARM = 51
.
.
.
SUBCASE 7
LOAD = 107
NLPARM = 57
SUBCASE 8
LOAD = 108
NLPARM = 58
BEGIN BULK
NLPARM  58      5      AUTO    20      20      PW      YES
          1-3    1-3    1-7     -3
SPCD    108    1118    3      0.025
FORCE  108    1118      0.      0.      0.      1.
ENDDATA

```

For a full data recovery restart after the completion of a complete analysis proceed as above but do not actually add the new Subcase in Case Control. For example, assume the analysis has ended in subcase 8 above with the following messages:

```

*** USER      INFORMATION MESSAGE 6186,
*** SOLUTION HAS CONVERGED ***

      SUBID 8  LOADINC 5  LOOPID 72  LOAD STEP  8.000  LOAD FACTOR  1.00000

^^^ DMAP INFORMATION MESSAGE 9005 (NLSTATIC) -  THE SOLUTION FOR LOOPID= 72
    IS SAVED FOR RESTART

```

Perform a full data recovery restart as follows:

```

RESTART
TIME 10000
SOL 106
DIAG 50
CEND
TITLE = FULL DATA RECOVERY RESTART
MAXLINES = 10000000
$ POINT TO A NEW SUBCASE BUT DO NOT ACTUALLY ADD A NEW SUBCASE
PARAM,SUBID,9
PARAM,LOOPID,72

```



```

SPC = 200
MPC = 250
SET 1 = 1118
$DISPLACEMENT = 1
$ ADD DISPLACEMENT AND STRESS REQUESTS FOR DATA RECOVERY
DISPLACEMENT=ALL
STRESS=ALL
SPCF=1
SUBCASE 1
LOAD = 101
NLPARM = 51
.
.
.
SUBCASE 7
LOAD = 107
NLPARM = 57
SUBCASE 8
LOAD = 108
NLPARM = 58
BEGIN BULK
ENDDATA

```

Note that in the above, there is no SUBCASE 9 added.

The result of this last restart will be the complete displacement and stress results output for each converged solution saved on the data base starting with the first converged solution and ending with the results of the converged solution at LOOPID = 72.

13.3.17 Nonlinear Buckling Analysis

A simple approach to nonlinear buckling analysis was introduced in SOL 106 (or 66) by using a quadratic extrapolation. A similar approach to the limit point calculations was published by Fujikake. The distinction is that MSC.Nastran uses two converged solutions in the vicinity of the buckling point to form a differential stiffness $[\Delta K]$, whereas Fujikake uses a converged point and an estimated data point beyond the buckling point. To find an instability point within a small range of nonlinear domain, two methods of idealization can be contemplated:

1. The tangent stiffness matrix is proportional to the external loads, which implies that the critical load may be linearly interpolated, i.e., $P_{cr} = P_n + \alpha \Delta P$.
2. The tangent stiffness matrix is proportional to the displacement increments, which implies that the critical displacements may be obtained by extrapolating from the current state, i.e., $U_{cr} = U_n + \lambda \cdot \Delta U$.

It appears that Fujikake has chosen the first approach. However, he did not show how he obtains the critical displacements. MSC.Nastran has chosen the second approach. Since the tangent matrix is assumed to change linearly, the internal loads are quadratic function of displacements. The distinction between the two approaches disappears in the linear bifurcation problem.

Restart is required for a nonlinear buckling analysis after the non-positive definite stiffness matrix is detected. Since the modified Newton's method is employed with quasi-Newton updates in the general purpose program, tangential stiffness matrices in the two successive increments are not immediately available in general. Details of the method are given below.

Activate the user-specified parameter BUCKLE with a value of +1. The operational steps are then:

1. Run SOL 106 for static analysis and create a database for restarts. It is recommended to continue the nonlinear static analysis with small steps until a negative determinant of the stiffness matrix is encountered.
2. Make a restart run as follows:
 - a. Select PARAM entries SUBID and LOOPID for restart.
 - b. Provide two small loading steps (below buckling). This may be provided by adding a new subcase, skipping the rest of the increments of the subcase in which the stiffness matrix becomes singular.
 - c. Keep "KSTEP = 1" in the NLPRAM entry for two load steps to be solved so that the stiffness matrix is updated for each solution.
 - d. Include a EIGB Bulk Data entry with a METHOD command in the Case Control Section for the eigenvalue analysis. The SINV method is recommended for the eigenvalue extraction.
 - e. Provide mode shape PLOT data if desired.
 - f. Include PARAM,BUCKLE,1.

In order to provide a more convenient and versatile capability for nonlinear buckling analysis, an option for a buckling analysis using a cold-start run with PARAM,BUCKLE,2 was introduced in MSC.Nastran Version 70.7. This option allows buckling analysis in any Subcase that has a METHOD command specified for an eigenvalue analysis. A few errors in the BUCKLE=2 option have been detected in Version 70.7 and corrected in MSC.Nastran 2001.

Example (NLSOLIDB.DAT)

The following input file shows how a nonlinear buckling analysis can be performed in a cold start run:

```

ID NLSOLIDB,V2001 $ SHL 4/27/00
SOL 106 $
DIAG 8 ,50 $
TIME 60 $ FOR VAX
CEND
TITLE=ELASTIC-PLASTIC BUCKLING OF IMPERFECT SPHERICAL SHELL
SUBTITLE=HYDROSTATIC PRESSURE APPLIED,PERIPHERY CLAMPED
SET 10 =100,200,109,209,119,219,131,231
SET 20 =10,11,15,20,25
SET 30 = 1000
ECHO=UNSORT
DISP=10
OLOAD=30
STRESS(PLOT)=20
SPC=10
$ buckling analysis in cold-start
PARAM,BUCKLE,2
$
SUBCASE 1
LOAD=10
NLPARM=10
SUBCASE 2
LOAD=20
NLPARM=20
METHOD=30 $ for buckling analysis
SUBCASE 3
LOAD=30
NLPARM=30
METHOD=30 $ for buckling analysis
BEGIN BULK
$ FOR BUCKLING ANALYSIS
EIGB 30 SINV 0.0 2. 20 2 2
$ PARAMETERS
PARAM POST 0
PARAM LGDISP 1
NLPARM 10 2 AUTO 7 30 yes
NLPARM 20 5 AUTO 1 30 yes
NLPARM 30 2 AUTO1 yes
$ LOADING
LOAD 10 1. 2. 1 2. 2
LOAD 20 1. 3. 1 3. 2
LOAD 30 1. 3.5 1 3.5 2
$ unit pressure is 1000 psi
PLOAD4 1 10 1000. 100
PLOAD4 1 11 1000. 101 104
...
$ a dummy force applied to a fixed point to output critical buckling load
FORCE 2 1000 1000. 0. 0. 1.
GRID 1000 0. 0. 0. 123456
...
...
ENDDATA

```

The buckling analysis is requested in Subcases 2 and 3 in the example. Notice that the KSTEP field on each of the NLPARAM entries for both Subcases 2 and 3 are set to 1 to force a stiffness matrix update at every solution step. The PARAM,BUCKLE,2 is placed above Subcase 1, but it could be placed in the Bulk Data Section or under the desired subcases. The buckling solutions in Subcases 2 and 3 will be different because the nonlinear buckling solution is an approximate solution based on extrapolation of two consecutive incremental solutions. The buckling solution tends to be more accurate when these two consecutive incremental steps are closer to the buckling point. However, the user does not know where the buckling point is until the analysis is performed. For this reason, the user may want to divide the load into a number of subcases and perform buckling analysis in multiple subcases.

In the previous example, pressure loads of 2000 and 3000 psi are applied in the first and the second subcase, respectively. The second subcase has five increments with an incremental pressure of 200 psi.

At the end of the second subcase, an eigenvalue analysis is performed due to PARAM,BUCKLE,2, which resulted in $\alpha = 0.4681$. Then the critical buckling load is calculated as

$$P_{cr} = P_n + \alpha\Delta P = 3000 + 0.4681 \cdot 200 = 3093.62$$

The third subcase applies a pressure load of 3500 psi in two increments, followed by a buckling analysis.

The buckling analysis resulted in $\alpha = 0.1135$, and the buckling load from this second eigenvalue analysis yields

$$P_{cr} = P_n + \alpha\Delta P = 3500 + 0.1135 \cdot 250 = 3528.38$$

If this buckling analysis is repeated by a restart run into subcase 3 using PARAM,BUCKLE,1, the eigenvalue analysis should give an identical solution within round-off error.

Discussion

Let us return to the example list of the input data for a review. Notice the OLOAD request for grid point 1000 that is fixed in space. However, a force of 1000 is applied to this dummy node with the same scaling in the LOAD Bulk Data as the pressure load. This OLOAD request prints the applied load for GRID 1000 at the eigenvalue solution step under the heading LOAD VECTOR. This applied load at buckling is identical to the critical buckling load calculated above. One can use this value as the critical buckling load instead of performing the above calculation.

There are two more ways to estimate the critical buckling load. One method is the arclength method that can provide solutions past the critical buckling load into the post-buckling state. Using the arclength method, the nodal displacement of a point with the most noticeable motion should be traced to detect the peak load. The applied load at this peak resembles the critical buckling load. Another method is to use the Newton's method until the solution cannot be obtained due to divergence, in which case adaptive bisection method is activated in the vicinity of the critical buckling load and stops at the limit load close to the critical buckling load. When the example problem was analyzed using an arclength method (CRIS), the solution went through the peak load of 3559.2 psi. When the same problem was analyzed using the Newton's method with bisection procedure, the solution traced up to the maximum load of 3540.63 psi before diverging due to instability. When the model was remeshed with about 100 times more elements, the buckling analysis procedure predicted a critical buckling load at 3545.43 psi. In this refined model, Newton's method with bisection converged up to 3546.88 psi and the arclength method peaked at 3555.72 psi.

Concluding Remarks

The following conclusion can be drawn from the preceding example. More refined models generally result in more accurate solutions as the eigenvalue analysis predicts a more accurate buckling point as the two solution points for extrapolation approach the actual buckling point. Comparison of the results confirms that the limit load obtained from Newton's method with adaptive bisection process closely predicts the critical buckling load. The arclength method tends to predict a slightly higher value for the buckling load.

13.3.18 Nonlinear Modal Analysis

Introduction

Prestressed normal mode in nonlinear static analysis (SOL 106) has been improved in MSC.Nastran 2001. All known errors related to this subject--including errors 38614 and 38682--have been corrected. The DMAP alter (s38614.alt) can be used to avoid the above errors in Version 70.7. Many of these features have been available since Version 70.7, but were not well documented.

User Interface

Prestressed normal modes can be requested at the end of each subcase in SOL 106 by adding the following items to the input file.

- Add METHOD=SID in the subcase of interest. A prestressed normal mode analysis will then be performed at the end of this subcase. The METHOD command points to an EIGRL or EIGR Bulk Data entry which then selects an eigenvalue method.
- For multiple normal mode analyses in SOL 106, the METHOD command may appear in more than one subcase or above all subcases. In the latter case, a normal mode analysis is performed at the end of each subcase. The stiffness used for modal analysis corresponds to the last step of the subcase. Modal analysis cannot be performed at intermediate solution step.
- A PARAM,NMLOOP,loopid command must appear in the Case Control or Bulk Data Section to request a normal mode analysis at the end of those subcases with a METHOD command. The actual value of loopid is unimportant as long as it is a positive integer. This alleviates the cumbersome task of figuring out the exact loopid.
- Add the appropriate EIGRL or EIGR entry in the Bulk Data Section of the *MSC.Nastran Quick Reference Guide*.

Example (TURBINE1.DAT)

The following example is a normal mode analysis of a prestressed turbine blade. (See [Figure 13-28](#)) The blade is modeled with CHEXA elements and loaded with centrifugal forces (RFORCE). The default iteration strategy is selected on the NLPARM Bulk Data entry. MSC.Nastran 2001 automatically updates the stiffness matrix at the end of the subcase if a METHOD Case Control command appears in the subcase. In Version 70.7, it was necessary to use the s38614.alt alter to get the same results as in MSC.Nastran 2001.

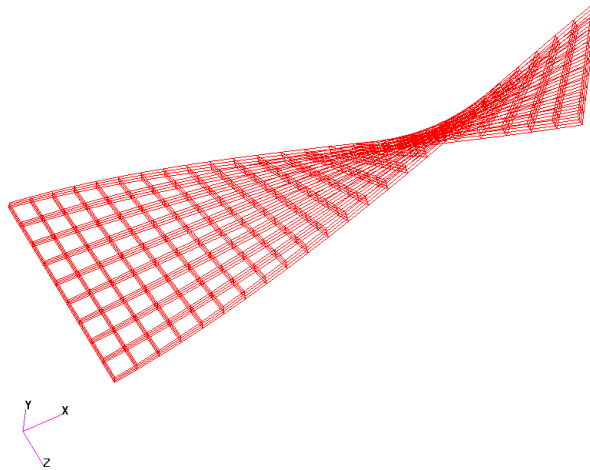


Figure 13-28 Prestressed Modes of a Turbine Blade

An excerpt of the input file is shown below.

```

SOL      106
DIAG     8
CEND
TITLE = Turbine Blade
  ECHO = SORT,PUNCH
  SPC  = 200
  LOAD      = 300
  NLPAR     = 400
  METHO     = 500
  DISP(PLOT) = ALL
  ELST(PLOT) = ALL
  NLSTRESS  = NONE
BEGIN BULK
PSOLID   100      100      100
MAT1     100      1.+7      .3      1.
RFORCE   300      .5
NLPARM   400      1          200      1.      2          +N
+N       1.-6      1.-12
+B
EIGRL    500      6
PARAM    COUPMASS1
PARAM    LGDISP      1
PARAM    NMLOOP      1
PARAM    POST      0
$
CHEXA    1        100      10000      10001      10101      10100      11000      11001      +M00000
+M00000  11101      11100
GRID     19324      44.      4.5      -.2      456
SPC1     200      123      10000      10100      10200      10300      11000      11100      +M00648
+M00648  11200      11300      12000      12100      12200      12300      13000      13100      +M00649
+M00649  13200      13300      14000      14100      14200      14300      15000      15100      +M00650
+M00650  15200      15300      16000      16100      16200      16300      17000      17100      +M00651
+M00651  17200      17300      18000      18100      18200      18300      19000      19100      +M00652
+M00652  19200      19300
$
$ PLUS THE REST OF THE MODEL
$
ENDDATA

```

Restart from SOL 106 into SOL 103 or into Another Linear Solution Sequence

For the purpose of a prestressed normal modes analysis, the old way of restarting from SOL 106 into SOL 103 is no longer necessary; the user can, instead, restart from a SOL 106 run into another SOL 106 run to perform the prestressed normal modes calculation.

Restarts from SOL 106 into linear solution sequences are not recommended to the novice user because of several limitations. The results of the linear restart are incremental values with respect to the preload, not total values. However, some experienced users restart from SOL 106 into SOL 103 to perform prestressed modal analysis with changing boundary conditions, or restart into another linear solution sequence to perform a perturbed linear solution on a preloaded structure. In the SOL 106 cold start run, KMETHOD='ITER' or 'AUTO' and KSTEP=1 must be specified on the NLPARM Bulk Data entry in order to restart from the end of the subcase. With these parameters, all nonlinear information is stored on the database so that the nonlinear tangent stiffness can be recovered in the linear restart run. The following command must be included in the linear restart run:

- PARAM,NMLOOP,loopid in the Case Control or Bulk Data Section, where loopid is the load step id at the end of a subcase from the SOL 106 cold start run.

Restarts are also allowed at intermediate steps within a subcase. The parameter INTOUT on the NLPARM entry must be set to 'YES' or 'ALL' in the SOL 106 cold start run to store intermediate steps on the database for restart purposes. However, the follower force stiffness may not be updated at intermediate steps. Therefore, the user should restart from the end of a subcase if the effect of follower force stiffness is important.

13.3.19 Differential Stiffness and Follower Force Stiffness in Linear and Nonlinear Solution Sequences

For prestressed normal mode analysis, it is useful to know which stiffness contributions are included. The table below summarizes the action of the parameters LGDISP and FOLLOWK in SOL 106 as compared to the action of FOLLOWK in linear solution sequences. In some cases, the default settings for linear and nonlinear solution sequences are not the same. In nonlinear static analysis (SOL 106), for the default of LGDISP=-1 (geometric linear), the parameter FOLLOWK (follower force stiffness) is ignored. For LGDISP=1, FOLLOWK is set to YES by default. In the linear solution sequences, FOLLOWK=YES is the default, see Case No. 3 in [Table 13-14](#).

In theory, the follower force stiffness is unsymmetric. The linear solution sequences always symmetrize the follower force stiffness. In Version 70.7, SOL 106 used the unsymmetric follower force stiffness and performance problems have been reported. Starting in MSC.Nastran 2001, by default, SOL 106 uses the symmetrized follower force stiffness to improve performance. If the user wants to use the unsymmetric follower force stiffness, the parameter FKSVMFAC can be specified for this purpose. See “**FKSYMFACT**” on page 645 of the *MSC.Nastran Quick Reference Guide* for further details. The use of the symmetrized follower force stiffness is recommended because it gives sufficient accurate results with substantial performance improvement.

In SOL 129, follower forces are included, however, the follower force stiffness is never calculated.

Table 13-14 Differential Stiffness and Follower Force Stiffness Contribution

Parameters		SOL 106 Cold Start or Restart		Restart from SOL 106 into SOL 103	SOL 103 Cold Start Using STATSUB	
LGDISP	FOLLOWK	Nonlinear Element Force	Stiffness	Stiffness	Stiffness	
1	1	YES	L+M+D+F	L+M+D+F	L+M+D+F	L+D'+F'
2	1	NO	L+M+D+F	L+M+D	L+M+D	L+D'
3	-1	YES	L+M	L+M	L+M+D'+F'	L+D'+F'
4	-1	NO	L+M	L+M	L+M+D'	L+D'
5	2	YES	L+M+D	L+M+D+F	L+M+D+F	L+D'+F'
6	2	NO	L+M+D	L+M+D	L+M+D	L+D'

where:

L = linear part of element force or stiffness

M = material nonlinear part of element force or stiffness

D = differential or geometric nonlinear part of element force or stiffness

D' = differential part of stiffness for small deformations

F = follower force part of element force or stiffness

F' = follower force part of stiffness for small deformations

All other linear solution sequences (e.g., SOL 105, 108, etc.) have the same default as SOL 103.

13.3.20 Buckling Analysis Enhancement

Introduction

An accurate solution to a buckling problem requires more meticulous efforts than just following a numerical procedure. There are a number of factors to consider before a buckling solution can be accepted with confidence. Such points along with a few examples are discussed by comparing solutions obtained from linear buckling analysis with preloads and nonlinear buckling with/without unsymmetric stiffness matrix.

Linear Buckling Analysis with Preloads

A starting step for a nonlinear buckling analysis should be a linear buckling analysis. The linear buckling analysis capability in SOL 105 used two subcases until Version 70.5: the first subcase for a linear static analysis with a buckling load, followed by the second subcase for an eigenvalue analysis. However, there may be other static loads sustained by the structure, in addition to the buckling load in question. In order to provide more versatile capability for general buckling cases, a new capability was introduced in MSC.Nastran Version 70.7, allowing constant preloads in the linear buckling analysis.

Theory

Let us start with a linear equation of motion for a preloaded structure, i.e.,

$$M\ddot{u} + C\dot{u} + Ku + K^d u = P(t) \quad \text{Eq. 13-13}$$

where M , C , K and K^d are mass, viscous damping, material and differential stiffness matrices, and $P(t)$ is a forcing function in time domain. The differential stiffness is created by the initial stress due to preloads, and it may include the follower stiffness if applicable. Ignoring the damping term to avoid complex arithmetic, an eigenvalue problem may be formulated as

$$[(K + K^d) - \omega^2 M]\{\phi\} = \{0\} \quad \text{Eq. 13-14}$$

which is a governing equation for normal mode analysis with a preload, but it could be recast for a dynamic buckling analysis at a constant frequency. For a static buckling, that constant frequency of vibration is zero and the inertia term drops out, i.e.,

$$[K + K^d]\{\phi\} = \{0\} \quad \text{Eq. 13-15}$$

in which $\{\phi\}$ represents virtual displacements, or buckling modes shapes. A non-trivial solution exists for an eigenvalue that makes the determinant of $[K + K^d]$ vanish, which leads to an eigenvalue problem

$$[K + \lambda K^d]\{\phi\} = \{0\} \quad \text{Eq. 13-16}$$

where λ is an eigenvalue which is a multiplier to the applied load to attain a critical buckling load. If there exists constant preloads other than the buckling load in question, the above equation should include additional differential stiffness, i.e.,

$$[K + K_{preload}^d + \lambda K_{buckle}^d]\{\phi\} = \{0\} \quad \text{Eq. 13-17}$$

in which differential stiffness is distinguished for constant preload and variable buckling load. Notice that no eigenvalue solutions are meaningful if the preload makes the structure buckle, i.e., $[K + K_{preload}^d]$ should be positive definite. The buckling analysis with an excessive preload can render wrong solutions, unless the positive definiteness is ensured.

Example - prld_buck.dat

An example of input data for a linear buckling analysis with a preload is shown below:

```
ID plate105, v2001 $   SHL 2/23/01
DIAG 8, 15 $
SOL 105
TIME 60
CEND
TITLE = MSC.Nastran job created on 20-Feb-01
      SPC = 2   $ clamp the bottom edge
      DISP=ALL
SUBCASE 10
      SUBTITLE= preload tensile edge force 300 lb/in
      LOAD = 2
SUBCASE 20
      Subtitle= top pressure 100 lb/in.
      LOAD = 7
SUBCASE 30
      SUBTITLE=buckling analysis
      STATSUB(preload)=10
      STATSUB(buckle)=20
      METHOD=1
BEGIN BULK
PARAM   POST      0
EIGRL  1          0.      6.      3      0          1.
```

ENDDATA

This is an example of a plate model (5 in. by 10 in., 0.1 in thick, steel, clamped on the bottom edge) with lateral tension by an edge load (in x) of 300 lb/in for a preload and vertical edge load (in y) of 100 lb/in for a buckling load. Alternatively, the preload and the buckling load may be switched to perform a similar analysis. The first buckling mode is a bending mode at vertical load of -64.55 lb/in, as shown in **Figure 13-29**. The second buckling mode in bending at -593 lb/in is also shown in **Figure 13-29**, but it is of no physical significance and shown here for verification purposes only. If the lateral load is in compression instead of tension, the plate could buckle in a twist mode at -123.29 lb/in in the absence of a vertical load, as shown in **Figure 13-30**. A parametric study, conducted with various combinations of lateral and vertical loads, resulted in a buckling envelope (or locus) on a loading plane as shown in **Figure 13-31**. The buckling shape is a bending mode above the curve and a twist mode in the left side of the curve. Bifurcation behavior can be observed in the transitional region near the corner point, where either the twist or bending mode could appear as the first and the second mode. Lower-right side of the curve is the safety zone.

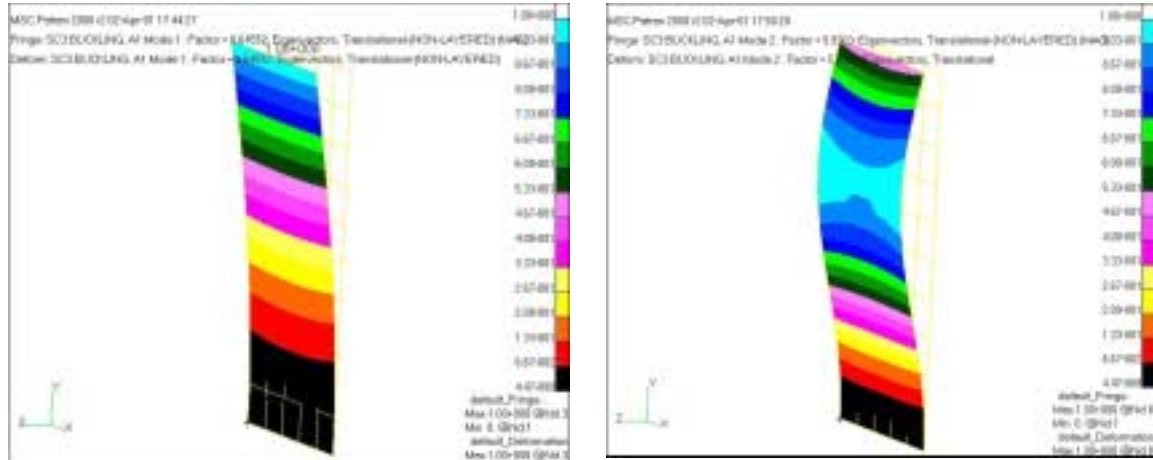


Figure 13-29 Panel Buckling Shapes for the First Two Bending Modes

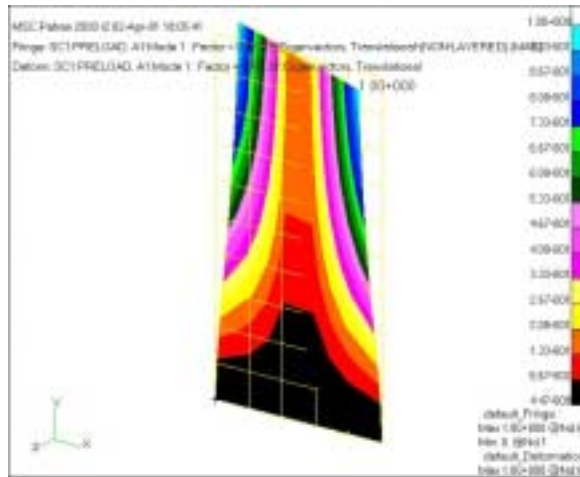


Figure 13-30 Panel Buckling in Twist Mode Under Lateral Compression

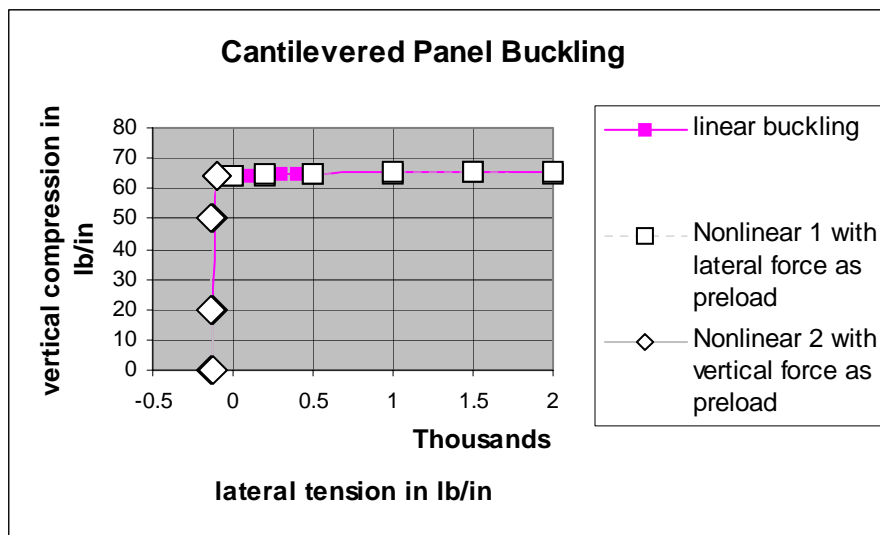


Figure 13-31 Cantilevered Panel Buckling Envelope

Nonlinear Buckling Analysis with Symmetric Stiffness Matrix

Nonlinear buckling analysis capability has been available in SOL 106 by restarts using PARAM, BUCKLE, 1. While serving the purpose of a nonlinear buckling analysis following a static nonlinear analysis, this buckling analysis procedure is cumbersome to the user because it requires a restart. To provide a more convenient and versatile capability for nonlinear buckling analysis, an option for a buckling analysis in a cold-start run with PARAM, BUCKLE, 2 was introduced in MSC.Nastran Version 70.7. This option allows a buckling analysis in any Subcase that has the METHOD command specified for an eigenvalue analysis.

There are two more ways in nonlinear analysis to estimate the critical buckling load. One method is the arc-length method that can provide solutions past the critical buckling load into the post-buckling state. Using the arc-length method, the nodal displacement of a point with the most noticeable motion should be traced to illustrate the peak point. The applied load at this peak resembles the critical buckling load. Another method is to use Newton's method until the solution cannot be obtained due to divergence, in which case the adaptive bisection method is activated in the vicinity of the critical buckling load and stops at the limit load, very close to the critical buckling point. A similar method was applied to the nonlinear dynamic stability analysis using adaptive bisection algorithm in time domain for buckling or bifurcation prediction in the implicit direct-time integration.

The nonlinear buckling analysis procedure in MSC.Nastran allows buckling analysis with preloads. However, the nonlinear procedure follows a different numerical procedure. Therefore, it could provide a means for validation of linear solutions. A parametric study using nonlinear buckling analysis rendered data points for the panel buckling problem, labeled "Nonlinear 1 and 2" in [Figure 13-31](#), using the lateral force and vertical force as preloads, respectively. Although two different methods have been used in the nonlinear buckling prediction: namely, (1) eigenvalue estimation by extrapolation and (2) detection of the critical load by consecutive bisections with no recourse due to divergence, all the nonlinear solution points fell, by and large, on the same curve. This figure also shows that the problem is basically linear, because the "Large Displacement" option introduced no visible effects.

Nonlinear Buckling Analysis with Unsymmetric Stiffness Matrix

Follower forces usually create unsymmetric follower stiffness unless the follower forces are completely enclosed, exerting zero resultant forces. This asymmetry condition should be addressed with caution. The follower stiffness in SOL 106 was introduced in Version 70.7, where symmetrization was applied conservatively so that it is activated in rare occasions. The logic was modified in MSC.Nastran 2001 to always symmetrize the follower stiffness by default. Some problems exhibited convergence difficulties as a drawback of symmetrizing follower stiffness. The symmetrization does not show adverse effects until the follower stiffness effect becomes dominant under large displacements. In MSC.Nastran 2004, the symmetrization is modified to be activated adaptively, detecting the severity of the asymmetry.

For adaptive symmetrization of the follower stiffness, we need a criterion to determine the extent of asymmetry. This can be achieved by normalizing the norm of unsymmetric part of the follower stiffness by a stiffness matrix that is independent from follower stiffness. The differential stiffness was chosen for this purpose after

some numerical experiments. Then, the ratio of asymmetry is compared with a tolerance, which is represented by a user parameter FKSVMFAC so that the user can control the process. The default value for FKSVMFAC is determined to be 0.024 by a parametric study, allowing symmetrization of the follower stiffness during the stiffness update process if the iterative process can render converged solutions. If the asymmetry ratio is greater than FKSVMFAC, symmetrization process will not be activated. The asymmetry ratio increases as the geometric nonlinearity intensifies, which is confirmed by numerous test runs.

When the follower stiffness becomes significantly asymmetric, the real eigenvalue analysis would be invalid or erroneous. The eigenvalue analysis with unsymmetric matrices requires complex arithmetic with left and right-handed eigenvectors. The complex eigenvalue analysis capability is added to SOL 106 in MSC.Nastran 2004 with some modifications to complex Hessenberg and Lanczos methods for unsymmetric buckling analysis. To use the complex eigenvalue analysis, you must specify CMETHOD instead of METHOD in the Subcase and the CMETHOD should reference EIGC Bulk Data entry. The eigenvalues obtained from the complex eigenvalue analysis for buckling with asymmetric stiffness are complex numbers, of which imaginary numbers become zeros. All the eigenvectors are also real numbers. By comparing results from both real and complex eigenvalue analysis, it is confirmed that the real eigenvalue analysis renders wrong solutions to unsymmetric stiffness for nonlinear buckling. Particularly, the mode shapes are erroneous with real eigenvalue analysis from unsymmetric problem, whereas the discrepancy in the critical buckling load is not pronounced.

The buckling shape should be examined graphically to fully understand the physical phenomenon. Fortunately, the complex eigenvalue analysis produces real eigenvectors from real and unsymmetric matrices. However, the buckling shapes displayed with eigenvectors currently are with reference to the undeformed shape, and do not visualize the real buckling shape if the deformed shape at the onset of the buckling has very large displacements.

Example

An example problem is a cantilevered beam model subject to an end force that is a follower force, i.e., FORCE1. The end points rotates 180 degrees by making a complete U shape at the load factor of 2.8 before coming to the shape of deformation shown in [Figure 13-32](#) at a load factor 4. The eigenvalue analysis is performed at load factor 4, where the asymmetry ratio is 0.069, which produced buckling shapes shown in [Figure 13-33](#) (obtained by complex method) and [Figure 13-34](#) (obtained by real eigenvalue method). They are different and the real eigenvalue method is likely to be

in error. The display tool should be improved so that the buckling shapes be plotted with an eigenvector as a perturbation with reference to the deformed shape to fully perceive the physics.

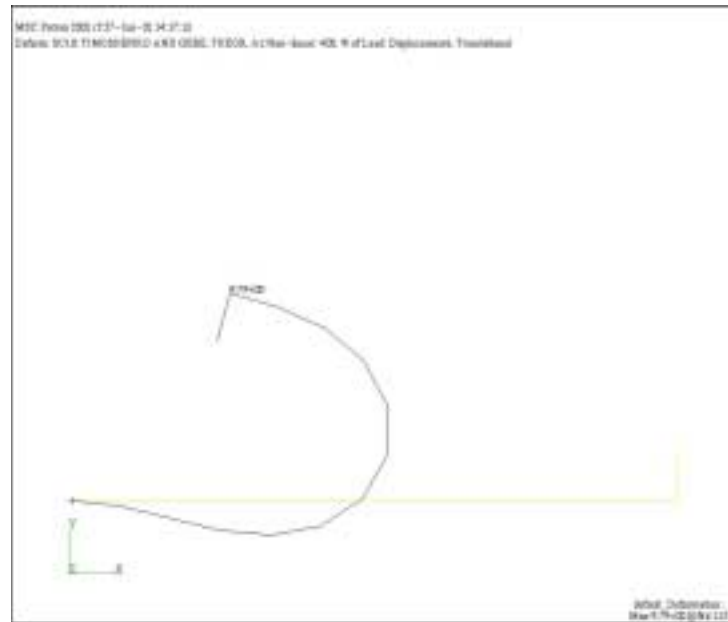


Figure 13-32 Deformed shape at load factor 4 (F1=771.2)

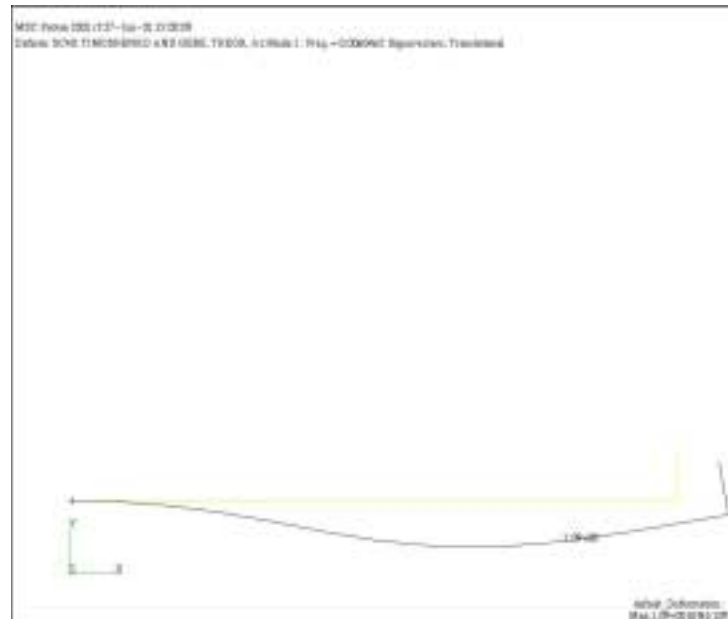


Figure 13-33 Buckling Shape by Complex Method at L.F.=4

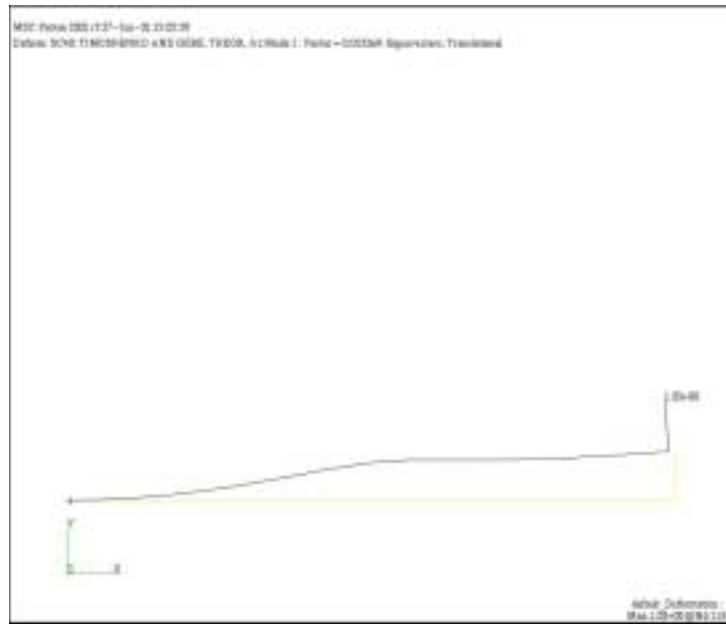


Figure 13-34 Buckling Shape by Real Eigenvalue Analysis at L.F.=4

An Example of Cylinder Buckling

A hollow cylinder is used to demonstrate an interesting behavior in buckling. Its geometric attributes include: 50 in. in diameter, 100 in. in height and 0.125 in. wall thickness with closed ends of 0.5 in. in thickness. The cylinder, made of steel, is subjected to a combined load of external pressure on the cylindrical surface and vertical edge force on the top, while the bottom is fixed in z .

The cylinder could buckle in two different modes depending on the combination of loads as shown in [Figure 13-35](#) and [Figure 13-36](#), where the first mode with vertical creases appear when the external pressure is predominant (showing a star-shaped cross-section at the mid-span) and the second mode with horizontal creases appear when the vertical edge force is more dominant. A failure curve (or buckling envelope) can be constructed by a parametric study using various magnitudes of the preloads with external pressure or vertical edge load, as shown in [Figure 13-37](#). The first buckling shape appears above the curve and the second buckling in the right side of the curve. A bifurcation occurs at the corner, where either buckling mode may occur. The cylinder is free from buckling in the lower-left region of the curve. It is noted that the buckling resistance against the vertical load does not increase even if the pressure direction is reversed to an internal pressure (creating a tensile hoop stress), although the buckling load increases against the external pressure as the vertical compression decreases.

Nonlinear buckling solution points are added to the curve rendered by linear buckling solutions, where three categories of nonlinear solutions are presented. Solution points in square symbol are from the eigenvalue analysis using an extrapolation scheme, and those in diamond symbol are from the limit process using consecutive bisections upon divergence during the incremental/iterative processes. Although the second buckling loads in the right side of the curve predicted by nonlinear analyses are almost identical to the linear solutions, nonlinear solutions for the first buckling shape above the curve are somewhat dispersed. In the absence of an exact solution for a reference, these nonlinear solutions seem plausible to indicate that the geometric nonlinearity has significant effects on the buckling of the first kind with vertical creases, while the geometric nonlinear effect is negligible in the second kind. These solutions, however, were obtained with symmetrized stiffness matrix. The solution points in triangular symbol show tremendously different trend in the upper side of the curve, where the buckling is caused by the external pressure.

The pressure load creates follower stiffness, which is unsymmetric in this problem. The solutions in triangular symbol were obtained with unsymmetric matrices. By preserving asymmetric stiffness matrices, the solutions converged further and the predicted buckling loads were higher. The buckling analysis with asymmetric matrices (but real-valued) requires complex eigenvalue analysis, although the solution turns out to be real-valued. Furthermore, an ideal cylinder without the end condition does not buckle under pressure because of the follower stiffness effect. That is why the solutions with asymmetric stiffness converge to much higher external pressure, and the complex eigenvalue analyses with extrapolation at the last converged steps render solution points shown by triangles. Solutions may continue for increased pressure and it has been observed that the predicted buckling point increases as the eigenvalue analysis is performed at higher pressure. Two of the buckling modes of these solutions are shown in [Figure 13-38](#) and [Figure 13-39](#), which are rather spurious and fortuitous. However, the deformed shape of the second kind of buckling as shown in [Figure 13-40](#) was obtained by a limit process using adaptive bisections upon divergence, which depicts a genuine buckling shape. It also signifies that the nonlinear eigenvalue analysis based on extrapolation is an approximation and it can vary appreciably depending on where the extrapolation is applied.

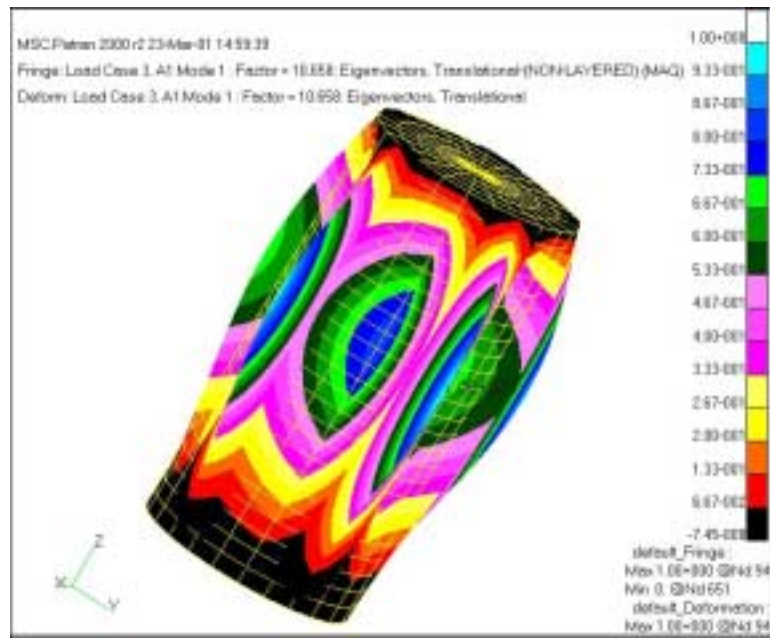


Figure 13-35 Cylinder Buckling Shape 1 With Vertical Creases

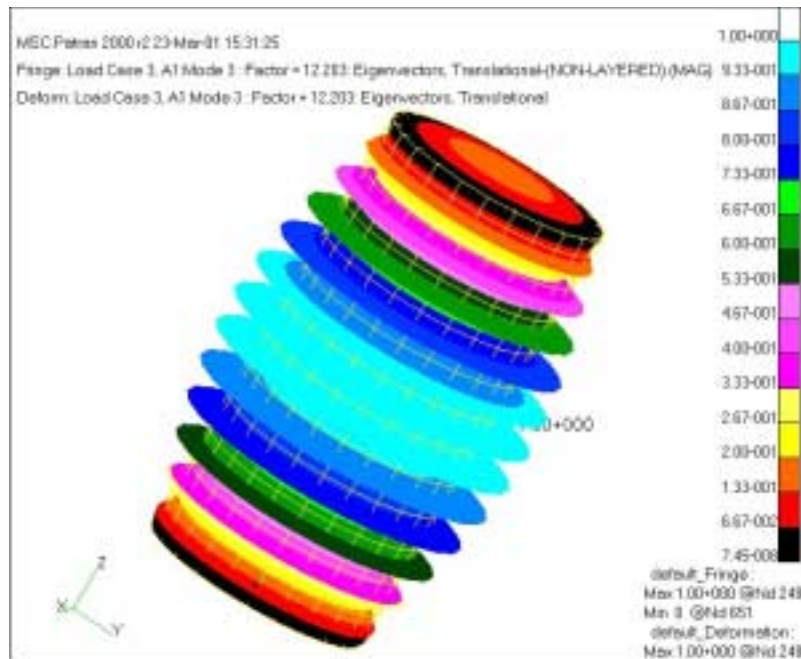


Figure 13-36 Cylinder Buckling Shape 2 with Horizontal Creases

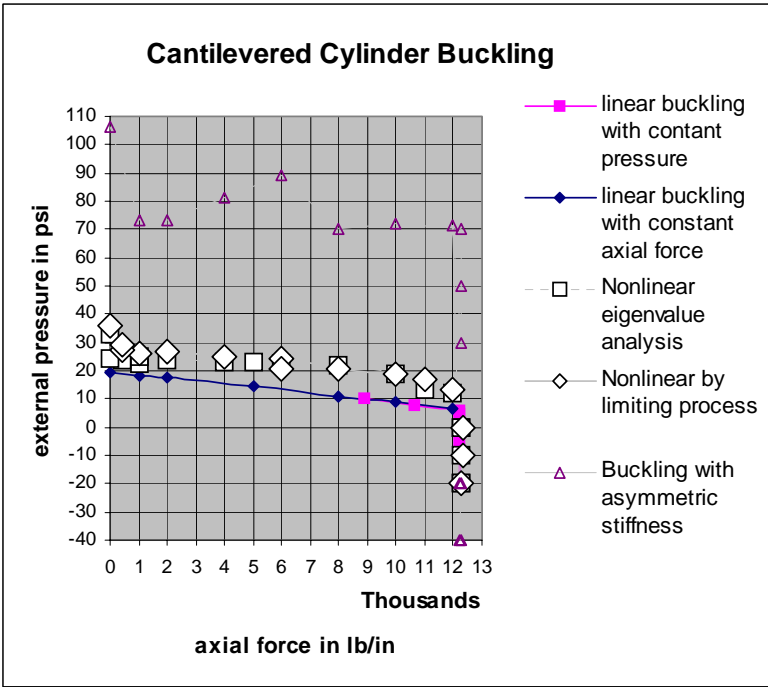


Figure 13-37 Cylinder Buckling Envelope (With Nonlinear Solutions)

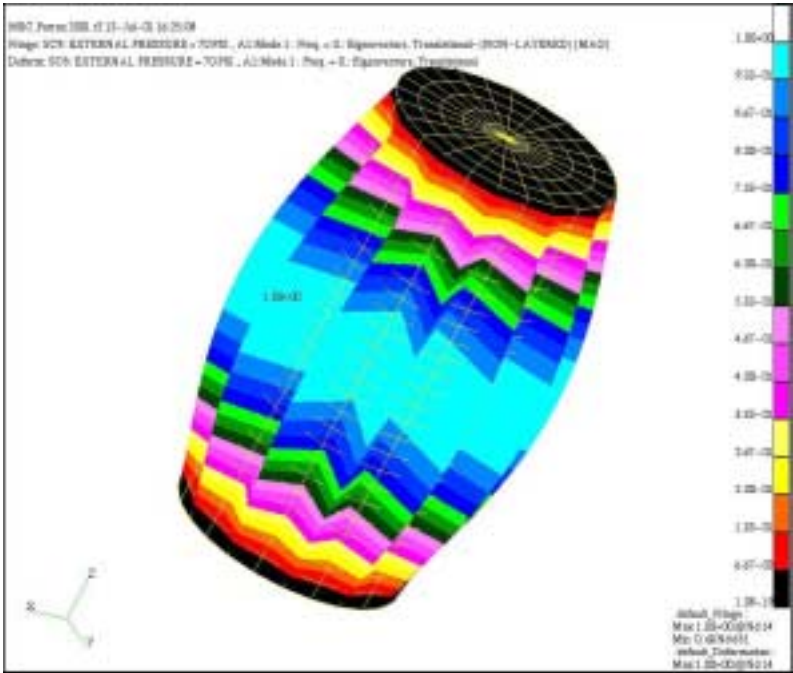


Figure 13-38 Buckling Shape at P axial=2000 and External Pressure=73.44

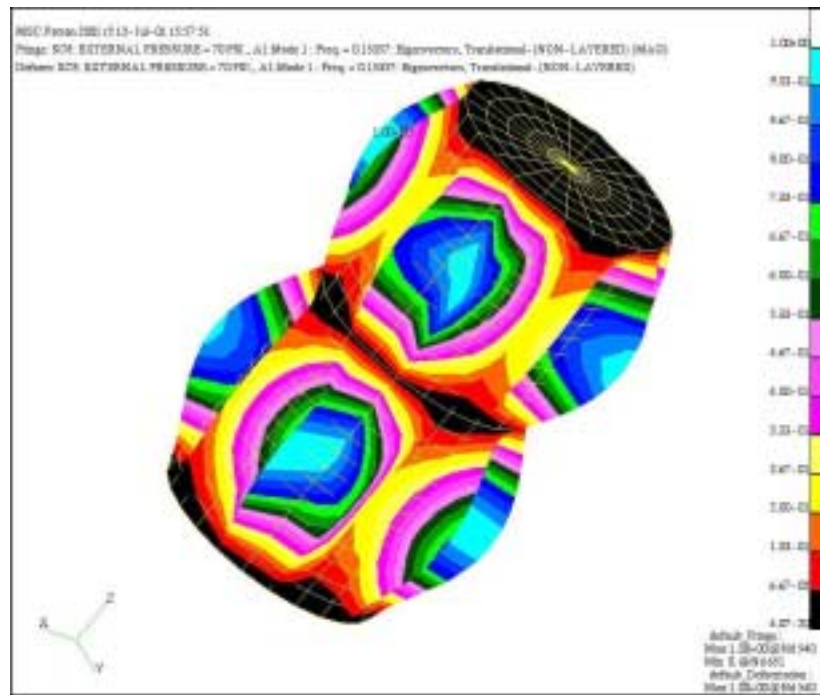


Figure 13-39 Buckling Shape at Axial=10000 and External Pressure=72.02

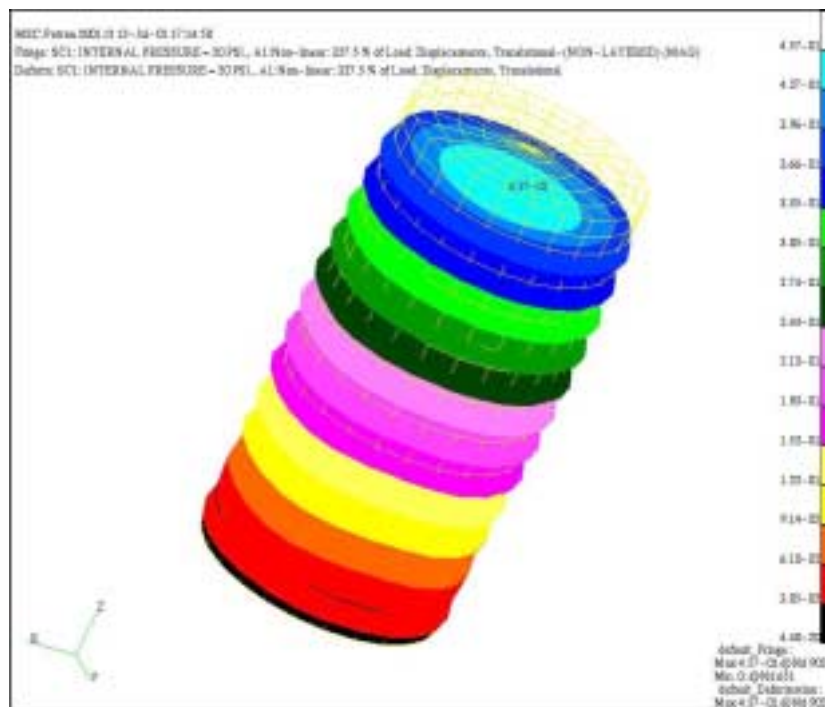


Figure 13-40 Deformed Shape on Buckling at Axial=12243.4 and Internal Pressure=20

13.3.21 Nonlinear Transient Response Analysis

User Interface

The input data required for SOL 129 is a combination of direct time integration control data, similar to SOL 109 (for direct linear transient with superelements), and nonlinear modeling data similar to SOL 106 (for nonlinear statics). The nonlinear properties and/or effects are defined by nonlinear material data (MATS1, MATHP, and TABLES1), gap elements (GAP) or slideline contact (BCONP, BLSEG, and BFRIC) for nonlinear interface, and PARAMeter LGDISP for geometric nonlinearity. The transient effects are produced by time-dependent loading functions (TLOADi, DAREA, LSEQ, etc.), damping (parameters, elements and material data), and mass properties. Massless degree-of-freedom should be avoided for numerical stability and the small damping is also recommended.

The unique data required for SOL 129 is supplied on the TSTEPNL Bulk Data entry. The TSTEPNL entry in itself is a combination of the TSTEP entry for direct time integration and the NLPARM entry for nonlinear iteration control. Restarts are controlled by parameters (LOOPID, STIME, SLOOPID, and SDATA) which can be specified either in the Case Control Section or Bulk Data Section. Some optional parameters (TSTATIC, NDAMP) are provided for additional control or capabilities.

Case Control

Each subcase defines a time interval starting from the last time step of the previous subcase and the interval is subdivided into small time steps. The output time is labeled by the cumulative time, including all previous subcases. There are advantages to divide the total duration of analysis into many subcases so that each subcase does not have excessive number of time steps, e.g., not to exceed 200 steps. The data blocks containing solutions are generated at the end of each subcase to store in the database for output process and restarts. As such, converged solutions are apt to be saved at many intermediate steps in case of divergence and more flexible control becomes possible with multiple subcases.

The input loading functions may be changed for each subcase or continued by repeating the same DLOAD command. However, it is recommended to use the same TLOADi Bulk Data for all the subcases in order to maintain the continuity between subcases, because TLOADi entries define the loading history as a function of cumulative time. Static loads (PLOADi, FORCEi, MOMENTi) may be associated with time-dependent functions by the LSEQ Bulk Data entry that can be selected by a Case Control command LOADSET. However, no thermal loads or enforced displacements (SPCD) are not allowed in the nonlinear transient analysis. Nonlinear forces as

function of displacements or velocities (NOLINI) may be selected and printed by Case Control commands NONLINEAR and NLLOAD, respectively. Each subcase may have a different time step size, time interval, and iteration control selected by the TSTEPNL request. The Case Control commands which may not be changed after the first subcase are: SPC, MPC, DMIG, and TF.

Output requests for STRESS, SDISPLACEMENT, SVELOCITY, SACCELERATION, and NONLINEAR may be specified within a subcase. DISPLACEMENT, VELOCITY, ACCELERATION, OLOAD, FORCE, and SPCFORCE must be specified above the subcase level. However, FORCE output and GPSTRESS output are not available for nonlinear elements.

Initial conditions (displacement or velocity) can be specified by the Bulk Data TIC selectable by the Case Control command IC. If initial conditions were given, all of the nonlinear element forces and stresses must be computed to satisfy equilibrium with the prescribed initial conditions. On the other hand, initial conditions could be a part of the nonlinear analysis by applying static analysis for the preload using PARAM, TSTATIC in the first subcase. Then the transient analysis can be performed in the ensuing subcases. Associated with the adaptive time stepping method, the PARAMeter NDAMP is used to control the stability in the ADAPT method. The parameter NDAMP represents the numerical damping (recommended value for usual case is 0.01) which often required to improve the stability and convergence in the contact problems.

All the superelement model generation options and matrix reduction options are allowed for the linear portion of the structure. The generalized dynamic reduction, component mode synthesis, and Guyan reduction may be performed for upstream superelements. The residual superelement may contain scalar degree-of-freedom representing linear modal formulations.

Limitations

1. No constraint set changes are allowed between subcases or in restarts.
2. No initial preloads such as closed gaps or terminal loads are allowed in a cold start. See PARAM,LOOPID under “Restarts” in this section.
3. Overall structure damping (PARAM,G) is not provided in nonlinear elements. Use the damping field in the material (MATi) data entry and the PARAMeter W4 to supply damping based on the linear properties.
4. Element forces are not computed for nonlinear elements. Output requests will be ignored.

5. When multiple subcases are specified, it is recommended that a single time function (see the TLOADi Bulk Data entry description) be assigned to an applied load throughout the entire time span of the solution. This time function can be used for all subcases. If different time functions are to be used between subcases, ensure that the time functions are continuous between subcases.
6. Element stresses for nonlinear elements are computed and printed during the nonlinear iteration phase only. They are not merged with those of the linear elements.

Direct Time Integration

If the effects of inertia and damping forces are to be included in the nonlinear analysis, the nonlinear transient response is analyzed by a step-by-step integration of an equation of motion, i.e.,

$$M\ddot{u} + C\dot{u} + F(u) = P(t) \quad \text{Eq. 13-18}$$

By direct numerical integration, the equilibrium of [Eq. 13-18](#) is satisfied at discrete time steps with an interval of Δt . The equilibrium is ensured by iterations until the solution converges to the preset error tolerance.

Procedures, such as the quasi-Newton update and line search process, provide users with additional options to adjust incremental and iterative processes in search of more efficient and effective strategy for obtaining solutions. A self-adaptive time stepping method that adjusts time step automatically during analysis is available.

Newmark's direct time integration method is implemented using the two-point recurrence (or one-step) formula as a foundation of this self-adaptive time stepping algorithm. The optimal time step size, which is required for accuracy and efficiency, changes continuously in the transient dynamic environment. The primary concept of automatic time step adjustment is that the proper size of the time step can be predicted based on the dominant frequency in the incremental deformation pattern at the previous time step. This concept presents a deficiency of time lag involved in the prediction process. Furthermore, changes in nonlinearity cannot be predicted from the deformation pattern at the previous time step. This deficiency is overcome by the bisection process, which is activated when any difficulties arise in terms of convergence during the iteration.

Frequent decomposition of the dynamic stiffness matrix is inevitable in adaptive time stepping using the implicit integration method. Efforts have been made to minimize the thrashing in the algorithm (e.g., too frequent or repetitive adjustment of the time step size) and yet to provide a solution reliably to any type of problem. The iteration

process for the equilibrium employs expeditious methods such as BFGS updates and line searches as well as an adaptive stiffness matrix update strategy. The present self-adaptive time stepping algorithm for the nonlinear transient analysis is proving to be a robust and practical method, aimed at efficiency, effectiveness, and user friendliness.

The transient response analysis capability is provided in the self-contained solution sequences, SOLs 129 and 99. SOL 129 is a structured DMAP with the same functionality as SOL 99. Although these solution sequences are developed for nonlinear transient response analysis, linear transient response analysis can be performed in these solution sequences in order to take advantage of the automatic time stepping and restart capabilities. The core parts of SOL 99 and 129 are the nonlinear modules (NLTRD2 and NLTRD) which perform the incremental and iterative processes for implicit direct time integration.

Implicit Integration Control: TSTEPNL Data Entry

The input fields of the TSTEPNL Bulk Data entry specify the time step size, the number of steps, and the output interval as well as the nonlinear iteration options. The TSTEPNL Bulk Data is selectable by the Case Control command TSTEPNL. Although the same TSTEPNL Bulk Data may be selected by more than one subcase, it is recommended to select different TSTEPNL entry for each subcase in preparation for changes in the restarts.

The choice of time step size is determined primarily by the frequency content of the input load and the highest frequency mode-of-interest. A general guideline is that seven or more steps per cycle be provided for reasonable accuracy. Modes with shorter periods (higher frequency) will be attenuated by the numerical process. Highly nonlinear problems may require smaller step size. However, the adaptive time stepping capability is the vital part of 129 and is recommended for any problem (linear or nonlinear). Nevertheless, the initial time step size should be estimated by the user according to the aforementioned principles.

A caution is necessary in using the automatic time step adjustment if the forcing function is a short duration pulse. Since the automatic time step adjustment algorithm does not consider the loading history, short pulses could be skipped if the time step is automatically adjusted to a larger value than the pulse duration. It is advised that a drastic change in the time step size between subcases be avoided. The drastic change, e.g., a ratio exceeding 1000, could cause a loss of accuracy upon subcase switch. In such case, an intermediate subcase should be provided for a transition period of short interval to reduce the ratio.

The parameters for controlling the accuracy and stability of the incremental and iterative process can be specified in the TSTEPNL Bulk Data entry. The controls are applicable to the automatic time step adjustment and bisection process in addition to stiffness matrix updates, BFGS updates and line searches similar to those on the NLPARM Bulk Data entry. Since default values are resulted from numerous test runs, the analysis should be started with the default setting and changed if necessary. The TSTEPNL data format is shown below with default values:

Format:

1	2	3	4	5	6	7	8	9	10
TSTEPNL	ID	NDT	DT	NO		KSTEP	MAXITER	CONV	
	EPSU	EPSP	EPSW	MAXDIV	MAXQN	MAXLS	FSTRESS		
	MAXBIS	ADJUST	MSTEP	RB	MAXR	UTOL	RTOLB		

Example With Defaults:

1	2	3	4	5	6	7	8	9	10
TSTEPNL	250	100	.01	1		2	10	PW	
	1.E-2	1.E-3	1.E-6	2	10	2	0.2		
	5	5		0.75	16.0	0.1	20.		

The TSTEPNL Bulk Data entry is selected using ID by the Case Control command TSTEPNL. Each subcase (residual superelement solutions only) requires a TSTEPNL entry. Multiple subcases are assumed to occur sequentially in time. Therefore, the initial conditions of each subcase are defined by the end conditions of the previous subcase.

The NDT field specifies the number of time steps with DT as the size of each time step. The total duration for the subcase can be assessed by multiplying NDT with DT (i.e., $NDT \cdot DT$). The time increment (Δt) remains constant during the analysis in AUTO and TSTEP options, and is equal to DT. However, the time increment (Δt) changes during the analysis in the ADAPT option and the actual number of time steps will not be equal to NDT. In ADAPT option DT is used as an initial value for Δt .

The NO field specifies the time step interval for output, i.e., every NOth step solution is saved for output. The data will be output at steps 0, NO, ..., etc., and the last converged step for printing and plotting purposes. The Case Control command OTIME may also be used to control the output points.

The program automatically adjusts the incremental time and uses the bisection algorithm in case of divergence. During the bisection process the stiffness is updated at every KSTEPth successful bisection. The stiffness matrix is always updated for a new subcase or restart, irrespective of the option selected.

The number of iterations for a time step is limited to MAXITER. If the solution does not converge in MAXITER iterations, the process is treated as a divergent process, i.e., either a bisection or stiffness matrix update takes place based on the value of MAXBIS. The sign of MAXITER provides a control over the ultimate recourse (reiteration) in case of failure in convergence or bisection. If the MAXITER is negative, the analysis is terminated when the divergence condition is encountered twice during the same time step or the solution diverges for five consecutive time steps. If MAXITER is positive, the program computes the best attainable solution and continues the analysis on second divergence.

The convergence test is controlled by convergence test flags (U for displacement error test, P for load equilibrium error test, W for work error test) and the error tolerances (EPSU, EPSP and EPSW) which define the convergence criteria. All requested criteria (combination of U, P, and/or W) are satisfied upon convergence. It should be noted that at least two iterations are necessary to check the displacement convergence criterion.

The MAXDIV field provides control over diverging solutions. Depending on the rate of divergence, the number of diverging solutions (NDIV) is incremented by 1 or 2. The solution is assumed to be divergent when NDIV reaches MAXDIV during the iteration. If the bisection option is used, the time step is bisected upon divergence. Otherwise, the solution for the time step is repeated with a new stiffness based on the converged state at the beginning of the time step. If NDIV reaches MAXDIV twice within the same time step, the analysis is terminated with a fatal message.

The BEGS quasi-Newton updates and the line search process work in the way as in static analysis except for the default setting. The MAXQN field defines the maximum number of quasi-Newton vectors to be saved on the database and the MAXLS defines the number of line searches allowed per iteration. Nonzero values of MAXQN and MAXLS activate the quasi-Newton update and the line search process, respectively.

The FSTRESS field defines a fraction of the effective stress ($\bar{\sigma}$) which is used to limit the subincrement size in the material routine. The number of subincrements in the material routines is determined such that the subincrement size is approximately $FSTRESS \cdot \bar{\sigma}$ (equivalent stress). FSTRESS is also used to establish a tolerance for error correction in the elasto-plastic material, i.e.,

$$Error \text{ in yield function} < FSTRESS \cdot \bar{\sigma}$$

Eq. 13-19

If the limit is exceeded at the converging state, the program will exit with a fatal error message. Otherwise, the stress state is adjusted to the current yield surface.

MAXBIS is the maximum number of bisections allowed for each time step ($-9 \leq \text{MAXBIS} \leq 9$). The bisection process is activated when divergence occurs and $\text{MAXBIS} \neq 0$. The number of bisections for a time increment is limited to $|\text{MAXBIS}|$. If MAXBIS is positive and the solution does not converge after MAXBIS bisections, the best solution is computed and the analysis is continued to the next time step. If MAXBIS is negative and the solution does not converge in $|\text{MAXBIS}|$ bisection, the analysis is terminated.

The parameter ADJUST allows the user to control the automatic time stepping in the ADAPT option. A value of zero for ADJUST turns off the automatic adjustment completely. If ADJUST is positive, the time increment is continually adjusted for the first few steps until a good value of Δt is obtained. After this initial adjustment, the time increment is adjusted every ADJUST time steps only. A value of ADJUST an order greater than NDT will turn off adjustment after the initial adjustment. Since the automatic time step adjustment is based on the mode of response and not on the loading pattern, it may be necessary to limit the adjustable step size when the period of the forcing function is much shorter than the period of dominant response frequency of the structure. It is the user's responsibility to ensure that the loading history is properly traced with ADJUST option. The ADJUST option should be suppressed for the duration of short pulse loading. If unsure, start with DT which is much smaller than the pulse duration in order to properly represent the loading pattern.

MSTEP defines the desired number of time steps to obtain the dominant period response accurately ($10 \leq \text{Integer} \leq 200$). RB defines bounds for maintaining the same time step for the stepping function in the automatic time step adjustment method ($0.1 \leq \text{Real} \leq 1.0$). Parameters MSTEP and RB are used to adjust the time increment during the analysis in the ADAPT option. The adjustment is based on the number of time steps desired to capture the dominant frequency response accurately. The time increment is adjusted as follows:

$$\Delta t_{n+1} = f(r)\Delta t_n$$

where:

$$r = \frac{1}{\text{MSTEP}} \left(\frac{2\pi}{w_n} \right) \left(\frac{1}{\Delta t_n} \right)$$

with:

$$f = 0.25 \text{ for } r < 0.5 \cdot \text{RB}$$

$$f = 0.5 \text{ for } 0.5 \cdot RB < r < RB$$

$$f = 1.0 \text{ for } RB < r < 2$$

$$f = 2.0 \text{ for } 2. \leq r < 3./RB$$

$$f = 4.0 \text{ for } r > 3./RB$$

The recommended value of MSTEP for nearly linear problems is 20. A larger value (e.g., 40) is required for highly nonlinear problems. In the default option, the program automatically computes the value of MSTEP based on the changes in the stiffness.

The MAXR field defines the maximum ratio for the adjusted incremental time relative to DT allowed for time step adjustment ($1.0 \leq \text{Real} \leq 32.0$). MAXR is used to define the upper and lower bounds for adjusted time step size, i.e.,

$$\text{MIN}\left(\frac{DT}{2^{\text{MAXBIS}}}, \frac{DT}{\text{MAXR}}\right) \leq \Delta T \leq \text{MAXR} \cdot DT \quad \text{Eq. 13-20}$$

The UTOL field defines the tolerance on displacement increment below which there is no time step adjustment ($0.001 < \text{Real} \leq 1.0$). UTOL is used to filter undesirable time step adjustment, i.e., no time step adjustment is performed if

$$\frac{\|\dot{U}_n\|}{\|\dot{U}_n\|_{\text{max}}} < \text{UTOL} \quad \text{Eq. 13-21}$$

The RTOLB field defines the maximum value of incremental rotation (in degrees) allowed per iteration to activate bisection ($\text{Real} > 2.0$). The bisection is activated if the incremental rotation for any degree-of-freedom ($\Delta\theta_x$, $\Delta\theta_y$ or $\Delta\theta_z$) exceeds the value specified for RTOLB. This bisection strategy based on the incremental rotation is controlled by MAXBIS field.

Iteration Related Output Data

During the incremental and iterative computation, the process information consisting of iteration data is printed at the end of each iteration or time step. They are printed under the following heading:

TIME	Cumulative time for the duration of the analysis.
ITER	Iteration count for each time step.

DISP Relative error in terms of displacements defined as

$$E_U^i = \lambda^i \frac{\|u^i - u^{i-1}\|}{(1 - \lambda^i)u_{max}} \quad \text{for ADAPT method}$$

$$E_U^i = \frac{\|u^i - u^{i-1}\|}{(1 - \lambda^i)u_{max}} \quad \text{for AUTO or TSTEP}$$

where $u_{max} = \max(\|u_1\|, \|u_2\|, \dots, \|u_n\|)$

LOAD Relative error in terms of load vectors defined as

$$E_p^i = \frac{\|R^i\|}{\max(\|F_n\|, \|P_n\|)} \quad \text{for ADAPT method}$$

$$E_p^i = \frac{\|R^i\|}{\max(\|\bar{F}\|, \|\bar{P}\|)} \quad \text{for AUTO or TSTEP}$$

where \bar{F} and \bar{P} are three-point average values for internal and external forces, respectively.

WORK Relative error in terms of work defined as

$$E_w^i = \frac{\{u^i - u^{i-1}\}^T R^i}{\max(\{u_n\}^T \{F_n\}, \{u_n\}^T \{P_n\})} \quad \text{Eq. 13-22}$$

LAMBDA(I) Rate of Convergence in iteration where

$$\lambda^i = \frac{E_p^i}{E_p^{i-1}} \quad \text{Eq. 13-23}$$

LAMBDA(T) Ratio of the load error for two consecutive time steps computed only for AUTO or TSTEP method

$$\lambda_n^n = \frac{E_p^n}{E_p^{n-1}} \quad \text{Eq. 13-24}$$

LAMBDA-BAR Average of LAMBDA(T) over the last 3 steps, computed only for AUTO or TSTEP method.

DLMAG	Absolute norm of the load vector, $\ R\ $. The absolute convergence is defined using DLMAG by $\ R\ < 10^{-12}$.
FACTOR	Final value of the line search parameter.
E-FIRST	Divergence rate, error before line search.
E-FINAL	Error at the end of line search.
NQNV	Number of quasi-Newton vectors appended.
NLS	Number of line searches performed during the iteration.
ITR DIV	Number of occurrences of divergence detected during the adaptive iteration by the iteration module NLTRD2.
MAT DIV	Number of occurrences of bisection conditions in the material routine (excessive stress increment) or in the rotation angle (excessive rotation) during the iteration using the ADAPT method.
NO. BIS	Number of bisections executed for the current time interval.
ADJUST	Ratio of time step adjustment relative to DT within a subcase.

Restarts

Since SOL 106 and SOL 129 share the same database storage for nonlinear tables and matrices, the restart system for transient analysis can use either a previous static or transient nonlinear analysis as its initial conditions. A restart from SOL 129 into SOL 106 is not available.

Restarting from SOL 106 into SOL 129

The options for a restart from SOL 106 (66) into SOL 129 (99) are static to static and static to dynamic analysis. For a restart from a previous static analysis, only the first subcase is affected. Simply provide a database created in SOL 106 (or 129) and specify the parameter PARAM,SLOOPID,N where N is the printed value of LOOPID for the desired static solution. The initial transient load should be identical to static loads at the restart state. Constraint sets, direct input matrices, mass, and damping may be changed.

Restarting within SOL 129 or 99

Restarting within SOL 129 or 99 allows static to static, static to dynamic, and dynamic to dynamic analysis. Restarts from a previous nonlinear transient execution are available for a number of cases. If the same model is to be reexecuted, only the

residual superelement needs to be reassembled. If the final results from the previous transient run are to be used as the initial conditions at $t = 0$, add N dummy SUBCASE commands to start the residual Case Control execution and $STIME = 0$.

The normal restart for a transient run is to be continued from the last step of a previous subcase with different loads and/or TSTEPNL data. For the normal restart provide the following parameters:

LOOPID = N : Start from the N th subcase
 STIME = t : Start from time t

Note that constraint sets should not be changed to avoid incompatible matrix sizes. The values of LOOPID and STIME, which are printed with the iteration information for each subcase, can be directly read from the printout of the previous run. If the adaptive time stepping algorithm is not activated, the value of t may also be calculated by the following equation:

$$t = \left(\sum_{i=1}^N NDT_i \right) \cdot DT_i \quad \text{Eq. 13-25}$$

where NDT_i and DT_i are the number of time steps and the time increment of the i -th subcase, respectively.

If a SOL 129 or 99 run is terminated abnormally in the middle of a subcase, it may or may not be restartable depending upon the cause of the abnormal exit. If the job is stopped due to a diverging solution, it can be restarted either from the end of a previous subcase or from the last saved solution step. The restart procedure for the former is identical to that for the normal restart as described in the preceding paragraph. The latter case also requires parameters LOOPID and STIME, however, the input value for STIME differs depending on the value of METHOD specified on the TSTEPNL entry. If METHOD = AUTO or TSTEP (the NLTRD module), STIME is the time corresponding to the last output step which may be calculated based on the output skip factor (i.e., the NO on the TSTEPNL entry). If METHOD = ADAPT (NLTRD2 module) the last converged solution is treated as an output step and is always saved for the restart so that STIME can be the time of the last converged step. The values of STIME and LOOPID can also be found in the printout, if the ADAPT method is used. Once STIME and LOOPID are known, determine the number of remaining time steps in the subcase and create a new TSTEPNL entry for the remaining time. Insert a new subcase which references the new TSTEPNL entry prior to the remaining subcases in the Case Control Section.

A solution may be terminated in the middle of a subcase due to insufficient CPU time: (1) the CPU time specified in the Executive Control Section is insufficient so the run is forced to exit by the program, or (2) the CPU time exceeds the limit specified in the computer system which leads to a sudden job abortion by the system. In the first case, MSC.Nastran is able to detect the specified CPU time in the Executive Control Section and automatically activate the wrapping-up procedure before time expiration. When completed, the solution can be restarted from the termination point as in the solution diverging case. In the second case the solution can only be restarted from the end of a subcase.

Restarts may also be performed solely for data recovery by providing the following parameters:

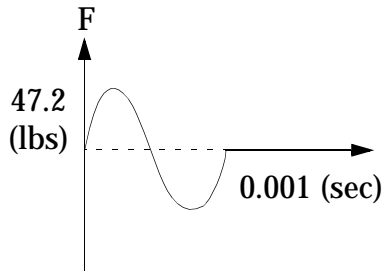
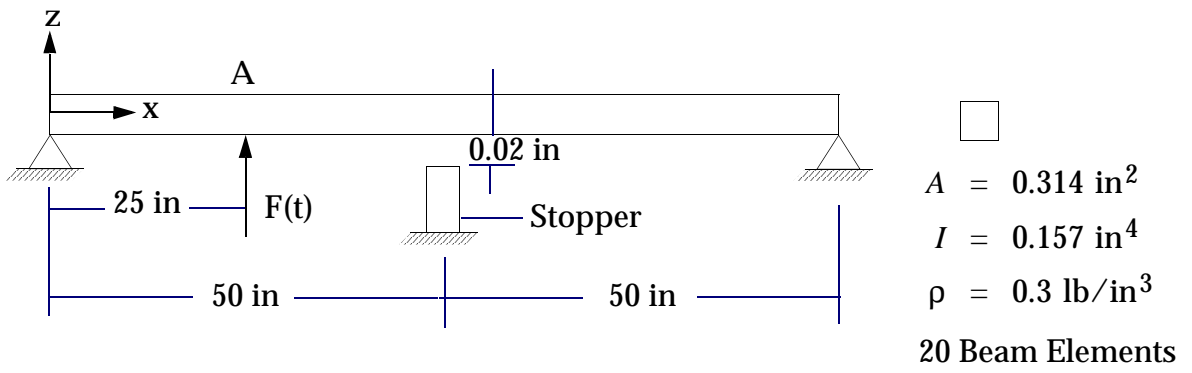
SDATA = -1 : Recover data without running the solution module LOOPID = N : from the 1st through the N-th subcases.

Note that solution sets DISP, VELO, ACCE, OLOAD, SPCF (printout and plotting) and NLLOAD (plotting only) are recoverable while STRESS, SDISP, SVELO and SACCE sets are not.

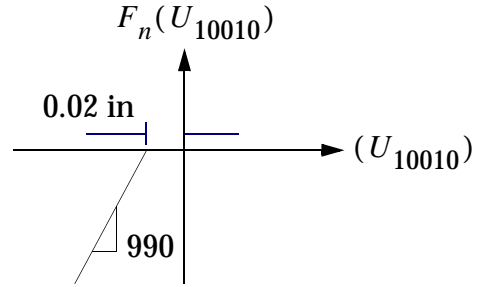
Example

A simply supported elastic beam is subjected to a single cycle of a sinusoidal forcing function at the quarter span as illustrated in [Figure 13-41](#). The beam is modeled using 20 linear CBEAM elements. A stopper (snubber) is present underneath the center of the beam with a clearance of 0.02 inches. This gap is simulated by a displacement dependent force which is active only when the beam is in contact with the snubber. The problem is linear except for the effect of the stopper. The input data for this problem is given in [Listing 13-13](#).

One subcase with 100 time steps and an initial time step size of 0.1 msec was analyzed in the cold start run. Since the adaptive time stepping algorithm was activated, the job ran to completion at 9.7 msec instead of the specified time span, $100 \times 0.1 = 10$ msec. A new subcase with 600 time steps was added in the first restart run to extend the time span for another 30 msec. The adaptive time stepping method was turned off and the execution time was specified to be 0.8 min. As expected, the run was aborted at 36 msec due to insufficient time. In the second restart run, the execution time was changed to 5.0 min. and another subcase with 164 time steps and a time increment of 0.1 msec was added to extend the analysis time span up to 40 msec. The third restart was simply to recover the DISP, VELO, ACCE, OLOAD, NLLOAD and SPC output. The response at the loading point of the beam is shown in [Figure 13-42](#).



Forcing Function



NOLIN1 Representing GAP

Figure 13-41 Simply Supported Beam With a Restrained Motion

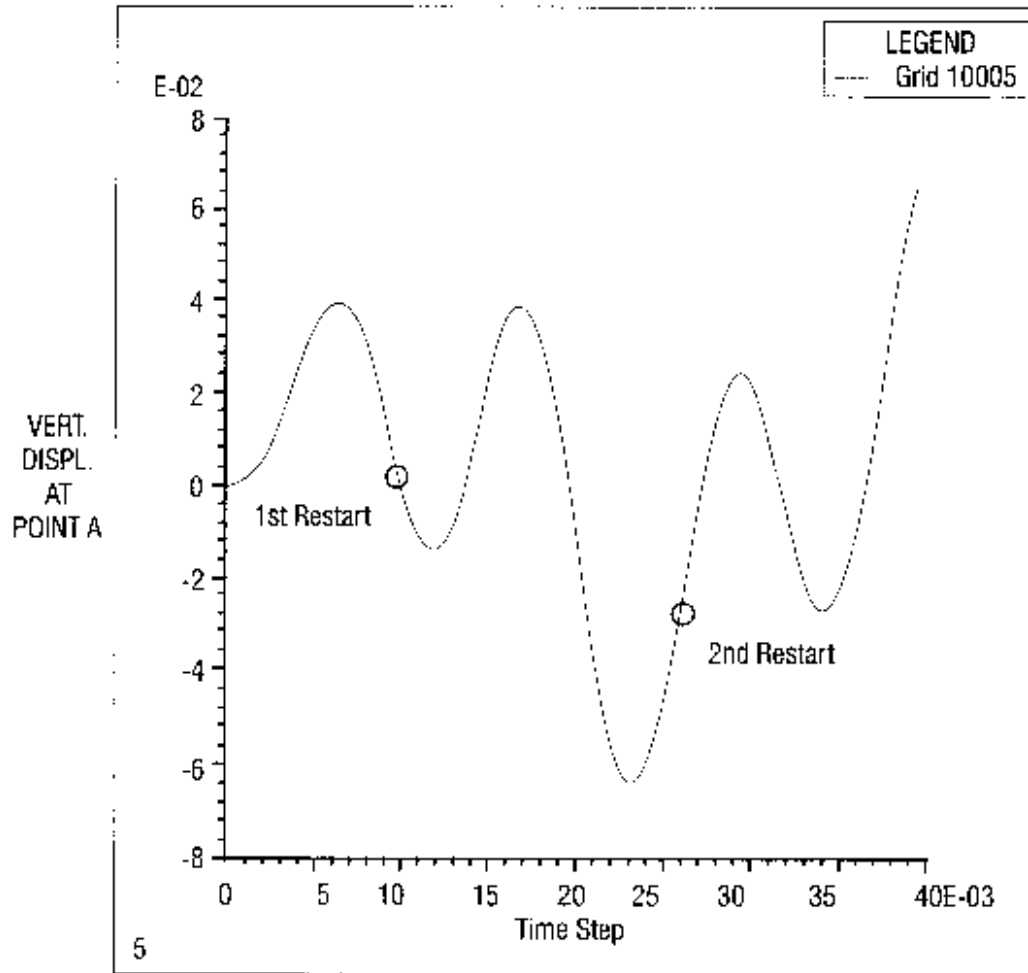


Figure 13-42 Simply Supported Beam: Loading Point Response

Listing 13-13 Input Data Listing for Simply Supported Beam

```

$RESTART          $ ***added for the restart runs
ID TNOLIN, V66    $ SSH 7/10/91
TIME $           $
SOL 99           $ NONLINEAR TRANSIENT ANALYSIS
DIAG 8,50        $ PRINT MATRIX TRAILERS AND ITERATIONS
CEND
TITLE = DYNAMICS HANDBOOK DEMONSTRATION PROBLEM          D9918
SUBTITLE = DIRECT TRANSIENT RESPONSE, NONLINEAR FORCE
LABEL = NOLIN IN SOL 99
SEALL = ALL
ECHO = UNSORT
  SET 1 = 10005
  SET 2 = 10010
  SET 3 = 10000
  SPC = 1002
$-----
$PARAM,LOOPID,1      *** used for the 1st restart
$PARAM,STIME,0.0093
$-----
$PARAM,LOOPID,2      *** used for the 2nd restart
$PARAM,STIME,0.0236
$-----
$ DISPL=2
$ VELO=2             *** used for the 3rd restart
$ ACCEL=2
$ OLOAD=1
$ NLLOAD=2
$ SPCF=3
$PARAM,LOOPID,3
$PARAM,SDATA,-1
$-----
SUBCASE 1
  DLOAD          = 30
  TSTEPNL        = 10
  NONLINEAR      = 13  $ SELECT NONLINEAR FORCE
SUBCASE 2
  DLOAD          = 30
  TSTEPNL        = 20
  NONLINEAR      = 13  $ SELECT NONLINEAR FORCE
SUBCASE 3
  DLOAD          = 30
  TSTEPNL        = 30
  NONLINEAR      = 13  $ SELECT NONLINEAR FORCE
BEGIN BULK
$ MODELING INFORMATION FOR BEAM ONLY
CBAR      101    100    10000  10001  0.      0.      1.
CBAR      102    100    10001  10002  0.      0.      1.
CBAR      103    100    10002  10003  0.      0.      1.
CBAR      104    100    10003  10004  0.      0.      1.
CBAR      105    100    10004  10005  0.      0.      1.
CBAR      106    100    10005  10006  0.      0.      1.
CBAR      107    100    10006  10007  0.      0.      1.
CBAR      108    100    10007  10008  0.      0.      1.
CBAR      109    100    10008  10009  0.      0.      1.
CBAR      110    100    10009  10010  0.      0.      1.
CBAR      111    100    10010  10011  0.      0.      1.
CBAR      112    100    10011  10012  0.      0.      1.
CBAR      113    100    10012  10013  0.      0.      1.
CBAR      114    100    10013  10014  0.      0.      1.
CBAR      115    100    10014  10015  0.      0.      1.

```

Listing 13-13 Input Data Listing for Simply Supported Beam (continued)

```

CBAR      116      100      10015      10016      0.      0.      1.
CBAR      117      100      10016      10017      0.      0.      1.
CBAR      118      100      10017      10018      0.      0.      1.
CBAR      119      100      10018      10019      0.      0.      1.
CBAR      120      100      10019      10020      0.      0.      1.
CONM2     12      10010          .1
GRID      10          50.          -1.
GRID     10000          0.      0.      0.      1246
GRID     10001          5.      0.      0.      1246
GRID     10002          10.     0.      0.      1246
GRID     10003          15.     0.      0.      1246
GRID     10004          20.     0.      0.      1246
GRID     10005          25.     0.      0.      1246
GRID     10006          30.     0.      0.      1246
GRID     10007          35.     0.      0.      1246
GRID     10008          40.     0.      0.      1246
GRID     10009          45.     0.      0.      1246
GRID     10010          50.     0.      0.      1246
GRID     10011          55.     0.      0.      1246
GRID     10012          60.     0.      0.      1246
GRID     10013          65.     0.      0.      1246
GRID     10014          70.     0.      0.      1246
GRID     10015          75.     0.      0.      1246
GRID     10016          80.     0.      0.      1246
GRID     10017          85.     0.      0.      1246
GRID     10018          90.     0.      0.      1246
GRID     10019          95.     0.      0.      1246
GRID     10020         100.     0.      0.      1246
MAT1      1000      3.+7          .3      .3
PARAM     GRDPNT  10010
PARAM     WTMASS  .002588
PBAR      100      1000      .31416      .15708          +PBl
+PBl 1.      0.
SPC       1002      10      123456
SPC       1002      10020      3          10000      3
$
$ MODELING INFORMATION FOR CENTER SPRING
CROD      10      10      10      10010
MAT1      10      10.     0.
PROD      10      10      1.
MATS1     10      PLASTIC 0.      1      1      3.+8
$ LOADING AND SOLUTION INFORMATION
TLOAD2    30      33      0.          .011451      87.33      -90.
DAREA     33      10005      3          47.2
TSTEPNL   10      100      .0001      1      ADAPT
TSTEPNL   20      600      .00005     1      ADAPT          +T521
+T521          +T522
+T522      0
TSTEPNL   30      164      .00010     1      ADAPT          +T531
+T531          +T532
+T532      0
$
$ MODELING INFORMATION FOR NONLINEAR SPRING
NOLIN1   13          10010      3      1.      10010      3      13
TABLED1   13          +TAB
+TAB      -2.5-2      4.95      -2.0-2      0.      0.      0.      ENDT
$
ENDDATA

```

13.3.22 Performance Improvements in SOL 129

Introduction

Nonlinear transient response analysis in SOL 129 has become more robust and reliable. It can be used for linear analysis to take advantage of the automatic time stepping. If the problem is truly linear, i.e. without NOLIN, SOL 129 should converge in one iteration at every time step. The computing speed is comparable to SOL 109 in such cases if the same number of time steps are used. MSC.Nastran 2004 implements an updated automatic time step adjustment algorithm. The automatic time stepping algorithm implemented over ten years ago was hinged on the dominant response frequency which is based on the system characteristics. This strategy is not effective if the loading changes faster than the system response. In case of an impact loading, an indiscriminate use of automatic time stepping would cause the impulse represented inadequately or unrecognized by the algorithm. This could be avoided by turning off the automatic time stepping. But it relies on the user's intervention and therefore is not user-friendly.

Benefits

SOL 129 will activate time step adjustment unless it is suppressed. As a result, the solution will be more accurate and the solution time is reduced, especially the engineering time will be significantly saved because accurate solution may be attained in the first trial. That is the goal of the automatic and adaptive time stepping method. However, if the model includes NOLIN entries, SOL 129 is likely to activate iteration for equilibrium, while SOL 109 takes one-step solution for each time step. That is because SOL 109 uses explicit approach for the NOLIN process, taking NOLIN forces at the previous time step for the equilibrium of the current time step. The iteration makes SOL 129 slower than SOL 109 when NOLIN entries are present, but the solution is more accurate. The iteration could be avoided by using large tolerances for convergence criteria.

Saving engineering time is by far more efficient even if the computation takes more time. The automatic time stepping strives for a one-shot solution for this reason, and the algorithm takes a rather conservative approach, i.e., taking a smaller stride to avoid divergence or convergence difficulties. Particularly, bisection algorithms are devised for accuracy and effectiveness, not for efficiency. If the accuracy can be compromised for efficiency, the adaptive bisections should be turned off. Also larger strides can be achieved by specifying a smaller value for MSTEP field (e.g., 10) in the TSTEPNL entry.

The following performance comparison of the analyses of a model with a lot of NOLIN entries shows the contrast in efficiency versus strategy:

Table 13-15

Version	method	tsteps	usertime	sysstime	elapsed	Remarks
2001	fixed	3200	1787.69	1208.59	50:52	no bisections
2001	auto		1662.74	1187.90	49:20	no bisections
2004	fixed	3200	1520.04	1147.16	45:20	nolin=0, tzeromax=0
2004	fixed	37039	17567.10	10535.30	9:48:34	bisections allowed
2004	auto	4540	1434.06	1228.18	47:42	nolin=0, tzeromax=0
2004	auto	14652	11854.37	5724.78	5:02:58	nolin=1., tzeromax=1
2004	auto	1320	829.84	570.50	23:38	nolin=0., tzeromax=1, mstep=10

Theory

Follower Force Generation

First, the load generation should be performed in the iteration module for efficiency. Direct time integration (in NLTRD2) is to solve the equation

$$M\ddot{u} + C\dot{u} + F(u) = P(t) \quad \text{Eq. 13-26}$$

at every incremental time step. This process involves an integration in time domain combined with an iterative process at each time step to satisfy equilibrium. The iteration is based on an incremental process using

$$\left[\frac{4}{\Delta t^2} M + \frac{2(1-\eta)}{\Delta t} C + \tilde{K} \right] \{ \Delta U^{i+1} \} = \left\{ R_{n+1}^i \right\} \quad \text{Eq. 13-27}$$

with

$$\left\{ R_{n+1}^i \right\} = \left\{ P_{n+1}^i - F_{n+1}^i \right\} + \frac{4(1-\eta)}{\Delta t} M \{ \dot{U}_n \} + (1-2\eta) \{ P_n - F_n \} - \left[\frac{4}{\Delta t^2} M + \frac{2(1-\eta)}{\Delta t} C \right] \left\{ U_{n+1}^i - U_n \right\} \quad \text{Eq. 13-28}$$

where η denotes numerical damping (PARAM, NDAMP).

Currently, the load vectors (P_n) are generated for a constant time interval by the module TRLG, and they are interpolated or extrapolated in the NLTRD2 module adaptive to the time step size. This is not only inefficient, but also makes it difficult to adjust the time step size based on the loading history. Particularly, the follower forces are recomputed in the NLTRD2 at every iteration (i) at time t as follows:

$$P_t^i(u) = \frac{\Delta t_r}{\Delta t} \{ P_n \} + \left(1 - \frac{\Delta t_r}{\Delta t} \right) \{ P_{n+1} \} + f(t) \cdot \left\{ P_f(u^i) - P_f(0) \right\} + N_t^i(u, u) \quad \text{Eq. 13-29}$$

where the first two terms represent a linear interpolation between two load vectors at steps n and $n+1$ (from TRLG), the third term a correction for follower forces, and the fourth term NOLIN forces.

The new design implemented in MSC.Nastran 2004 is to generate load vectors in NLTRD2 one column at each time step, so that no update or interpolation is necessary; i.e.,

$$P_t^i(u) = f_{nf}(t) \cdot \{ P_{nf} \} + f_f(t) \cdot \left\{ P_f^i \right\} + N_t^i \quad \text{Eq. 13-30}$$

where the first term represents non-follower forces, which is to be generated in NLTRD2 (migrated from TRLG). Notice that the non-follower force vectors are formed once per time step, but the follower force vector (the second and the third terms) must be computed at every iteration. It improves not only load-dependent adaptive time stepping but also the memory usage and reduces the I/O operations.

Load-Dependent Time Step Adjustment

The second step is to make the time step adjustment algorithm adaptive to the loading history based on the TLOADi data. The adaptive time step algorithm for TLOAD1 is implemented by considering time steps specified in the TABLEDi entries. A new time step size is compared against time interval in the TABLED(1-4) at every time step. When adjusted step exceeds the user-specified load increment interval or causes an

abrupt changes in load magnitude, the adjusted value is bisected to be consistent with the binary-based time adjustment algorithm. This process will continue sweeping through the time duration for these subcase.

The transient loads specified by TLOAD2 are analytical functions, continuous and smooth, but the excitation frequency should be considered for adaptive time stepping so that the period (field F on TLOAD2) is less than 8 times the incremental time step, i.e., bisect if the time step size is larger than 1/8 of a period.

This algorithm activates an adaptive bisection upon load spikes to account for impulsive loads with adequate number of time steps. This modifications work for both structural analysis and heat transfer analysis. Linear forces (i.e., any loads other than follower forces) have not been implemented in MSC.Nastran 2004, and yet to be included in the time step adjustment.

Adaptive Time Stepping for NOLIN and NLRGAP

The third phase makes the time step adjustment algorithm sensitive to the NOLIN data. The NOLIN forces, denoted by $N_t(u, \dot{u})$ in Eq. 13-29, are implicit functions of time (except for NLRGAP that could be explicit function of time), which may vary abruptly as a function of time. Then the solution may not render a correct solution with a large time step. The NOLIN entries consists of

- NOLIN1 with TABLEDi: $N1_i = s \cdot T_j(u, \dot{u})$
- NOLIN2: $N2_i = s \cdot X_j \cdot X_k$
- NOLIN3: $N3_i = s \cdot X_j^A$
- NOLIN4: $N4_i = -s \cdot [-X_j]^A$
- NLRGAP: simulates a radial forces on a journal bearing joint

The criteria for the abruptness can be as follows:

$$\Delta t_{n+1} = 0.5 \cdot \Delta t \quad \text{if} \quad |N_{t+\Delta t} - N_t| > F_n \cdot \text{Max}(|N_t|, 10^{-4})$$

where the factor F_n is determined based on the numerical experiments (defaulted to 1000). This factor F_n is controlled by system cell 386 (=nolin), i.e.,

$$F_n = 1000. / nolin$$

where nolin is the alias name of the real-valued sys(386) as follows:

- nolin = 0 bisection is suppressed (same as MSC.Nastran 2001)
 nolin = 0.001 increase accuracy slightly
 nolin = 0.01 increase accuracy a little more
 nolin = 1.0 allow full adaptive bisection (default)

Consecutive bisections are allowed (KSTEP+1) times, in which KSTEP is defaulted to 2 in the TSTENL entry.

Until MSC.Nastran 2001, NOLIN forces were added to the right-hand side and treated like an external loads as shown in [Eq. 13-29](#). In order to make a full implementation of NOLIN forces as internal forces, they have to be subtracted from element forces, and thus NOLINs are elements having stiffness and damping. To remedy the automatic time stepping without full implementation as elements, NOLINs need to be considered like an internal force in computing a dominant frequency for a time stepping criterion. The dominant frequency should then be computed by

$$\omega_n^2 = \frac{\{\Delta u_n\}^T \{\Delta F_n - \Delta N_n\}}{\{\Delta u_n\}^T [M] \{\Delta u_n\}} \quad \text{Eq. 13-31}$$

where:

$$\Delta N_n = \{N_n\} - \{N_{n-1}\}$$

The effectiveness for NOLIN-dependent time step adjustment is demonstrated in the two examples below: a vibrating beam with a stopper simulated by NOLIN1 and a thermostat model with a NOLIN1. The result is the accuracy improvement, not efficiency, as shown in the figures. The vibrating beam experiences abrupt changes in the NOLIN1 force (contact) when the GAP opens or closes. The adaptive bisection algorithm detects the transition from open to closed or vice versa. The thermostat model has similar characteristics as a vibrating beam.

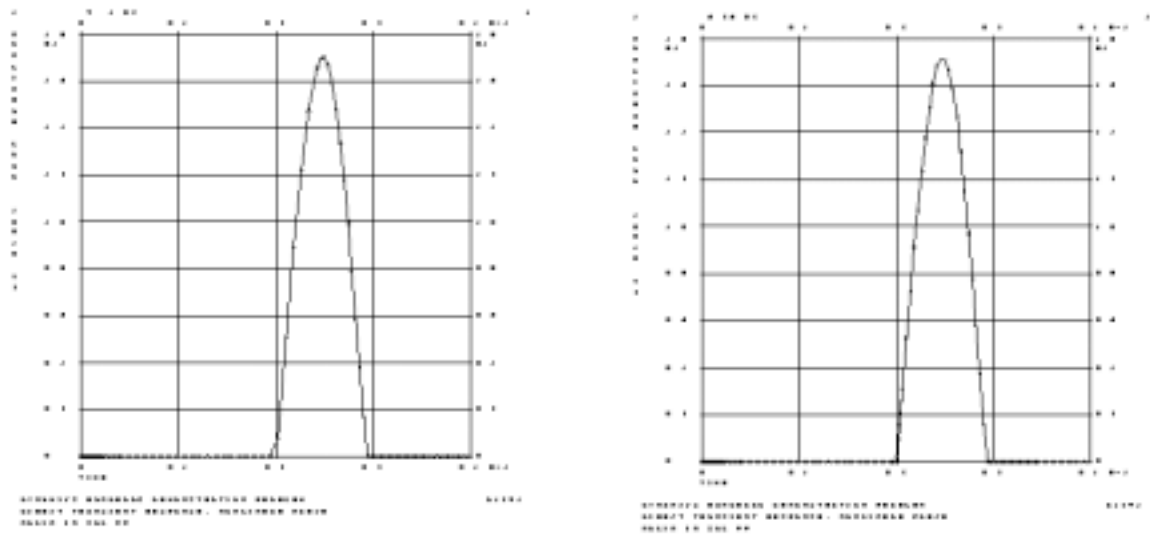


Figure 13-43 Vibrating with a Stopper (a) MSC.Nastran 2001 versus (b) MSC.Nastran 2004

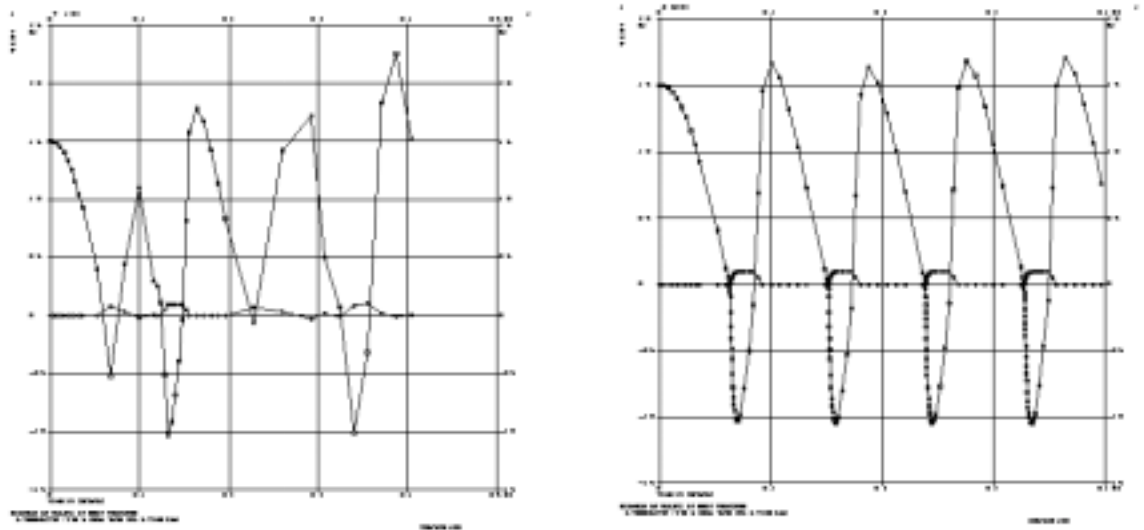


Figure 13-44 Thermal System with a Thermostat (a) MSC.Nastran 2001 versus (b) MSC.Nastran 2004

Initial Time Step Adjustment

The adaptive time stepping algorithm in the NLTRD2 module starts with a user-specified value (DT field in the TSTEPNL entry). The algorithm estimates a dominant frequency in the system, from which an appropriate time step size is computed.

However, this process requires two converged time steps. If the process continues without reverting back to time zero, the solution could be erroneous for the first two steps with a large DT. This is particularly true if you specify excessively (two orders of magnitude) large DT as demonstrated in some heat transfer problems.

An initial quadri-section algorithm has been implemented in MSC.Nastran 2004 to remedy this deficiency. This initial quadri-section is repeated until a right time step size is attained, which is controlled by an integer system cell 373 called TZEROMAX, i.e.,

TZEROMAX > 0 : maximum number of times to return to time zero

TZEROMAX = 0 : no initial time step adjustment (identical to V2001)

TZEROMAX < 0 : no limit on DT adjustment

with a default value of TZEROMAX=4. When the initial time step adjustment is activated, a UWM (7600) is issued as:

INITIAL TIME STEP SIZE IS REDUCED TO X.XXXE-X.

This new algorithm is applicable to both structural and heat transfer analyses, but especially effective for heat transfer analysis. It allows the adjustment of user-specified value of DT by almost three orders of magnitude by default. This initial time step adjustment algorithm accommodates the existing automatic time stepping capability as it currently works. However, the parameter MAXR (in TSTEPNL entry) has been modified to accept any real number without a bound, with a default value of 32.

An example of a thermal conduction process is analyzed with/without initial time step adjustment starting with DT=1.0e-4. The solution without initial time step adjustment shows gross error at the beginning as shown in [Figure 13-45](#). The initial time step adjustment algorithm adjust DT to 1.56e-6 to render the second solution.

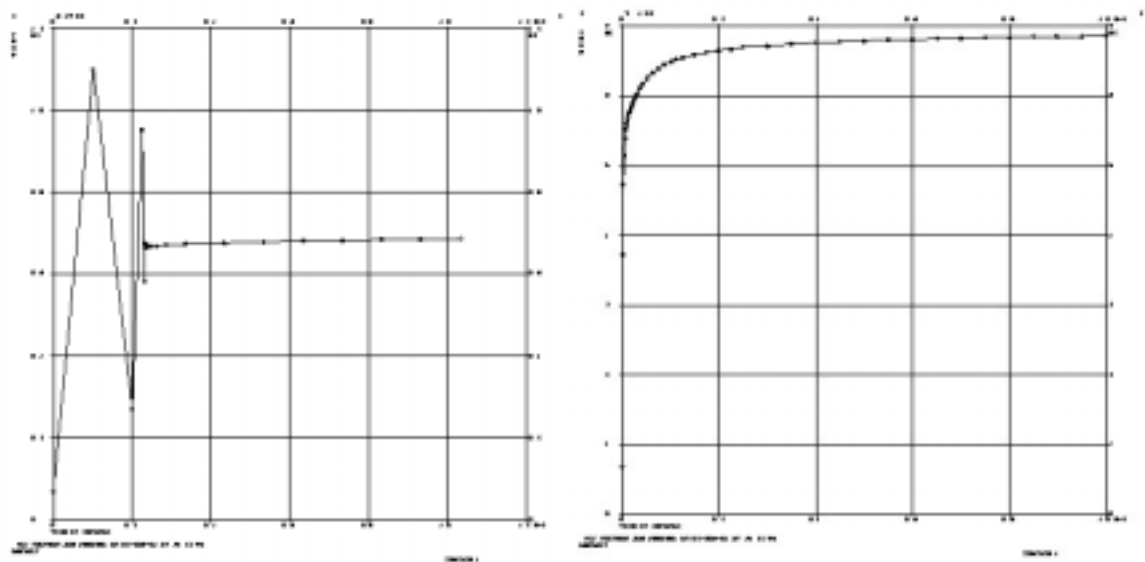


Figure 13-45 Temperature as Function of (a) MSC.Nastran 2001 Versus (b) MSC.Nastran 2004

Enhancements For Automatic Time Stepping Due to Error Corrections

Since the automatic time stepping and adaptive bisection/doubling were introduced in Version 67, some errors were identified and/or introduced. Extensive cleaning up on the time stepping routines improved automatic time stepping, adaptive bisection, and doubling scheme. These enhancements resulted in improved accuracy and efficiency. Some analyses show more accurate solution, while some converges now, which diverged before. In order to measure the efficiency, the total number of time steps taken for each Subcase is printed at the end of the Subcase along with LOOPID.

During the enhancement project, an error in PLOADX1 follower force computation has been corrected. The most prominent error discovered was in the double-precision table lookup. The double-precision table lookup is used in NOLIN processes, some of which exhibited erratic numerical behavior that could not be traced because the anomaly shows up after thousands of time steps. The anomaly is manifested by consecutive bisections that make the time step tiny in the middle of the run. The new double-precision table lookup routine is deemed to have corrected this erratic behavior. Also an error in the temperature-dependent thermal conductivity (MATT4) for transient heat transfer analysis has been corrected. In addition, a new PARAM, SEKD has been introduced in the DMAP to control the differential stiffness generation for superelements.

In **Figure 13-46**, the example shows dramatic improvement in the solution for a phase transformation process, melting a solid.

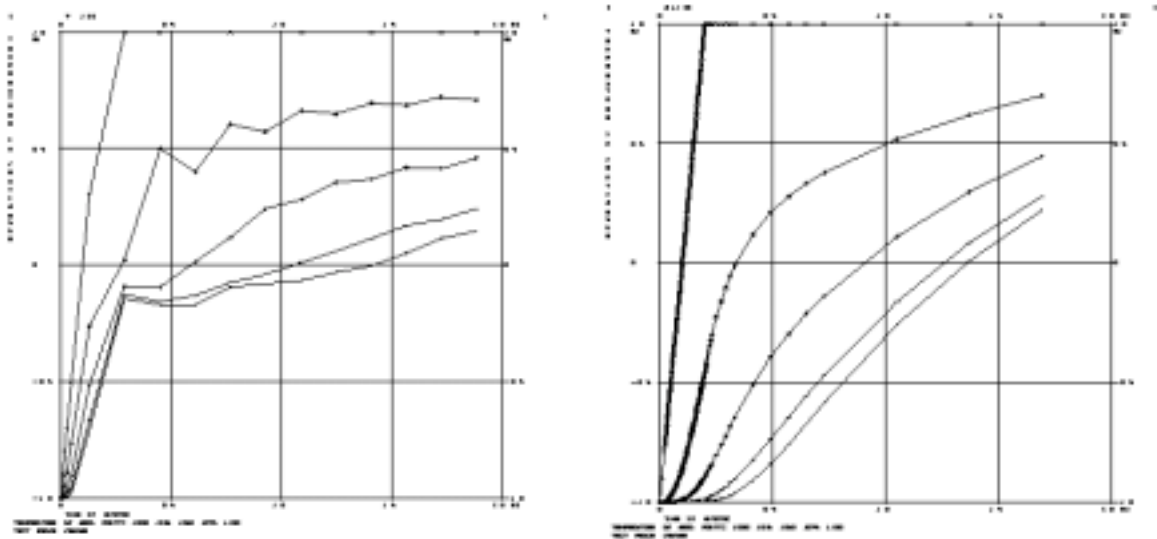


Figure 13-46 Phase Transformation (a) MSC.Nastran 2001 Versus (b) MSC.Nastran 2004

13.3.23 Nonlinear Analysis With Superelements

In nonlinear analysis with superelements, the nonlinear model must be in the residual model (Superelement = 0). All basic superelement rules apply with the exception that a nonlinear element can not have any of its degrees-of-freedom in the “o-set”.

The unique feature pertaining to load application for superelements are listed in “**3-D Slideline Contact Capability**” on page 639 under “Load Combination”.

13.3.24 Grid Point Force Balance and Element Strain Energy in Nonlinear Static Analysis

In nonlinear analysis, the strain energy, E , for an element is defined by integrating the specific energy rate, the inner product of strain rate and stress, over element volume and time

$$E = \int_0^t \int_V \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} dV d\tau \quad \text{Eq. 13-32}$$

where:

σ = stress tensor

ε = strain rate

V = element volume

t = actual time in the load history

The integration over time leads to the following recursive formula using the trapezoidal rule

$$E_{n+1} = E_n + \Delta E_{n+1} = E_n + \frac{1}{2} \int \Delta \varepsilon_{n+1}^T (\sigma_n + \sigma_{n+1}) dV \quad \text{Eq. 13-33}$$

where:

$n + 1$ = current load step

n = previous load step

ΔE = strain energy increment

$\Delta \varepsilon$ = strain-increment

The steps n and $n + 1$ are converged solution steps.

By integrating [Eq. 13-33](#) over the element volume, we get

$$E_{n+1} = E_n + \frac{1}{2} \Delta u_{n+1}^T (f_n + f_{n+1}) \quad \text{Eq. 13-34}$$

where:

f = internal element forces

Δu = displacement increment from n to $n + 1$

For computational convenience, MSC.Nastran uses [Eq. 13-34](#) to calculate the element strain energy. The internal element forces are readily available in every step because they are needed for the force equilibrium. Note that temperature effects are included in the internal element forces.

When loads from temperature differences or element deformation are present, the default definition of element strain energy for linear elements differ from the definition for nonlinear elements. For linear elements, the element strain energy is defined as

$$E = \frac{1}{2}u^T K_e u - u^T P_{et} \quad \text{Eq. 13-35}$$

where P_{et} is the element load vector for temperature loads and element deformation. **Eq. 13-35** assumes that the temperatures are constant within a subcase. For nonlinear elements, the definition of **Eq. 13-32** is used. In the case of linear material and geometry, **Eq. 13-32** becomes

$$E = \frac{1}{2}u^T K_e u - \frac{1}{2}u^T P_{et} \quad \text{Eq. 13-36}$$

Eq. 13-36 assumes that the temperature varies linearly within a subcase. The user may request the definition of **Eq. 13-36** to be applied to linear elements by adding `PARAM,XFLAG,2` to the input file.

User Input

The output of grid point force balance and strain energy in nonlinear analysis is requested with the existing `GPFORCE` and `ESE` Case Control commands, respectively. The Case Control commands remain unchanged.

Printed Output

The output is the same as in linear analysis. An example appears below.

Example

The following MSC.Nastran input file represents a simplified model of a mechanical clutch that consists of springs, beams, rigid elements, and gap elements. A geometric nonlinear analysis is performed. Both `GPFORCE` and `ESE` output requests are applied above all subcases. This is a good example to show the grid point force balance with both linear and nonlinear elements, that includes:

- applied loads
- element forces
- SPC forces
- MPC forces.

The model is shown in **Figure 13-47**. For clarity, only the elements are displayed.

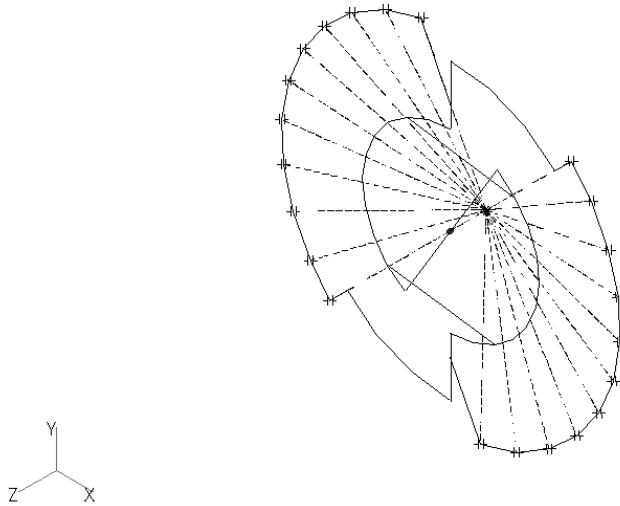


Figure 13-47

The dashed lines represent rigid elements, the rest are gap and beam elements. Spring elements are located at the intersection point of all dashed lines and are not visible in the figure because their connection points are coincident.

Grid Point Force Output

1 GPF005B - FORD ELETRO-MECHANICAL CLUTCH			SEPTEMBER 26, 2000 MSC.NASTRAN 9/25/00 PAGE 248						
DISPLACEMENT IN MM., FORCE IN NEWTON, STRESS IN MPA			SUBCASE 2						
0 LOAD STEP = 2.00000E+00			GRID POINT FORCE BALANCE						
POINT-ID	ELEMENT-ID	SOURCE	T1	T2	T3	R1	R2	R3	
1		APP-LOAD	.0	.0	.0	.0	.0	-2.400000E+02	
1		F-OF-SPC	.0	.0	.0	.0	.0	2.400000E+02	
1	10001	BEAM	.0	.0	.0	.0	.0	.0	
1	10002	BEAM	.0	.0	.0	.0	.0	.0	
1		*TOTALS*	.0	.0	.0	.0	.0	.0	
0	100	F-OF-SPC	-5.366200E-25	-4.514879E-24	.0	4.535559E-23	-6.400176E-24	.0	
100		F-OF-MPC	5.366200E-25	4.514879E-24	.0	-4.535559E-23	6.400176E-24	1.292178E-24	
100	3	ELAS2	.0	.0	.0	.0	.0	1.070983E-24	
100		*TOTALS*	.0	.0	.0	.0	.0	2.363161E-24	
0	101	F-OF-SPC	.0	.0	-3.078859E-28	-3.549490E-26	2.535672E-24	.0	
101		BEAM	-3.701448E-08	2.907432E-07	.0	.0	.0	-5.474456E-09	
101	2004	BEAM	-3.888788E-07	-1.144153E-07	3.078859E-28	3.549490E-26	-2.535672E-24	-1.777338E-23	
101	1	GAP	1.270000E-10	1.438901E-27	.0	.0	.0	.0	
101		*TOTALS*	-4.257663E-07	1.763280E-07	.0	.0	.0	-5.474456E-09	
0	102	101 BEAM	-2.715682E-08	-2.918291E-07	.0	.0	.0	-2.549396E-09	
102	102 BEAM	-1.637746E-08	1.273060E-07	.0	.0	.0	.0	-1.374512E-25	
102	2 GAP	1.270000E-10	-9.144001E-27	.0	.0	.0	.0	.0	

Element Strain Energy Output

```

1  GPF005B - FORD ELETRO-MECHANICAL CLUTCH          SEPTEMBER 26, 2000 MSC.NASTRAN 9/25/00 PAGE 231
DISPLACEMENT IN MM., FORCE IN NEWTON, STRESS IN MPA
0
LOAD STEP = 1.00000E+00                                SUBCASE 1
                                E L E M E N T   S T R A I N   E N E R G I E S
                                * TOTAL ENERGY OF ALL ELEMENTS IN PROBLEM = 4.422530E+02
ELEMENT-TYPE = BEAM                                * TOTAL ENERGY OF ALL ELEMENTS IN SET -1 = 4.422530E+02
SUBCASE 1
0
ELEMENT-ID      STRAIN-ENERGY      PERCENT OF TOTAL      STRAIN-ENERGY-DENSITY
101             1.211380E+01             2.6928                5.680606E-02
102             1.049874E+00             .2334                4.923244E-03
103             1.323984E-01             .0294                6.208644E-04
104             1.323428E-01             .0294                6.206040E-04
105             1.374425E-01             .0306                6.445181E-04
106             1.407006E-01             .0313                6.597968E-04
107             1.478413E-01             .0329                6.932822E-04
108             1.483829E-01             .0330                6.958218E-04
109             1.032130E-01             .0229                1.936014E-04
110             3.163312E-01             .0703                7.020313E-04
111             4.605607E-01             .1024                1.927717E-03
112             7.067609E-01             .1571                2.958209E-03
113             4.186914E-01             .0931                1.752469E-03
121             3.068841E+00             .6822                2.905759E-02

```

Remarks

- Linear and nonlinear elements can be mixed. The output for grid point force and element strain energy in nonlinear analysis follow the same format as in linear analysis.
- The reference system for the grid point force output is the grid point global coordinate system. The user cannot specify another output coordinate system. In nonlinear, the grid point forces are not aligned with element edges. Therefore, PARAM,NOELOF and PARAM,NOELOP are ignored in nonlinear analysis.
- In nonlinear analysis, the element strain energy must be calculated for each intermediate load step even if the output is requested only in the last load step. To save computations, the element strain energy is only calculated upon user request. The Case Control commands, GPFORCE or ESE must be present to activate grid point force output or element strain energy calculations and output. If the user wants to save the grid point forces and element strain energy on the database and wants an output later in an eventual restart run, the Case Control commands, GPFORCE(PLOT) or ESE(PLOT), must be specified in the cold start run.
- By default, if GPFORCE or ESE is present, then the output of grid point forces or element strain energy is only available at the end of each subcase. For output of intermediate steps, the parameter INTOUT on the NLPARAM Bulk Data entry must be set to YES or ALL.

- For a model with superelements, the grid point force balance is not in equilibrium along the boundary of the residual and upstream superelements. This is because the applied load in the residual does not include the load contributed from upstream superelements when the grid point force balance is computed. The same situation also occurs in each upstream superelement whose applied load does not include the contribution from other superelements and the residual. When the total of the grid force balances of all superelements is added together, equilibrium is reached.

For example, grid point 123 is located at the boundary between the residual (superelement ID = 0) and an upstream superelement (ID = 10) and is connected to two QUAD4 elements (ID 22 and 21). The output of the grid point force balance is shown below.

1	GPF006A - NONLINEAR GPFORCE TEST PROBLEM			SEPTEMBER 26, 2000		MSC.NASTRAN	9/26/00	PAGE	26
	Q4 ELEMENTS + MULTIPLE S.E.'S							SUPERELEMENT 0	
0	NONLINEARITY ANALYSIS - AUTO/NLPCI + INTOUT=YES							SUBCASE 1	
	LOAD STEP = 1.00000E+00								
			GRID POINT FORCE BALANCE						
POINT-ID	ELEMENT-ID	SOURCE	T1	T2	T3	R1	R2	R3	
0	123	APP-LOAD	-3.287216E-01	1.006007E-02	-5.592173E-03	.0	.0	.0	
	123	F-OF-SPC	.0	.0	5.127304E+00	-2.287795E-04	-1.193804E-02	.0	
	123	QUAD4	-6.765498E-01	8.791827E+00	-2.111628E+00	8.897618E-04	1.142736E-02	-3.166363E-02	
	123	*TOTALS*	-1.005271E+00	8.801888E+00	3.010083E+00	6.609823E-04	-5.106755E-04	-3.166363E-02	
1	GPF006A - NONLINEAR GPFORCE TEST PROBLEM			SEPTEMBER 26, 2000		MSC.NASTRAN	9/26/00	PAGE	33
	Q4 ELEMENTS + MULTIPLE S.E.'S							SUPERELEMENT 10	
0	NONLINEARITY ANALYSIS - AUTO/NLPCI + INTOUT=YES							SUBCASE 1	
	LOAD STEP = 1.00000E+00								
			GRID POINT FORCE BALANCE						
POINT-ID	ELEMENT-ID	SOURCE	T1	T2	T3	R1	R2	R3	
	123	22 QUAD4	1.005275E+00	-8.801861E+00	-3.010083E+00	-6.609823E-04	5.106755E-04	3.166376E-02	
	123	*TOTALS*	1.005275E+00	-8.801861E+00	-3.010083E+00	-6.609823E-04	5.106755E-04	3.166376E-02	

Obviously, neither the residual nor the upstream superelement has the balanced grid point force. However, if both sides are added together, the balance will be reached.

- Computational time and memory requirement increase when grid point force balance or element strain energy output is requested. For a large nonlinear model, allocation of sufficient memory is important. Spill logic is built in the code, but it may increase the overall computational time.

13.4 Prerelease of the General Nonlinear Solution Sequence SOL 400

Introduction

In order to improve the nonlinear solution procedure in MSC.Nastran, a general nonlinear solution sequence SOL 400 is introduced. This solution sequence will include all nonlinear analyses types, such as nonlinear static analysis, nonlinear transient analysis, nonlinear buckling analysis, and nonlinear normal modes analysis. SOL 400 also supports linear transient analysis. Eventually, SOL 400 is intended to replace the current nonlinear static solution sequence, SOL 106, and the nonlinear transient solution sequence SOL 129. In future releases, the nonlinear heat transfer solution sequences, SOL 153 and SOL 159 will be included in SOL 400. In addition, it is planned to include 3D surface contact with the same text interface as SOL 600. The prerelease is intended to get user feedback. In the current stage, SOL 400 does not replace SOL 106 and 129.

Benefits

- A new SUBCASE/STEP combination allows the user to perform nonlinear analysis similar to the existing methods plus the option to include independent loadings in a single run.
- When fully implemented, all nonlinear solution procedures, such as linear static analysis, nonlinear static analysis, transient nonlinear analysis, etc., can be performed in a single run. For example, you can run a transient analysis using the nonlinear static analysis results of a previous step of the initial condition.
- An improved nonlinear iteration procedure, which speeds up convergence.
- Allows the simulation of existing nonlinear solution sequences such as SOL 106 or SOL 129.
- The new Lagrange rigid elements allow correct solution of the rigid elements such as RBAR in a geometric nonlinear analysis.
- A new large rotation representation method, the right rotation method, to be used in a geometric nonlinear analysis.
- A comprehensive Nonlinear Iteration Summary Table allowed user to track the solution status easily.
- A new user friendly restart procedure.

- A more flexible output method allowing intermediate results output during the analysis.

Limitations for the Current Release

Because SOL 400 is a pre-release, the following capabilities are not supported:

- Nonlinear transient analysis.
- Nonlinear normal modes and nonlinear buckling analysis.
- RFORCE and Creep.
- Arc-length method (input by NLPCI Bulk Data entry).
- The GPFORCE and ESE output requests.
- Slide line contact.

The first 5 items will be supported in future release. The last item 'line contact' will be replaced by a more general contact capability with the same text input as SOL 600.

Features supported by SOL 400 in the current release are discussed in the following section.

Analysis Steps

The sequence of load application and analysis method is important in determining the final response of a nonlinear structure. On the other hand, it is convenient for the user to have the ability to perform analyses of multiple independent cases in a single run. For this reason, a new Case Control command, STEP, is created exclusively for the new nonlinear solution sequence SOL 400. This command provides a mechanism for defining the multiple load steps and other characteristics for a SUBCASE in a sequential manner, and, at same time, allow the user to run multiple independent cases in a single run. The format of this Case Control command is:

STEP n

where n is the step identification number. The usage of the SUBCASE and STEP Case Control commands are shown below.

- The simplest input contains one SUBCASE and one STEP. Either or both the SUBCASE and STEP commands may be omitted from the input. The default value for both SUBCASE and STEP is 1.

- With one SUBCASE and multiple steps, the procedure is only slightly more complicated than previous example. Each step defines the total new external load and other characteristics for the step, which will be applied by the completion of the step. The solution of any STEP is a continuation of the solution of its previous STEP. STEP identification numbers must be in ascending order. The following is a typical example:

```

SUBCASE 1
NLPARAM = 200
  STEP 10
  LOAD = 10
  STEP 20
  LOAD = 20
  STEP 30
  LOAD = 30

```

- Multiple SUBCASEs may be executed in a single analysis where loads and boundary conditions may change. All SUBCASEs are independent of each other, i.e., no load history information is transmitted from one SUBCASE to the next. At the start of each SUBCASE, the deflections, stresses and strains throughout the model are zero. In each SUBCASE, there may be a number of STEPs. For example

```

NLPARAM = 200
SUBCASE 1
  STEP 10
  LOAD = 10
  STEP 20
  LOAD = 20
SUBCASE 2
  STEP 100
  LOAD = 100

```

In the previous example, the solutions for SUBCASE 1 and SUBCASE 2 are independent of each other. If the solution diverges in a particular step, MSC.Nastran will terminate the solution of the current subcase and proceed to the next subcase if it exits.

- A case control command placed below the step level allows that command to vary from on step to another. If it is placed above the step level, the command remains constant for all steps in the subcase. Most of the case control commands, which can be placed below the subcase level, can also be placed below the step level. For example, all steps in above examples use the same Case Control command, NLPARAM = 200.
- The meaning of multiple SUBCASEs **without** STEP is dependent on the system cell NASTRAN SYSTEM (366) as follows:

0 - The solutions of all SUBCASEs are independent of each other. This is consistent with the new SOL 400 procedure. MSC.Nastran will keep all SUBCASE commands in the Case Control file and insert internally a "STEP 1" for each SUBCASE. This is a default value.

1 - The solution of each SUBCASE is a continuation of the previous SUBCASE. This is similar to the solution sequence SOL 106 procedure. MSC.Nastran internally converts all the SUBCASE identification number to STEP identification numbers and insert a "SUBCASE 1" before the first STEP.

The default is 0.

Linear Static Analysis and Nonlinear Static Analysis

The SOL 400 uses an enhanced dynamic solution algorithm, which makes the linear static solution and the nonlinear static solution become special cases of the general nonlinear solution procedure. For this release, only the linear static analysis and the nonlinear static analysis are supported.

The linear static analysis and the nonlinear static analysis are requested by the Case Control command, ANALYSIS. For example:

```
SUBCASE 10
  STEP 1
    ANALYSIS = LNSTATIC
    LOAD = 10
  STEP 2
    ANALYSIS = NLSTATIC
    LOAD = 20
    NLPARM = 20
```

In the previous example, SUBCASE 10 has two steps: the first step requests a linear static analysis and the second step requests a nonlinear static analysis. The default ANALYSIS method, i.e., ANALYSIS command in the Case Control file, is NLSTATIC.

Vector Operations and Convergence Criteria

The convergence criteria are specified by using the Bulk Data entry, NLPARM. In performing the convergence tests, we compute three error factors: the displacement, the load, and the work (energy) error factors, which are printed in the “**Nonlinear Iteration Summary Table**” on page 726. These three error factors must satisfy the error tolerance rules specified by CONV, EPSU, EPSP, and EPSW on the NLPARM Bulk Data entry.

In computing the error factors, SOL 106 used the l-set (the leftover set) vectors for displacements and forces. By using this method, the effect of SPC loads and MPC constraints are accounted for only indirectly. Also, there are difficulties accounting for

the effect of Lagrange multipliers for the Lagrange rigid elements, which are discussed below. For these reasons, in SOL 400, whenever possible, the matrix and vector operations, which include the computations of error factors, are performed in p-set (the physical set). For MSC.Nastran set definition, please refer to the “[Degree-of-Freedom Set Definitions](#)” on page 844 of the *MSC.Nastran Quick Reference Guide*.

Another major modification is the computation of the work error. In SOL 106, the work error is based on the multiplication of the residual force and the displacement change. During iteration, both the residual force and the displacement change become smaller; therefore, the convergence rate of this value is proportional to the square of the convergence rate of the solution. Thus it can become very small near convergence. Also, it does not have a counter part in the physical world. In SOL 400, the total work done to structure model is computed during iterations and the work error is estimated based on the total work. Using this method, the work error gives an estimation of error in actual work done to the structural model. The total work for each iteration is printed on the “[Nonlinear Iteration Summary Table](#)” on page 726. Note, however, that this total work is only an approximation.

Solution Algorithm and Simulation of SOL 106

In SOL 400, the solution algorithm is modified substantially in the following areas:

- The nonlinear iterations and stiffness update algorithm for the AUTO and SEMI of the KMETHOD on the NLPARM Bulk Data entry. The action of system 401 is as follows:
- The algorithm for load bisections.
- The solution divergence processing.
- For most of the problems tested, SOL 400 yields equal or better performance than that of SOL 106. However, in order to provide an option similar to that of SOL 106, system cell ITRFMT (SYSTEM (401)) can be used.
- 0 – Uses the SOL 400 algorithm. This is the default value.
- -2 – Uses algorithm similar to the SOL 106 algorithm.

The default is 0. Note that even with ITRFMT=-2, the iteration solution sequence of SOL 400 will not be the same as that of SOL 106. One reason is that the vector operations in SOL 400 use the p-set, while SOL 106 use the l-set. See the section on “[Vector Operations and Convergence Criteria](#)” on page 724 for further details.

Lagrange Rigid Elements and the Right Rotation Method

The linear rigid elements such as RBAR, RBE1, etc., have been available in MSC.Nastran since the inception of MSC.Nastran. However, these rigid elements use a small rotation theory, which may give erroneous solution in a geometrical nonlinear analysis.

In the current release, a new type of rigid elements, the Lagrange rigid elements, is implemented. For the nonlinear analysis, the Lagrange rigid element is available for SOL 400 only. The Lagrange rigid elements use the large rotation theory in a geometric nonlinear analysis (PARAM, LGDISP, 1) so that the nonlinear solution is correctly computed. Also, for the large rotation theory, a new rotation representation method, the right rotation method, is implemented. For the Lagrange rigid elements and the right rotation method, please refer to “[Rigid Element Enhancements](#)” on page 243.

Nonlinear Iteration Summary Table

In order to allow the user to track the solution sequence during the nonlinear iteration, a detailed Nonlinear Iteration Summary Table is generated. A line for each iteration is generated in the f06 file during the nonlinear iteration. Due to printing of the average and the maximum displacements, you can see the solution status before the end of the solution. This can be useful for large nonlinear problems. However, even for small problems, you can examine the table to observe the progress of the analysis. An example of this table is given below and the descriptions of information given in this table are shown in [Table 13-16](#).

NON - LINEAR ITERATION MODULE OUTPUT																
STIFFNESS UPDATE TIME 0.01 SECONDS															SUBCASE	
STEP 1																
ITERATION TIME 0.02 SECONDS																
LOAD	NO.	-- ERROR FACTORS --			CONV	ITR	MAT	NO.	AVG	TOTL	----- DISP -----					
LINE_S	NO.	TOT	TOT													
STEP	INC	ITR	DISP	LOAD	WORK	RATE	DIV	DIV	BIS	R_FORCE	WORK	AVG	MAX	AT GRID	C	FACT NO QNV KUD ITR
0.1000	1	1	1.00E+00	1.62E+02	1.62E+02	1.000	0	1	0	1.56E+02	9.852E-01	1.32E-02	9.800E-02			8

Table 13-16 Description of the Nonlinear Iteration Summary Table

LOAD STEP	The percent of the current step + (step no. - 1). For example, 0.1 means 10% of the first step and 1.337 means 33.7% of the second step.
NO. INC	Number of increment. The total number of load increment for a step is requested by NINC on the NLPARM Bulk Data entry.

Table 13-16 Description of the Nonlinear Iteration Summary Table (continued)

ERROR FACTORS: DISP LOAD WORK	There are three error factors: displacement, load, and works. In order for an increment to converge, these factors must satisfy the error tolerance rules specified by CONV, EPSU, EPSP, and EPSW on the NLPARM Bulk Data entry.
CONV RATE	Converge rate, which denote how fast the solution converges for the current increment. A value of 0.0 means fast converges and a value > 1.0 means that the solution will never converge.
ITR DIV	Number of iteration divergence. Action to correct solution divergence will be taken if ITRDIV $>$ MAXDIV.
MAT DIV	Number of material divergence + 1, i.e., it will be 1 if there is no material divergence. The material divergence is due to bad creep strain or excessive sub-increments in plasticity.
NO. BIS	Number of bisection performed.
AVG R_FORCE	Average residual force. In order for an increment to converge, this value must become very small.
TOTAL WORK	Accumulated total work done to the structure model. This value is only an approximation.
DISP: AVG MAX AT GRID C	The average displacement, the maximum displacement and its grid point identification number and component number.
LINE_S: FACT NO	Line search factor and number of line searches performed.
NO. QNV	Number of quasi-Newton vectors computed.
TOT KUD	Total number of stiffness updates performed
TOT ITER	Total number iterations performed, including the number of stiffness updates.

Restart

The purpose of a nonlinear restart is to allow the user to use the material or the geometrical properties of a previously converged solution as a new starting point to continue the analysis. This is useful when the user want to change the loading sequence, the solution criteria, or to extend the analysis.

For SOL 400, a user-friendly restart procedure has been implemented. For this restart procedure, the following principles are noted:

- The restart must be continued at a previously converged solution point by specifying a SUBCASE, STEP, and/or load factor. This is accomplished by using the Case Control command, NLRESTART.
- The geometry and the initial material properties of the structural model cannot be modified. This is obvious because any modification to the geometry or the initial material properties would invalidate the previous analysis and require the nonlinear solution to start from the very beginning. In such cases, it is simpler to initiate another cold start run.

Procedures for performing the restart are described in the following subsections.

File Management Statements

For a restart, the data of the cold start must be made available by using the File Management statements. For nonlinear restart, two statements are needed: ASSIGN and RESTART. These two statements are existing restart statements and no special requirements are needed for SOL 400.

There are many methods to fetch data for a restart. One method is given in the example below. For other methods, please refer to the “[The File Management Section \(FMS\)](#)” on page 33 of the *MSC.Nastran Quick Reference Guide* or “[Interface With Other Programs](#)” on page 515 of the *MSC.Nastran Reference Guide*.

Case Control Modifications

The presence of a Case Control command NLRESTART indicates that the current run is a restart run. The Case Control file contains **both** subcases and steps that were included in the cold start run, **and** those that are to be added in the restart run. The first subcase, step and/or load factor to be executed in the restart is indicated by the options on the NLRESTART command. The is shown by the following example:

```
NLRESTART SUBCASE 1, STEP 2, LOADFAC 0.3
SUBCASE 1
    STEP 1
        LOAD = 10
    STEP 2
```

```

        LOAD = 20
STEP 3
        LOAD = 30

```

In the above example, the first step through load factor 0.3 of the second step has been previously executed. The restart execution begins with load factor 0.3 of the second step, and continues through the end of the third step. For restart, the case control file structure for SUBCASE and STEP commands must be the same as the cold start run up to the restart point. After the restart point, the user may modify or add Case Control SUBCASE and STEP commands. For example, in above example, steps 1 and 2 must exist in the cold start. However, step 3 may or may not exist in the cold start run.

The following Case Control commands may be modified in a nonlinear restart:

- Boundary conditions such as MPC and SPC.
- Nonlinear solution control, NLPARM.
- LOAD requests.
- Output request such as DISP and NLSTRESS.
- The analysis type (ANALYSIS).

Depending on the option selected with the NLRESTART command, the nonlinear restart may be logically divided into three types: a case restart, a step restart, or a load factor restart:

- The case restart begins the execution with a SUBCASE command. All five types of modification described above are acceptable for a case restart.
- The step restart begins the execution with a STEP command, which may be a new step or a previously executed step. Although boundary condition and analysis type modifications are allowed, the user has the responsibility to determine whether they are meaningful. Special attention should be given to the **analysis type** modification, it may not be meaningful in many situations and, thus, may lead to erroneous results.
- The load factor restart begins execution with a LOADFAC command. For a load factor restart, user may not modify analysis type, boundary conditions, and load requests. User needs to exercise discretion when attempting other types of modification at this level. Also, in order to perform this type of restart, the INOUT=YES must be set on the NLPARM Bulk Data entry on a cold start run so that the required data are available for restart.

Bulk Data Modifications

The Bulk Data file for a nonlinear restart contains only those entries which are to be added to the cold start. The Bulk Data deletion entry ‘/’ should not be used. This is to serve as a reminder that the geometry and the initial properties cannot be modified. User may make modifications to the Bulk Data file by introducing new entries which may be copies of the original entries with appropriate changes and new identification numbers. The following is list of entries that can be added in a restart run:

- Load entries such as LOAD, FORCE, PLOAD4, and SPCD.
- NLPARAM entries.
- Boundary condition entries such as SPC, SPC1, and MPC.

Output

The output is requested by using the appropriate Case Control commands--e.g., DISP, STRESS, etc.

Two output are special for SOL 400. One is the Nonlinear Iteration Summary Table, which will allow the user to monitor the status of the nonlinear solution sequence. This table is discussed in Description of the “[Nonlinear Iteration Summary Table](#)” on page 726.

Another one is the PARAM, PH2OUT. In MSC.Nastran, the nonlinear analysis is divided into three phases. In phase I, the upstream superelements and geometrical data of the residue structure are processed. In phase II, the solution of the residue structure is obtained. And, in phase III, data are recovered for both the residual structure and upstream superelements. The output requests are processed in phase III. In many situations, the solution terminates prematurely in phase II and the user is unable to obtain any output for debugging the model. In order to remedy this deficiency, the parameter PH2OUT is for requesting phase II output. The phase II output consist of all output requested by Case Control commands for the residual structure only and are printed in sort 1 format. The parameter PH2OUT has following options:

- 0 -- Output phase III outputs only. In this case, both outputs for residual structure and upstream superelements are processed. This is the default value.
- 1 -- Output phase II outputs only. In this case, no output is processed for upstream superelements.
- 2 -- Output both phase II and phase III output. In this case, the output for the residual structure may be redundant.

If there is an abnormal termination in phase II, user can rerun the job with phase II output request for debugging.

User's Interfaces

The following section summarizes the User's Interface for SOL 400, which are important to the nonlinear analysis and can be used in SOL 400. Refer to the *MSC.Nastran Quick Reference Guide* for further details.

Nastran System Cells

- STPFLG (SYSTEM (366)) – Selects the SUBCASE or STEP methodology when there are a number of SUBCASE commands and no STEP command in the Case Control file, see “[Analysis Steps](#)” on page 722.
- ITRFMT (SYSTEM (401)) – Selects the convergence parameter computation method and the divergence solution checking method, see section “[Solution Algorithm and Simulation of SOL 106](#)” on page 725 of SOL 106.

File Management Statements

The following File Management statements are required for restarts. Please refer to the “[The File Management Section \(FMS\)](#)” on page 33 of the *MSC.Nastran Quick Reference Guide* or Chapter 12 of the “[Interface With Other Programs](#)” on page 515 of the *MSC.Nastran Reference Guide*.

- ASSIGN -- Assigns database physical file names used by MSC.Nastran.
- RESTART -- Requests that data stored in a previous run be used in the current run.

Executive Control Statements

- SOL 400 or SOL NONLIN -- Requests the SOL 400 general nonlinear solution sequence

Parameters

- PARAM, LANGLE -- Selects the method to represent large rotations in a geometric nonlinear analysis -- 1 for the Gimbals angle method, 2 for the left rotation method, and 3 for the right rotation method. For detailed descriptions, refer to User's Interface section in Rigid Elements of the current Release Guide. The default value is 2 for the current release and will become 3 for a future release.
- PARAM, LGDISP -- Requests a geometric nonlinear analysis.

- PARAM, FOLLOWK -- Requests whether the follower force stiffness will be used in a geometric nonlinear analysis.
- PARAM, FKSVMFAC -- Controls whether the symmetrical follower force stiffness will be used in a geometric nonlinear analysis.
- PARAM, MAXLP – Specifies maximum number of iterations for element relaxation and material point sub-increment process.
- PARAM, NLAYERS – Specifies the number of layer for through thickness integration in the material nonlinear analysis.
- PARAM, NLTOL – Selects defaults for CONV, EPSU, EPSP, and EPSW for the NLPARM Bulk Data entry.
- PARAM, PH2OUT – Requests phase II or phase III output for a nonlinear analysis, see “[Output](#)” on page 730.

Case Control Commands

- ANALYSIS – Selects solution method for an analysis step, see “[Linear Static Analysis and Nonlinear Static Analysis](#)” on page 724.
- NLRESTART – Requests a restart execution at a specific solution point for SOL 400, see “[Restart](#)” on page 728.
- NLSTRESS – Requests the form and type of the nonlinear element stress output.
- STEP – Delimits and identifies an analysis step, see “[Analysis Steps](#)” on page 722.

Bulk Data Entries

- MATHP- Specifies the hyperelastic material properties for an element.
- MATS1 – Specifies the stress-dependent material properties for an element.
- NLPARM – Defines a set of parameters for nonlinear static analysis iteration strategy.

Examples

The following two examples show the input for a cold start run and a restart run of the nonlinear static analysis. The intention of these examples is to show the input structure for SOL 400. The model itself and parameters specified on the NLPARM Bulk Data entry are not important.

Example one, NLSTS5, is a cold start run. This model only contains a single HEXA element and one QUAD4 element. It has both material nonlinearity (MATS1) and geometrical nonlinearity (PARAM, LGDISP, 1). All the **bold**-font statements are entries pertaining to the nonlinear analysis.

When the job is completed, in addition to the regular .f04, .f06 and .log files, two database files will also be created. They are 'inlstrs5.DBALL' and 'inlstrs5.MASTER', which are needed for the restart run.

```

ID MSC, NLSTRS5
TIME 5
SOL 400
CEND
TITLE      = THE COLD START RUN OF THE NONLINEAR STATIC ANALYSIS
SUBTITLE   = ELASTIC-PLASTIC ANISOTROPIC MATERIAL
LABEL      = UNIAXIAL STRESS STATE
$ this model has 1 HEXA AND 1 QUAD4.
ECHO       = SORT
ANALYSIS = NLSTAT
DISP       = ALL
SPCFORCES = ALL
SUBCASE 101
    NLPARM = 100
    STEP 10
        LOAD = 101
    STEP 20
        LOAD = 102
    STEP 30
        LOAD = 103
SUBCASE 102
    NLPARM = 200
    STEP 10
        LOAD = 101
    STEP 20
        LOAD = 102
    STEP 30
        LOAD = 103
BEGIN BULK
$ DEFINE GRID POINT COORDINATES AND BOUNDARY CONDITIONS.
$      GRID-ID COORD-SYS      X      Y      Z      DOFS FIXED
GRID   1      0      0.0    0.0    0.0      123456
GRID   2      0      1.0    0.0    0.0      3456
GRID   3      0      1.0    1.0    0.0      3456
GRID   4      0      0.0    1.0    0.0      13456
GRID  101     0      0.0    0.0    1.0      12456
GRID  102     0      1.0    0.0    1.0      456
GRID  103     0      1.0    1.0    1.0      456
GRID  104     0      0.0    1.0    1.0      1456
$
GRID   11     0      0.    0.    0.      123456
GRID   12     0      1.    0.    0.      3456
GRID   13     0      1.    1.    0.      3456

```



```

GRID      14      0      0.      1.      0.      13456
$
$ DEFINE ELEMENT - 8 NODE ISOPARAMETRIC SOLID ELEMENT.
$      EL-ID  PROPERTY      C O N N E C T I V I T Y
CHEXA     1      100      1      2      3      4      101      102      +1
+1        103      104
$
CQUAD4   11      101      11      12      13      14
$
$ DEFINE ELEMENT GEOMETRIC PROPERTIES.
$
PSOLID   100      120
PSHELL   101      121      1.0
$
$ DEFINE MATERIAL PROPERTIES.
$
MAT9     120      7.9812E6-6.149E50.      1.5136E60.      0.      .9539E6+MT1
+MT1     0.      1.5136E60.      0.      1.E6      0.      0.      0.      +MT2
+MT2     3.0824E60.      0.      3.E5      0.      3.E5
MATS1   120      PLASTIC 5.E5      5.E3
$
MAT2     121      7.9812E6-6.149E51.5136E64.9539E61.5136E63.0824E6
MATS1   121      PLASTIC 5.E5      5.E3
$
NLPARM  100      10      ITER      1      YES
NLPARM  200      20      AUTO
PARAM   LGDISP  1
$
$ DEFINE GRID POINT STATIC LOADS BY SPECIFYING VECTORS.
$      LOAD SET GRID  COORD.      SCALE  N1      N2      N3
$      ID      ID      SYS. ID      FACTOR
FORCE   101      2      0      5.0E3  0.5      0.0      0.0
FORCE   101      3      0      5.0E3  0.5      0.0      0.0
FORCE   101      102     0      5.0E3  0.5      0.0      0.0
FORCE   101      103     0      5.0E3  0.5      0.0      0.0
$
FORCE   101      12      0      5.0E3  1.0      0.0      0.0
FORCE   101      13      0      5.0E3  1.0      0.0      0.0
$
LOAD    102      2.0      1.0      101
LOAD    103      3.0      1.0      101
$
ENDDATA

```

Example two, NLSTRS5R, is a restart run. This example continues the job from the end of the last run with one extra step (STEP 40) in SUBCASE 102. Before the new STEP 40, the Case Control file structure for SUBCASE and STEP must be the same as that of the cold start. On the other hand, the output requests such as STRESS, etc. can be modified. In this example, the **bold**-font statements are pertaining to the restart run.

Two file management statements -- ASSIGN and RESTART -- are required to fetch stored data for restart. One way is shown here. There are many other ways to obtain data for restart. Please consult the “[The File Management Section \(FMS\)](#)” on page 33 of the *MSC.Nastran Quick Reference Guide* or “[Interface With Other Programs](#)” on page 515 of the *MSC.Nastran Reference Guide*.

```

ASSIGN MASTER='nlstrs5.MASTER' $ File management statement to assign
$                               master Dbset for the current run.
RESTART $ File management statement, request data stored in
$           'nlstr5.MASTER' are used for restart.
ID MSC, NLSTRS5R
TIME 5
SOL 400
CEND
TITLE      = THE RESTART START RUN OF THE NONLINEAR STATIC ANALYSIS
SUBTITLE   = ELASTIC-PLASTIC ANISOTROPIC MATERIAL
LABEL      = UNIAXIAL STRESS STATE
ECHO       = SORT
ANALYSIS   = NLSTAT
DISP       = ALL
SPCFORCES  = ALL
STRESS    = ALL
ELFORCE  = ALL
OLOAD    = ALL
NLRESTART $   Since there is nothing specified in this command,
$             job will restart from the end of the previous run.
SUBCASE 101
    NLPARM = 100
    STEP 10
        LOAD = 101
    STEP 20
        LOAD = 102
    STEP 30
        LOAD = 103
SUBCASE 102
    NLPARM = 200
    STEP 10
        LOAD = 101
    STEP 20
        LOAD = 102
    STEP 30
        LOAD = 103
    STEP 40           $ | New added STEP
        NLPARM = 300
        LOAD = 104

$
BEGIN BULK
PARAM,PH2OUT,1 $ Request the intermediate output in PHASE II - the
$             residual run
NLPARM 300      30          AUTO
LOAD   104      3.0       1.0       101
$

```

```
$ no other bulk data is required for restart.  
$  
ENDDATA
```

When checking the.f06 output file after the Restart run, you will find that the printout only includes the load increments that have been requested in the restart run. There is no printout for the load increments before the restart point.

13.5 MSC.Nastran Implicit Nonlinear -- SOL 600

13.5.1 Introduction

With MSC.Nastran Implicit Nonlinear, also known as SOL 600, you can execute complex nonlinear problems with minimum changes to MSC.Nastran input used for other linear and nonlinear analyses. With simple nonlinear large deformation or plasticity analyses, for example, you need only modify the SOL statement. For more complex nonlinear analyses, you need to use only a few new bulk data entries and parameters to describe the new features.

In most cases, you can add new entries to an existing MSC.Nastran file without the use of a GUI (rigid contact, described by nurbs, normally require a GUI). Because a GUI usually makes data input easier, examine the capabilities of MSC.Patran to incorporate SOL 600 features into your model.

SOL 600 introduces the following new features:

- 2D & 3D Contact
 - Flexible-flexible surfaces
 - Flexible-rigid surfaces
 - Enforced displacement or velocity of rigid surfaces
- Large strain for all element types
- Nonlinear options in static or dynamic analyses
 - Natural frequency calculation after any time or load step
 - Buckling calculations after any time or load step
- Advanced nonlinear material models, some of which are listed below:
 - Ogden rubber formulation
 - Arruda-Boyce rubber formulation
 - Gent rubber formulation
 - Formulations to describe foam materials
 - Additional plasticity options
 - Plasticity of orthotropic and anisotropic materials
 - Strain-rate dependent materials
 - Crack growth simulation
 - Analysis of porous materials

- Additional soil models
 - Temperature-dependent plasticity, rubber, etc. descriptions
 - Visco-elastic materials
 - Thermo Visco-elastic materials
 - Viscoplasticity for creep analysis
 - Gasket materials
- Failure of portions of the structure
 - Composite materials modeled using solid elements
 - Parallel execution using domain decomposition

13.5.2 Background

MSC.Nastran has incorporated the most important nonlinear features of structural and heat transfer analyses through solutions 106, 129, 153 and 159. Basic features include large deformation, plasticity and additional nonlinear material descriptions, such as soils. With the integration of the nonlinear program MSC.Marc, which contains nearly all the advanced features of nonlinear analysis, there have been many requests from MSC.Nastran customers to incorporate many of these nonlinear features into MSC.Nastran. The most common request has been 3D contact of deformable bodies and large strain.

Two development projects were initiated to address these user desires. One project, which is not completely ready for the initial MSC.Nastran 2004 release, adds MSC.Marc's 3D contact routines directly inside MSC.Nastran. In addition, large rotation of rigid bodies and additional types of rigid mechanisms are being added.

The other project allows the MSC.Nastran user access to nearly all of MSC.Marc's advanced nonlinear technology for structural analysis while still maintaining the MSC.Nastran environment (or look and feel) both from an input and output perspective. This project is known as SOL 600. As described in the introduction, many additional nonlinear features become available through this project that would not be possible to incorporate directly inside MSC.Nastran in a reasonable length of time. Most of the features not incorporated in the current release are planned for a future release.

MSC.Nastran Implicit Nonlinear works as follows.

1. Inside MSC.Nastran, a translator between MSC.Nastran and MSC.Marc is automatically activated when the SOL 600 Executive Control statement is encountered. MSC.Nastran Executive Control, Case Control and Bulk Data are all examined and translated to the proper MSC.Marc formats whenever possible. When such translation is not possible, error messages or Translator Warning messages are issued. The translation is performed in the IFP portion of MSC.Nastran and an actual MSC.Marc input file is produced and saved.
2. The MSC.Marc input file is then be executed directly inside MSC.Nastran if requested by the user (it can also be saved for editing and execution of MSC.Marc in a separate run if so desired). The default is to execute MSC.Marc.
3. MSC.Marc's "unformatted" or "binary" results file known as the t16 file may optionally be converted to MSC.Nastran op2, xdb, and/or punch files. The results can even be placed in the f06 file in the same "format" as other MSC.Nastran "printed output" if so desired.

Description of the new SOL 600 Executive Control Statement:

The most important options for the new SOL 600 Executive Control statement are shown below:

SOL 600, ID PATH= COPYR= OUTR=op2,xdb,pch,f06 STOP=

Example: SOL 600,106 PATH=/progs/marc2003/tools OUTR=op2,f06,xdb

SOL 600,ID is a new Executive Control statement like SOL. The difference between other solution sequences and SOL 600 is that the computations (element matrix formulations, matrix decomposition, etc.) will be performed by a companion program (in this case MSC.Marc) rather than directly within MSC.Nastran. Inputs and outputs as much as possible will be the same as (or similar to) the familiar MSC.Nastran inputs and outputs. The SOL 600 statement should normally be used only for nonlinear analysis, but it may also be used for certain classes of linear static or dynamic analyses. The recommended form of this statement is shown with the options provided above. If entered with "SOL 600,ID" only, it acts just like SOL except a MSC.Marc input data file "jid.marc.dat" will be generated. ("jid" is the name of the MSC.Nastran input file without the extension). For example, if the MSC.Nastran input file is named abcd.dat, (or abcd.bdf) then jid is abcd).

ID (required) designates the familiar MSC.Nastran solution type as designated by integers or names in the same manner as used with a standard SOL statement. Examples are 106, 129, NLSTATIC, NLTRAN, etc. The following solutions are not available: 107, 110, 114, 115, 116, 118, 144, 145, 146, 153, 159, 190, 200, and 400 (and their equivalent names). Solutions specified in table 3-3 of the standard SOL statement may be used except for 7, 10, 14-16 and their equivalent names. In future releases, it is anticipated that the heat transfer solutions will be added.

PATH is an optional item which designates where to find the MSC.Marc executable. If **PATH=1** (the recommended option and the default if PATH is omitted) MSC.Nastran will determine the proper command to execute the companion program. To aid MSC.Nastran in determining the program's location, a file named **marcrun.pth** must be available in the same directory where the MSC.Nastran input file resides. The marcrun.pth file must contain one line providing the location (complete path) of the run_marc script. Typical examples of the line in the file marcrun.pth would be

```
/mycomputer/marc200x/tools      (UNIX)
```

```
c:\marc2003\tools      (Windows)
```

To this path MSC.Nastran appends the string “/run_marc -jid name.marc -v no -iam nanl” and possible other items to form the complete string used to execute MSC.Marc. This complete string looks like the string shown in the PATH=3 example below.

If **PATH=2** is specified, it is expected that the directory with the run_marc script is on the PATH. If PATH=2 is specified, MSC.Marc will be executed from inside MSC.Nastran using the command:

```
run_marc -jid jid.marc.dat -v no -iam nanl
```

Note that the last item (-iam nanl) requests a special type of licensing where MSC.Marc is offered with MSC.Nastran at reduced cost. If you already have a full version of MSC.Marc, this command line option is not needed and can be turned off for those options which automatically generate the command line using the bulk data parameter, PARAM,MARCIAMN,0.

When **PATH=3** is specified, the complete command to execute MSC.Marc must be contained in a file named marc.pth (lowercase). This file should typically contain one line of the form:

```
/mycomputer/marc200x/tools/run_marc -jid name.marc -v no -iam nanl
```

COPYR is an optional item. If **COPYR** is specified, MSC.Marc output files will be copied to MSC.Nastran output files and/or deleted according to the option (=0, =1, or =2) shown.

COPYR Option	Copy MSC.Marc Output Files to MSC.Nastran Output Files	Delete MSC.Marc Input & Output Files
0 (default)	No	No
1 or -1 (see below)	Yes	Yes
2 or -2 (see below)	Yes	No
3	No	Yes

If **COPYR** is 1 or 2, Marc's out and log files will be copied exactly as produced by MSC.Marc.

If **COPYR** is -1 or -2 the actions as shown above for +1 or +2 will occur, and MSC.Marc-type text will be converted to MSC.Nastran-type text (or any other desired text) using an ASCII file named `marcfile.txt`. This file must be located in the same directory where the MSC.Nastran input resides or in the same directory where the MSC.Marc executable resides. The `marcfile.txt` file can contain as many lines as desired like that shown below:

"Marc string 1" "Replacement String 1"

"Marc string 2" "Replacement String 2"

That is, each line contains two strings. Each string starts and ends with a double quote sign ("). The MSC.Marc string must match the exact content and case as found in the MSC.Marc `.out` or `.log` files. The replacement string may be any string desired and can be the same length, shorter or longer than the MSC.Marc string. The two strings must be separated by at least one space, but more spaces are acceptable. Line lengths for `marcfile.txt`, as well as MSC.Marc's `.out` and `.log` files are limited to 200 characters for the text replacement option.

The following MSC.Marc files are potentially affected by the **COPYR** option:

MSC.Marc Output File COPYR	MSC.Nastran Output Copied to	
name.marc.out	name.f06	1,2,-1,-2
name.marc.log	name.log	1,2,-1,-2

MSC.Marc Output File COPYR	MSC.Nastran Output Copied to	
name.marc.t16	not copied, will remain if produced	
name.op2, fort.11 or ftn 11	not copied, will remain if produced	

OUTR is an optional item. If **OUTR** is specified, MSC.Marc output results will be converted to various type of MSC.Nastran formats. The type of output to be produced depends on The **OUTR** options entered as well as any **DMAP** entered in the executive control. If **OUTR** is omitted, no MSC.Marc output will be brought back into MSC.Nastran, but standard MSC.Marc .out, .t16 and/or .t19 will be available. This is the recommended option for MSC.Patran users because the MSC.Marc t16 results file can be read by the MSC.Nastran preference in MSC.Patran. The following options are available:

OUTR=OP2,F11,F06,PCH,XDB,T16,T19

Use any or all of the above to request the following options:

- OP2** Create output2 file named jid.op2 consisting of input model and output results datablocks
- F11** Create output file fort.11 or ftn11 (depending on the computer system) consisting of output results datablocks only.
- F06** Put MSC.Marc output results (displacements, stresses, strains) in MSC.Nastran's jid.f06 file using OFP. The resulting output will look just like any standard MSC.Nastran run.
- PCH** Create punch file named jid.pch with MSC.Marc's output in standard MSC.Nastran punch format.
- XDB** Create xdb database file named jid.xdb with input model and output results. This Option required PARAM,POST,0 in the Bulk Data Section.
- T16** MSC.Marc's results will be saved during the MSC.Marc execution on a binary (or unformatted) file named jid.marc.t16
- T19** MSC.Marc's results will be saved during the MSC.Marc execution on an ASCII file named jid.marc.t19

Note that geometry data blocks may be placed in the op2 or xdb files but not both (unless you add your own DMAP statements). This is accomplished using the bulk data PARAM,POST,-1 (for op2) or PARAM,POST,0 (for xdb) parameters. If PARAM,POST is omitted, the geometry datablocks will not be present. Therefore you should normally only specify either outr=op2 or outr=xdb but not both.

STOP is an optional item. STOP is used to prevent execution of MSC.Marc or halt MSC.Nastran after IFP module if so desired. DO NOT ENTER any of the STOP options if any of the OUTR options are entered as the DMAP generated automatically by MSC.Nastran will put an EXIT in the proper place. The various options are as follows:

If **STOP=1** MSC.Nastran will be gracefully stopped after IFP. This option is used to prevent MSC.Nastran from performing its own solution (normally used when the solution is performed by the MSC.Marc and required for 3D contact unless the OUTR option is used).

For **STOP=2** MSC.Marc will not be executed. This options is frequently used if the user wishes to examine the MSC.Marc input file and make changes prior to running MSC.Marc. However, if STOP=2 is entered, the OUTR options will not be available.

STOP=3 is a combination of STOP=1 and STOP=2. MSC.Nastran is stopped after IFP and MSC.Marc is not executed. This would be the normal STOP option if the user wants to examine a MSC.Marc input file, then execute MSC.Marc manually. The STOP option should not be entered if the user wants to obtain comparative results between standard MSC.Nastran solutions and MSC.Marc solutions (in which case all input options must be fully supported by both programs).

Summary of Case Control and Bulk Data Entries Relating to SOL 600

The following Case Control command and Bulk Data entries are available in SOL 600 only and should not be used for other solution sequences. Please refer to the *MSC.Nastran Quick Reference Guide* for a detailed description of each entry.

Case Control Command

BCONTACT Selects a set of contact bodies for a particular subcase Options are available to select the entire model or all contact body definitions.

Note: Line contact options are not available in SOL 600.

New Bulk Data Entries

BCBODY	Defines flexible or rigid contact bodies
BCBOX	Defines a 3D flexible contact region within a box (8 grid points or 8 xyz coordinates)
BCHANGE	Changes definition of contact bodies
BCMATL	Defines a 3D flexible contact region described by a material id
BCMOVE	Defines movement of bodies in contact
BCPARA	Contact parameters
BCPROP	Defines a 3D flexible contact region described by a property id
BCTABLE	Defines a contact table (which bodies can make contact)
BSURF	Defines a flexible contact surface using element IDs
GMNURB	Alternate way to describe a rigid contact surface made of nurbs
IPSTRAIN	Initial element plastic strains
ISTRESS	Initial element stresses
MARCIN	A way to directly input data to various portions of a MSC.Marc input file
MARCOUT	Selection of which outputs will be placed on MSC.Marc's t16/t19 results files
MATEP	Elasto-Plastic material property description
MATHE	Hyperelastic material property description (Mooney, Ogden, Foam, etc.)
MATHED	Damage model descriptions for hyperelastic materials
MATT	Failure model descriptions
MATG	Gasket material property description
MATORT	Orthotropic material property description
MATV	Visco-elastic material property description
MATVP	Viscoplastic or Creep material property description
MATTEP	Temperature dependent Elasto-Plastic material property description
MATTHE	Temperature dependent Hyperelastic material property description

MATTG	Temperature dependent Gasket material property description
MATTORT	Temperature dependent Orthotropic material property description
MATTVE	Temperature dependent Visco-elastic material property description
NLAUTO	Controls for automatic load or time stepping (MSC.Marc's AUTO STEP method)
NLDAMP	Damping constants for SOL 600
NLSTRAT	Solution strategy parameters for SOL 600
NTHICK	Define plate/shell thickness at nodes rather than from property entries
PARAMARC	Defines number of domains and how to define them for parallel processing in SOL 600
RESTART	Specifies restart options for SOL 600

Note: Line contact options are not available in SOL 600.

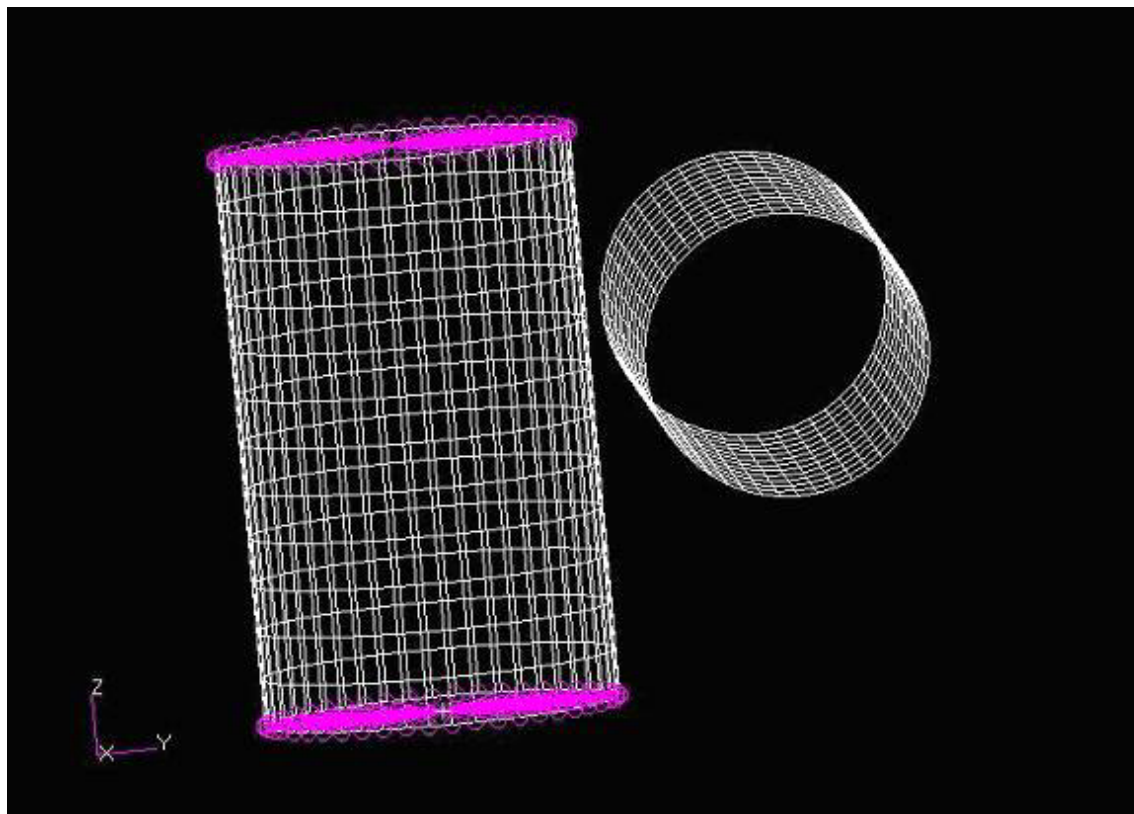
Bulk Data Parameters

LGSTRN	Selects small or large strain analysis and strain measures to use for SOL 600
STRESS	Selects stress measure to use for SOL 600 (*)
MARCPOS	Selects whether a non-positive definite matrix may or may not be solved for SOL 600 (*)
MARCOPT	Selects bandwidth optimization option for SOL 600 (*)
MARCMEM	Selects amount of memory to be used by MSC.Marc (*)
MARCSIZi	Allows you to specify all values on MSC.Marc's SIZING entry (*)
MARCDISi	Allows you to specify all values on MSC.Marc's DIST LOADS memory specification entry (*)
MARCCONi	Allows you to specify all values memory-related values on MSC.Marc's contact entries (*)
MARCVERS	Specifies which version of MSC.Marc will be used (the current released version is the default) (*)
MARCRBE2	Specifies the MSC.Marc formulation of RBE2s (*)
MARCRBE3	Specifies the MSC.Marc formulation of RBE3s (*)

MARCRBAR	Specifies the MSC.Marc formulation of RBARs (*)
MARCSINC	Controls how often spline output files are written (*)
MARCT19	Allows you to generate an ASCII MSC.Marc results file
MARCT16	Allows you to generate an unformatted (or binary) MSC.Marc results file, done by default (*)
MARCEKND	Specifies which type of strains (total, plastic, elastic) are placed in op2,xdb files (*)
MARCSHLT	Specifies how many shell integration layers are used in plasticity analyses (*)
MARCPST	Specifies the MSC.Marc version designation of MSC.Marc's t16/t19 files (default=10) (*)
MARUPDAT	Specifies whether to use Total Lagrange or Updated Lagrange method (*)
MARSUMY	Requests a summary of output at each time or load step in the .out file (*)
MRSRING	Specifies a stiffness value to be added to the main diagonal of the stiffness matrix
MARCIAMN	Specifies whether a special or full version of MSC.Marc is to be used
MARCDILT	Specifies whether constant dilation formulation or not
MARCASUM	Specifies whether assumed strain formulation is used or not (*)
MARCGAUS	Specifies whether output is at Gauss points or center (*)
MRCONRES	Specifies whether continuous or separate output files are produced for restart runs
MARCRBAL	Specifies whether left hand and right hand sides of equation should be re-balanced before computing eigenvalues.
MRALIAS	Specifies an alias element type (similar element with a different formulation) (*)
MALIASij	Specifies an alias element type (similar element with a different formulation) (*)

13.5.3 Example: Two Pipes Crushing (Contact & Large Strain):

Let's examine a problem to illustrate how easy it is to change a standard MSC.Nastran input file to describe two of the more advanced nonlinear capabilities. Consider the case of two pipes initially separated by a small distance as shown.



There is a force at the top and bottom of the left cylinder forcing it toward the right cylinder. The left cylinder has rigid (RBE2) connections at the top and bottom and is connected to ground with soft springs. The right cylinder is pinned at one end and restrained in the radial direction at the other end. The soft springs are necessary for a nonlinear static analysis as the left cylinder would otherwise be unconstrained.

Partial input for SOL 600 is shown below (note that ID=NLSTATIC which is the same as 106, path=1 means the path to run MSC.Marc is specified in a file named marcrun.pth in the same directory as the MSC.Nastran input data and outr=op2,f06 means that the results will be placed in a standard MSC.Nastran OP2 file as well as printed in the f06 file). New or modified input data is shown in **bold**.

```
SOL 600,NLSTATIC path=1 outr=op2,f06
TIME 10000
CEND
  ECHO = NONE
  DISPLACEMENT(print,PLOT) = ALL
$ NLSTRESS(PLOT) = ALL
```

```

STRESS(PLOT,CORNER) = ALL
strain(plot)=all
SPC = 1
LOAD = 1
NLPARM = 1
bcontact=121
BEGIN BULK
NLPARM          1      100          AUTO          1      25          YES+
+
+
PARAM,LGDISP,1
$PARAM,POST,-1
PARAM,OGEOM,NO
PARAM,AUTOSPC,YES
PARAM,GRDPNT,0
FORCE           1      1301          0          1.          0.          5000.          0.
FORCE           1      1302          0          1.          0.          5000.          0.
SPC             1      663          23         0.
SPC             1      664          23         0.

CQUAD4          1149          2      1297      1298      1015      1016
CQUAD4          1150          2      1298      1299      1014      1015
CQUAD4          1151          2      1299      1300      1013      1014
CQUAD4          1152          2      1300      1011      1012      1013
RBE2            1153      1301      123456      327      328      329      330      331+
+              332      333      334      335      336      337      338      339+
+              340      341      342      343      344      345      346      347+
+              348      349      14       15       16       17       18       19+
+              20       21       22       23       24       25       26       27+
+              28       29       30       31       32       33       34       35+
+              36       326      13
RBE2            1154      1302      123456      363      364      365      366      367+
+              368      369      370      371      372      373      374      375+
+              376      377      378      379      380      381      382      383+
+              384      385      50       51       52       53       54       55+
+              56       57       58       59       60       61       62       63+
+              64       65       66       67       68       69       70       71+
+              72       49       362
MAT1            21000000.          0.25
PROD            4          2      1.E-5
CROD            1155          4      1301      1303
PROD            5          2      1.E-5
CROD            1156          5      1302      1304
$
BSURF, 101, 1, THRU, 576
BSURF, 102, 577, THRU, 1152
BCBODY, 111, , DEFORM, 101, 0, .25
BCBODY, 112, , DEFORM, 102, 0, .25
BCTABLE, 121, , , 1, , , , +
+ , SLAVE, 111, , , .25,
+ , MASTER, 112
$
ENDDATA

```

Note that only the SOL 600,NLSTATIC Executive Control statement, the BCONTACT=121 Case Control command, BSURF, BCBODY and BCTABLE Bulk Data entries are changed or new compared to a standard MSC.Nastran input file. Of these, all except the SOL 600,NLSTATIC statement describe contact between the cylinders. Here is a detailed description of these commands:

SOL 600,NLSTATIC path=1 outr=op2,f06

The SOL 600,NLSTATIC statement is just like a standard MSC.Nastran Executive Control SOL statement except for the “600,” before NLSTATIC and the extra information after NLSTATIC. This information is as follows:

- path=1 -- This means that a file named marcrun.pth is in the same directory as the MSC.Nastran input data and will specify the full path to MSC.Marc’s “tools” directory.
- outr=op2,f06 -- This means that after the MSC.Marc execution has completed, displacements, stresses tensors and plastic strains tensors will be placed in standard op2 and f06 files. These results although calculated by MSC.Marc will look identical in format to those produced by a standard MSC.Nastran analysis which does not spawn MSC.Marc. This should not be confused with the copy option which would copy the MSC.Marc results to MSC.Nastran’s f06 file (in which case the output would still look like MSC.Marc output). The op2 and f06 results are produced by new code in MSC.Nastran which reads MSC.Marc’s binary output t16 file and generates a results-only op2 file which we call an f11 file. DMAP is generated “on the fly” to read the f11 file into the MSC.Nastran database (DBALL and MASTER) using INPUTT2. Additional DMAP is generated to produce the desired OP2 file which now contains geometry datablocks as well as results datablocks. Finally DMAP is generated on the fly to place the results in the f06 file using the OFP module.

BCONTACT=121

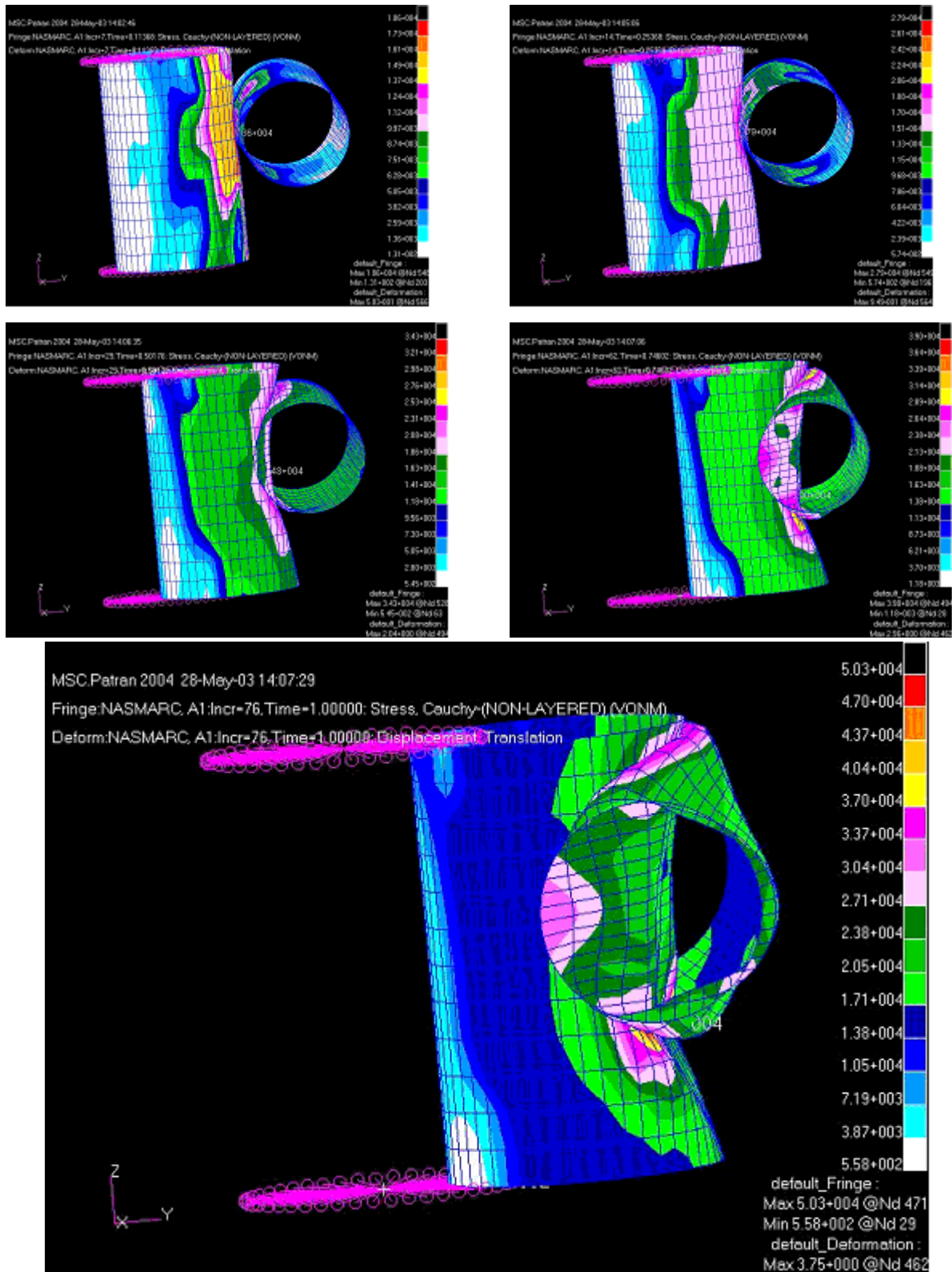
This Case Control command means that contact will be described using a BCTABLE Bulk Data entry with ID 121.

Contact Bulk Data Entries:

```
BSURF, 101, 1, THRU, 576
BSURF, 102, 577, THRU, 1152
BCBODY, 111, , DEFORM, 101, 0, .25
BCBODY, 112, , DEFORM, 102, 0, .25
BCTABLE, 121, , , 1, , , , , +
+ , SLAVE, 111, , , .25,
+ , MASTER, 112
```


These Bulk Data entries describe the contact surfaces. Because large deformation is expected, all elements of the left cylinder may potentially contact those of the right cylinder. The two BSURF entries describe the elements of each cylinder. The two BCBODY entries describe the friction coefficients (0.25) for each of the cylindrical bodies. Finally, the BCTABLE hooks up the two cylindrical surfaces (as described by the two BCBODY entries) as a contact condition for the subcase. If there were more bodies and more subcases, different contact bodies could be described for each subcase if so desired. For small models, it is not necessary to describe the bodies. The Case Control BCONTACT command could be changed to BCONTACT=ALL and then each element in the model would check for contact with every other element in the model at each time (or load) step. Obviously, this would require more computer resources than if those elements that might possibly contact are specified by the user.

The MSC.Nastran run is submitted the same as any other MSC.Nastran run. In this model, there is no stop option on the SOL 600, ID statement, so MSC.Marc will automatically be executed. Since OUTF=op2,f06 is specified, the results will be brought back into the "MSC.Nastran Database" to produce op2 and f06 (printed) output as explained above. The MSC.Marc input and output files will remain (in the same directory as the MSC.Nastran input data) for the user to inspect after the run. Results may then be examined or plotted using any standard MSC.Nastran postprocessor. For example, the following results were produced using MSC.Patran from the op2 file.



We elected to place the results in the f06 file as well as the op2 file. A sample of some of the results from the. f06 file follow.

The following are samples “total displacements i.e. relative to the initial position, Cauchy stresses and Plastic strains at the final position (“time”=1.0).

DISPLACEMENT VECTOR								
POINT ID.	TYPE	T1	T2	T3	R1	R2	R3	
2	G	-8.006727E-03	3.679180E+00	1.043056E-03	2.845422E-03	2.369322E-02	1.204350E-02	
3	G	-2.011038E-02	3.677816E+00	1.519643E-03	2.597190E-03	1.290626E-02	3.169994E-02	
5	G	-3.224353E-02	3.676953E+00	1.765113E-03	2.757947E-04	7.053518E-03	6.177027E-02	
6	G	-3.788499E-02	3.676907E+00	1.501512E-03	4.993313E-03	9.955634E-03	8.023518E-02	
10	G	-1.402493E-02	3.683437E+00	1.045132E-04	1.284459E-03	-2.054870E-02	2.056682E-02	

STRESSES IN QUADRILATERAL ELEMENTS (QUAD4)									
ELEMENT ID.	FIBER DISTANCE	STRESSES IN ELEMENT COORD SYSTEM			PRINCIPAL STRESSES (ZERO SHEAR)			VON MISES	
		NORMAL-X	NORMAL-Y	SHEAR-XY	ANGLE	MAJOR	MINOR		
0 1	-1.000000E+00	2.460815E+03	1.935240E+03	2.565853E+03	42.0762	4.777302E+03	-3.812473E+02	4.978885E+03	
	1.000000E+00	-9.141631E+03	-8.298519E+03	5.878244E+03	47.0510	-2.826734E+03	-1.461342E+04	1.342513E+04	
0 2	-1.000000E+00	-1.057425E+04	5.934804E+03	-1.950141E+03	-83.3538	6.162038E+03	-1.080149E+04	1.487286E+04	
	1.000000E+00	7.265749E+03	-5.363500E+03	6.036017E+03	21.8539	9.686568E+03	-7.784319E+03	1.516010E+04	
0 3	-1.000000E+00	2.479291E+03	1.367514E+04	4.870156E+03	69.4885	1.549713E+04	6.572963E+02	1.517916E+04	
	1.000000E+00	-3.051442E+03	-9.212279E+03	2.676518E+03	20.4934	-2.051085E+03	-1.021264E+04	9.357237E+03	

STRAINS IN QUADRILATERAL ELEMENTS (QUAD4)									
ELEMENT ID.	FIBER DISTANCE	STRAINS IN ELEMENT COORD SYSTEM			PRINCIPAL STRAINS (ZERO SHEAR)			VON MISES	
		NORMAL-X	NORMAL-Y	SHEAR-XY	ANGLE	MAJOR	MINOR		
0 1	-1.000000E+00	-1.928319E-04	1.118255E-04	3.583091E-04	56.5159	3.488417E-04	-4.298481E-04	6.755803E-04	
	1.000000E+00	-2.790075E-03	-1.048802E-04	2.158601E-03	60.9403	1.094594E-03	-3.989548E-03	4.634821E-03	
0 2	-1.000000E+00	-1.151780E-03	8.069643E-04	-1.218730E-03	-64.3927	1.391073E-03	-1.735889E-03	2.713512E-03	
	1.000000E+00	5.727272E-05	-6.128556E-04	2.944954E-03	41.7545	2.686162E-03	-3.241745E-03	5.141228E-03	
0 3	-1.000000E+00	-5.297469E-04	1.939139E-03	-9.462342E-04	-71.2644	2.260077E-03	-8.506845E-04	2.784641E-03	
	1.000000E+00	-1.830743E-05	-1.381564E-03	1.953345E-03	35.3817	1.368922E-03	-2.768794E-03	3.651086E-03	

Where Can I Find More Information

MC.Nastran Implicit Nonlinear Analysis -- SOL 600 is documented in the following manuals and guides:

- MSC.Nastran Quick Reference Guide
- MSC.Nastran Implicit Nonlinear User’s Guide
- MSC.Patran User’s Guide
- MSC.Patran – MSC.Nastran Preference Guide volumes 1 and 2
- MSC.Marc Volume A -- Theory and User Information
- MSC.Marc Volume B -- Element Library
- MSC.Marc Volume C -- Program Input

13.5 Coupled Fluid-Structure Interaction

13.5.1 Coupled Fluid-Structural Analysis

Several methods are available in MSC.Nastran to analyze the effects of a fluid on a structure.

- Coupled Fluid-Structure -- discussed in this section.
- Hydroelastic – discussed in “[Hydroelastic Modeling](#)” on page 794.
 - Fluid elements
 - Virtual mass
- Acoustic Cavity -- discussed in “[Acoustic Cavity Modeling](#)” on page 795.

Introduction

MSC.Nastran has the capability to perform a fully coupled fluid-structure analysis. Its principal application is in the area of acoustic and noise control analysis; for example, in the passenger compartments of automobiles and aircraft.

The approach used in this analysis is called the Pressure Method, which is analogous to the Displacement Method in structural analysis except that pressures, instead of displacements, are computed at the fluid points. The velocities and accelerations of the fluid points are analogous to forces in structural analysis.

The fluid may be modeled with existing three-dimensional elements: CHEXA, CPENTA, and CTETRA. These elements can assume the properties of irrotational and compressible fluids suitable for acoustic analysis or other types of analyses governed by the three-dimensional wave equation. In addition, two elements are available for acoustic analysis, the barrier and absorber. The absorber and barrier elements may be used to analyze acoustic noise control devices.

The interface between the fluid and the structure may be modeled so that the grid points of the fluid are coincident with those of the structure. This is called a matching mesh. If not, then it is called a nonmatching mesh. In either case and by default, coupling for the stiffness and mass is automatically computed. If coupling is not desired, then insert ASCOUP=NO on the FLSTCNT Case Control command.

Coupled fluid-structural analysis is available in the dynamic solution sequences using the Direct Method (SOLs 107 through 109 and 200 with ANALYSIS = DFREQ), and the Modal Method (SOLs 110 through 112 and 200 with ANALYSIS = MFREQ and ANALYSIS = MTRAN). It should be noted that in SOLUTIONs 110 through 112 and 200, the normal modes are computed separately for fluid and structure parts of the model;

in other words, the uncoupled modes of the fluid and structure are used in the SDAMPING Case Control command and the parameters G and W3 are applied to the structural portion of the model and the SDAMPING(FLUID) Case Control command and the parameters GFL and W3FL are applied to the fluid portion of the model. Design Sensitivities may be computed in SOL 200.

In SOLs 103 and 106 (PARAM,NMLOOP), normal modes are computed separately for the fluid and structure portions of the model and then put together in the eigenvalue table and displacement printouts in the .f06 file output. Please note that while the fluid and structure eigenvalues and eigenvectors are printed together, they are really independent of each other and not true combined eigenvalues. The combined normal modes computed in SOL 103 may not be used in a restart into SOLs 110, 111, 112, or 200.

Fluid Model Definition

Fluid grid points are defined on the GRID Bulk Data entry by specifying a value of -1 for CD in field 7. Fluid elements are defined on the CHEXA, CPENTA, and CTETRA Bulk Data entries. Also, on the referenced PSOLID entry, the character value PFLUID must be specified for FCTN in field 8 and MID, field 3, must reference a MAT10 material entry. The MAT10 entry defines the bulk modulus and the mass density properties of the fluid. If a PSOLID entry defines fluid elements (FCTN = PFLUID), then ISOP, field 7, is defaulted to 1 (FULL), resulting in a full integration scheme.

Panel

It is sometimes useful to find the contribution of a set of grid points to the noise level in an acoustic cavity. For example, in an automobile the entire roof assembly contributes significantly to the noise level inside the passenger cabin; as such it is necessary to compute and minimize the contribution of this “panel.” A “panel” may be defined with the PANEL entry, which references a set of structural grid points listed on a SET1 Bulk Data entry.

Acoustic Absorber

The acoustic absorber element is simulated by an assembly consisting of a mass attached to a spring and damper which are in parallel. The element and its properties are defined by the CHACAB and PACABS entries, respectively. The element consists of 8 to 16 structural grid points defining two surfaces.

The PACABS entry references three TABLEDi entries which define the resistance, Y1; reactance, Y2; and the confidence level, Y3 of the material. These properties are a function of frequency and are measured in the Standing Wave Tube Test (ASTM Standard). This data is used to calculate the value of the mass, spring, and damper used in the structural model that will emulate the behavior of the absorber material.

The last field on the PACABS entry defines the cutoff frequency which determines the extent of the data used in the least square fit operations. This is needed because finite element analysis is not applicable to acoustic analysis in the high frequency range. In fact, it is recommended that this methodology be restricted to frequencies with wavelengths much larger than the absorber thickness. Nevertheless, the user may override this cutoff frequency so that the experimental results of short wavelengths are used in the calculations.

Frequency Dependent Acoustic Absorber

The acoustic absorber element CACABSF is an element connecting 1 to 4 fluid grids at the fluid-structure interface. Its frequency dependent impedance is specified on the PAABSF entry.

If one grid point is given, then the impedance $Z(f) = Z_R + i Z_I$ is the total impedance at the point. If two grids are specified, then the impedance is the impedance per unit length. If three or four points are specified, then the impedance is the impedance per unit area. $Z_R(f) = TZREID(f) + B$ and $Z_I(f) = TZRIMD(f) - K/(2\pi f)$.

The resistance represents a damper quantity B. The reactance represents a quantity of the type $(\omega M - K/\omega)$. The impedance is defined as $Z = p/\dot{u}$ where p is the pressure and \dot{u} is the velocity. The scale factor S is used in computing element stiffness and damping terms as:

$$k = \frac{A}{S} \cdot \frac{2\pi f Z_I(f)}{Z_R^2 + Z_I^2} \int (\text{of shape functions})$$

$$b = \frac{A}{S} \cdot \frac{Z_R(f)}{Z_R^2 + Z_I^2} \int (\text{of shape functions})$$

The absorption coefficient is calculated as:

$$\alpha = \frac{4(Z_R/\rho c)}{(Z_R/\rho c + 1)^2 + (Z_I/\rho c)^2}$$

Acoustic Barrier

The acoustic barrier element is synthesized with two masses sandwiching a spring. The element and its properties are defined by the CHACBR and PACBAR entries, respectively. The element consists of 8 to 16 structural grid points defining two surfaces.

The PACBAR entry defines the mass-per-unit area of the backing and the septum, and the resonant frequency of the barrier. The resonant frequency of the sandwich is defined as the frequency with the lowest value of transmission loss (ASTM test). This frequency will be used to calculate the value of the spring used in the structural model that will emulate the behavior of the barrier material.

Loading

Loads on fluid elements are, in most cases, analogous to enforced displacements on structural elements. The types of loading available for fluid elements are:

1. Constant, frequency-dependent, or time-dependent enforced pressure at the grid points.
2. An acoustic source characterized by a volumetric flow rate and corresponding to a power spectral density function.

Enforced Pressures

The Large Mass approach is used to define an enforced pressure. This is not recommended for enforcing motion on more than one point.

Acoustic Source

An acoustic source is assumed to be a pulsating sphere in infinite space and is defined on the ACSRCE Bulk Data entry. The ACSRCE entry is selected by the DLOAD Case Control command and contains the material properties of the source and references a DAREA and TABLEDi entry. The TABLEDi entry defines the power-versus-frequency curve characterizing the acoustic source. The ACSRCE entry may also define a delay time and phase angle which is useful whenever multiple sources are present.

The complex source strength is defined in Kinsler, Frey, Coppens, and Sanders, *Fundamentals of Acoustics*, John Wiley, 1982. as

$$Qe^{i\omega t} = \int_s \dot{\mathbf{u}} \cdot \hat{\mathbf{n}} \, dS$$

Also, similar to the Maxwell-Betti reciprocity law in structures, they define a principle of acoustic reciprocity which states that in an unchanging environment if the locations of a small receiver and small source are interchanged, the received signal will remain the same.

With these definitions, they arrive at the expression for power $P(f)$ in terms of complex source strength, which for a pulsating sphere is a real value Q . The resulting expression as used in MSC.Nastran is:

$$Q = \frac{1}{2\pi f \lambda} \sqrt{\frac{8\pi c P(f)}{\rho}}$$

Single-Point Constraints

Single-point constraints of the fluid ($P = 0.0$) may be enforced using the SPC entry or the PS field on the GRID entry. However, only one degree-of-freedom per grid point may be constrained. This type of boundary condition occurs at free surfaces.

Fluid-Structure Interface

The fluid-structure interface is determined automatically by the program.

A matching interface is defined by entering IDENT in the INTER field of the ACMODL Bulk Data entry. This also requires that coincident grid points be defined for the structure and fluid elements at the interface. The wetted structural elements are determined by comparing grid point locations corresponding to structural elements that are within a tolerance specified by NORMAL on the ACMODL entry. If no structural elements are found, a warning message will be issued.

A nonmatching interface is defined by entering DIFF in the INTER field and either CP or BW in the METHOD field on the ACMODL Bulk Data entry. For large complex models, it is recommended initially, that the ACMODL not be used. "BW" requires an acoustic license.

Normal Modes Analysis

Since the fluid portion of the model may resonate within a different frequency range than the structural portion, it is possible to specify a different eigenvalue extraction method for each in the residual structure only. In modal analysis (SOLs 110 through 112 and 200), this is achieved by the specification of a keyword on the METHOD Case Control command. The METHOD(FLUID) command references the EIQR or EIGRL entry to be used for the fluid portion; METHOD(STRUCTURE) references the same entries for the structural portion. Either one or both may be specified in the Case Control Section. The default is STRUCTURE.

Superelement Analysis

Fluid-structure models may be defined using superelements with the following restrictions:

1. A superelement may contain either fluid or structural points, but not both. (The residual structure may contain both.)
2. The grid points at the fluid-structure interface may be assigned to the residual structure only.

This requires the specification of q-set points using the SEQSETi and ELGR or ELGRL Bulk Data entries and the METHOD Case Control command.

Case Control, Output and Plotting

If eigenvalue analysis is desired, METHOD(STRUCT) and METHOD(FLUID) are required in case control.

Fluid-structure superelement and frequency selection can be accomplished by the FLSFSEL command.

Fluid-structure symmetry, coupling, and force selection can be accomplished by the FLSTCNT command.

Fluid-structure mode participation output requests can be accomplished by the FLSPOUT command.

The DISPLACEMENT or PRESSURE Case Control command is used to request:

1. Pressure and peak sound pressure levels at the fluid points.
2. RMS sound pressure level is requested by the ACOUT=RMS on the FLSTCNT command. This output is available in Modal and Direct Frequency Response analysis only (SOLs 111, 108, and 200).
3. Sound pressure level in dB and dBA--a peak reference pressure must be specified by the PREFDB on the FLSTCNT command. The dB level is defined as $\text{dB} = 20 \log(P/\text{PREFDB})$ (for air $\text{PREFDB} = 20 \times 10^{-6}$ Pascal).

In SOLutions 110, 111, 112, and 200 the SVECTOR, SDISPLACEMENT, SVELOCITY, SACCELERATION Case Control commands result in separate output for the fluid and structural portion of the model. The first output is for the structural grid points and the second for the fluid grid points.

Particle velocity is requested with the FORCE Case Control command.

Basic Fluid Equations

This section collects up the various acoustic equations used in MSC.Nastran.

In MSC.Nastran the current acoustic equations are based on small motion theory with negligible convective momentum terms and locally linear pressure-density relationship. Thus Euler's equation is:

$$\ddot{u}_f = -\frac{1}{\rho_f} \ddot{\nabla} p \quad \text{Eq. 13-37}$$

and the Continuity equation is

$$p = -\beta \ddot{\nabla} \cdot \dot{u}_f \quad \text{Eq. 13-38}$$

with the compressibility β defined as

$$\beta = c^2 \rho_f \quad \text{Eq. 13-39}$$

Take the second time derivative of continuity to get

$$\ddot{\nabla} \cdot \ddot{u}_f = -\frac{1}{\beta} \ddot{p} \quad \text{Eq. 13-40}$$

and operate on Euler's equation with the del operator to get

$$\ddot{\nabla} \cdot \ddot{u}_f = -\frac{1}{\rho_f} \ddot{\nabla} \cdot \ddot{\nabla} p$$

or

$$\frac{1}{\beta} \ddot{p} - \frac{1}{\rho_f} \ddot{\nabla} \cdot \ddot{\nabla} p = 0 \quad \text{Eq. 13-41}$$

integrate this equation over the fluid volume and multiply by a virtual pressure δp to get the integral

$$\int_V \left[\frac{1}{\beta} \ddot{p} - \frac{1}{\rho_f} \ddot{\nabla} \cdot \ddot{\nabla} p \right] dV \delta p = 0 \quad \text{Eq. 13-42}$$

This represents the virtual work expression for acoustic pressure. Integrating the second term by parts using Green's theorem and the identities:

$$\vec{\nabla} \cdot (\alpha \vec{v}) = \vec{\nabla} \alpha \cdot \vec{v} + \alpha \vec{\nabla} \cdot \vec{v}$$

$$\alpha = \delta p$$

$$\vec{v} = \vec{\nabla} p$$

$$\vec{\nabla} \delta p = \delta \vec{\nabla} p$$

$$\delta(\vec{\nabla} p \cdot \vec{\nabla} p) = 2 \delta \vec{\nabla} p \cdot \vec{\nabla} p$$

we get

$$\int_V \frac{1}{\beta} \ddot{p} \delta p dV + \delta \int_V \frac{1}{2\rho_f} \vec{\nabla} p \cdot \vec{\nabla} p dV - \int_S \frac{1}{\rho_f} \vec{\nabla} p \cdot d\vec{S} \delta p = 0 \quad \text{Eq. 13-43}$$

The trick is to factor out the δ operator from each integral. The first term in the above equation, however, contains \ddot{p} which prevents a simple factoring out of δp . Thus no simple variational statement is directly possible. Because of this, it is standard practice at this point to apply the Galerkin method. This is equivalent to assuming that p but not \ddot{p} is subject to variation. This allows us to simply factor out the δp from the integral containing \ddot{p} . Doing this and also defining \ddot{u}_n^f as the acceleration of the fluid in the direction of the outward fluid normal, we obtain for the integral:

$$\Pi = \int_V \left[\frac{1}{\beta} p \ddot{p} + \frac{1}{2\rho_f} \vec{\nabla} p \cdot \vec{\nabla} p \right] dV + \int_S \ddot{u}_n^f p dS \quad \text{Eq. 13-44}$$

and the variational statement becomes

$$\delta \Pi = 0 \quad \text{Eq. 13-45}$$

Let the fluid pressure p at any point be defined as

$$p = [N_f] \{p\}$$

and

$$\ddot{p} = [N_f] \{\ddot{p}\}$$

where $[N_f]$ is the row matrix of pressure shape functions and $\{p\}$ is a vector of nodal pressures. The expression for Π can be expressed as:

$$\begin{aligned}\Pi &= [p] \int_V \frac{1}{\beta} \{N_f\} [N_f] dV \{p\} \\ &+ [p] \int_V \frac{1}{2\rho_f} \{\nabla N_f\} [\nabla N_f] dV \{p\} \\ &+ [p] \int_S \{N_f\} \ddot{u}_{n_o}^f dS\end{aligned}\quad \text{Eq. 13-46}$$

If \vec{n}_0 is the fluid outward normal at the fluid-structure and \vec{n} is the structure boundary normal pointing into the fluid, then $\vec{n}_0 = -\vec{n}$ and the surface integral can be written as

$$[p] \int_S \{N_f\} \ddot{u}_{n_0}^f dS = -[p] \int_S \{N_f\} \ddot{u}_n^s dS$$

where \ddot{u}_n^s is the normal acceleration of the structure.

If $[C]$ is the direction cosine matrix for \vec{n} and $[N_s]$ is the matrix of shape functions for the structure, we can define

$$\begin{aligned}\ddot{u}_n^s &= [C] [N_s] \{\ddot{u}_s\} \\ &= [N_s] \{\ddot{u}_s\}\end{aligned}\quad \text{Eq. 13-47}$$

where $\{\ddot{u}_s\}$ is the matrix of nodal accelerations at the boundary and $[N_s]$ is the row matrix of shape functions for the structural normal.

$$-[p] \int_S \{N_f\} \ddot{u}_n^s dS = -[p] \int_S \{N_f\} [N_s] dS \{\ddot{u}_s\} \quad \text{Eq. 13-48}$$

We may then compute $\delta\Pi$ and define the following integrals:

$$[M_f] = \int_V \frac{1}{\beta} \{N_f\} [N_f] dV \quad \text{Eq. 13-49}$$

$$[K_f] = \int_V \frac{1}{2\rho_f} \{\nabla N_f\} [\nabla N_f] dV \quad \text{Eq. 13-50}$$

$$[A^T] = \int_S \{N_f\} [N_s] dS \quad \text{Eq. 13-51}$$

and arrive at the relationship

$$[M_f]\{\ddot{p}\} + [K_f]\{p\} - [A^T]\{\ddot{u}_s\} = 0 \quad \text{Eq. 13-52}$$

MSC.Nastran allows for the effect of acoustic source density, which represents a term Q on the right-hand side of the continuity equation. When included into the virtual work expression, this will yield an acoustic load $\{P_f\}$ on the right-hand side of **Eq. 13-52**. Also, MSC.Nastran allows for frequency dependent wall impedance $z = p/\dot{u}_f$ at the structure-fluid $[B_f]\{\dot{p}\}$. Thus, **Eq. 13-52** becomes:

$$[M_f]\{\ddot{p}\} + [B_f]\{\dot{p}\} + [K_f]\{p\} - [A^T]\{\ddot{u}_s\} = \{P_f\} \quad \text{Eq. 13-53}$$

The fluid pressure on the structure boundary causes surface tractions on the structure represented by the standard relationship

$$\{F_s\} = \int_S [N_s]^T \{\Phi\} dS \quad \text{Eq. 13-54}$$

But, the traction Φ is related to the fluid pressure by the relation

$$\{\Phi\} = -p\{C\} \quad \text{Eq. 13-55}$$

or

$$\begin{aligned} \{F_s\} &= -\int_S [N_s]^T \{C\} p dS \\ &= -\int_S \{N_s\} [N_f] dS \{p\} \\ &= -[A]\{p\} \end{aligned} \quad \text{Eq. 13-56}$$

The structure equations can now be written as:

$$[M_s]\{\ddot{u}_s\} + [B_s]\{\dot{u}_s\} + [K_s]\{u_s\} = \{P_s\} - [A]\{p\} \quad \text{Eq. 13-57}$$

The fluid and structure equations can now be combined as

$$\begin{bmatrix} M_s & \mathbf{0} \\ -A^T & M_f \end{bmatrix} \begin{Bmatrix} \ddot{u}_s \\ \ddot{p} \end{Bmatrix} + \begin{bmatrix} B_s & \mathbf{0} \\ \mathbf{0} & B_f \end{bmatrix} \begin{Bmatrix} \dot{u}_s \\ \dot{p} \end{Bmatrix} + \begin{bmatrix} K_s & A \\ \mathbf{0} & K_f \end{bmatrix} \begin{Bmatrix} u_s \\ p \end{Bmatrix} = \begin{Bmatrix} P_s \\ P_f \end{Bmatrix} \quad \text{Eq. 13-58}$$

This is the fluid structure equation solved in MSC.Nastran. Note that this equation is nonsymmetric. By default, MSC.Nastran solves a symmetric version of this equation developed by Everstine as follows:

Let the velocity potential q be defined as

$$p = \dot{q} \quad \text{Eq. 13-59}$$

and note the following:

$$\begin{aligned} \ddot{u}_s &= \frac{d\dot{u}_s}{dt} \\ \ddot{p} &= \frac{d\dot{p}}{dt} \\ \dot{p} &= \frac{dp}{dt} \\ p &= \frac{dq}{dt} \end{aligned}$$

and define G as

$$G = -\int_0^t P_f(\tau) d\tau \quad \text{Eq. 13-60}$$

Substitute \dot{q} for p in the structure equation and for the fluid equation replace the vector terms with their derivative equivalents. Then multiplying the fluid equation by -1 and integrating with respect to time and recombining, we get the symmetric equation

$$\begin{bmatrix} M_s & \mathbf{0} \\ \mathbf{0} & -M_f \end{bmatrix} \begin{Bmatrix} \ddot{u}_s \\ \ddot{q} \end{Bmatrix} + \begin{bmatrix} B_s & A \\ A^T & -B_f \end{bmatrix} \begin{Bmatrix} \dot{u}_s \\ \dot{q} \end{Bmatrix} + \begin{bmatrix} K_s & \mathbf{0} \\ \mathbf{0} & -K_f \end{bmatrix} \begin{Bmatrix} u_s \\ q \end{Bmatrix} = \begin{Bmatrix} P_s \\ G \end{Bmatrix} \quad \text{Eq. 13-61}$$

The output from the solution is, however, expressed in terms of u_s and p .

In the frequency domain, we assume a harmonic function of the form $q = Q_0 e^{i\omega t}$ as a solution. Then the load integral becomes $\int G_0 e^{i\omega t} d\tau = G_0 / (i\omega) e^{i\omega t}$. The pressure is recovered from the relationship $p = i\omega q$.

Participation Factor Equations

Following Sung and Nefske, the participation factors are obtained from [Eq. 13-58](#) as follows, define the transformations:

$$\{u_s\} = [\Phi_s]\{\xi_s\} \quad \text{Eq. 13-62}$$

$$\{p\} = [\Phi_f]\{\xi_f\} \quad \text{Eq. 13-63}$$

where $[\Phi_s]$ are the uncoupled, undamped structural modes and $[\Phi_f]$ are the uncoupled, undamped, rigid-wall acoustic modes. The vectors $\{\xi_s\}$ and $\{\xi_f\}$ are the modal amplitudes. Substituting these relations into **Eq. 13-58** and pre-multiplying by the modal matrices, we get the equation

$$\begin{bmatrix} \Phi_s^T M_s \Phi_s & 0 \\ -\Phi_f^T A^T \Phi_s & \Phi_f^T M_f \Phi_f \end{bmatrix} \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} + \begin{bmatrix} \Phi_s^T B_s \Phi_s & 0 \\ 0 & \Phi_f^T B_f \Phi_f \end{bmatrix} \begin{Bmatrix} \dot{\xi}_s \\ \dot{\xi}_f \end{Bmatrix} + \begin{bmatrix} \Phi_s^T K_s \Phi_s & \Phi_s^T A \Phi_f \\ 0 & \Phi_f^T K_f \Phi_f \end{bmatrix} \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} = \begin{Bmatrix} \Phi_s^T P_s \\ \Phi_f^T P_f \end{Bmatrix} \quad \text{Eq. 13-64}$$

or

$$\begin{bmatrix} m_s & 0 \\ -a^T & m_f \end{bmatrix} \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} + \begin{bmatrix} b_s & 0 \\ 0 & b_f \end{bmatrix} \begin{Bmatrix} \dot{\xi}_s \\ \dot{\xi}_f \end{Bmatrix} + \begin{bmatrix} k_s & a \\ 0 & k_f \end{bmatrix} \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} = \begin{Bmatrix} Q_s \\ Q_f \end{Bmatrix} \quad \text{Eq. 13-65}$$

Define the following harmonic solution forcing frequency ω .

$$\{\xi\} = \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} e^{i\omega t} \quad \text{Eq. 13-66}$$

$$\{\dot{\xi}\} = i\omega \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} e^{i\omega t} \quad \text{Eq. 13-67}$$

$$\{\ddot{\xi}\} = -\omega^2 \begin{Bmatrix} \xi_s \\ \xi_f \end{Bmatrix} e^{i\omega t} \quad \text{Eq. 13-68}$$

For the bottom equation of **Eq. 13-65**, we get:

$$\omega^2 [a]^T \{\xi_s\} + [-\omega^2 [m_f] + i\omega [b_f] + [k_f]] \{\xi_f\} = \{Q_f\} \quad \text{Eq. 13-69}$$

Define $[Z_2]$ as

$$[Z_2] = [-\omega^2 [m_f] + i\omega [b_f] + [k_f]]^{-1} \quad \text{Eq. 13-70}$$

then

$$\{\xi_f\} = -\omega^2 [Z_2][\alpha]^T \{\xi_s\} + [Z_2]\{Q_f\} \quad \text{Eq. 13-71}$$

The fluid mode participation is defined as

$$[P_f] = [\Phi_f][\{\xi_f\}] \quad \text{Eq. 13-72}$$

where $\{\xi_f\}$ is the diagonalized vector of fluid modal amplitudes per excitation frequency.

The structure mode participation is defined as

$$[P_s] = -\omega^2 [\Phi_f][Z_2][a]^T [\{\xi_s\}] \quad \text{Eq. 13-73}$$

where $\{\xi_s\}$ is the diagonalized vector of structural modal amplitudes per excitation frequency.

The fluid load participation is defined as

$$\{P_l\} = [\Phi_f][Z_2]\{Q_f\} \quad \text{Eq. 13-74}$$

The fluid-structure panel participation is defined as

$$[P_p] = -\omega^2 [\Phi_f][Z_2][\Phi_f]^T [A]_{\text{panel}}^T [\Phi_s][\{\xi_s\}] \quad \text{Eq. 13-75}$$

The fluid-structure panel-grid participation is defined as

$$[P_g]_i = -\omega^2 [\Phi_f][Z_2][\Phi_f]^T \left[\left\{ A_b^T \right\} \right] [[\Phi_s]] [\xi_s] \quad \text{Eq. 13-76}$$

where

$$\left\{ A_b^T \right\}$$

are the columns extracted from the b-th boundary panel for panel grid i and $[\Phi_s]$ are the rows of the structural modal matrix corresponding to panel grid i.

Outputs

Typical MSC.Nastran outputs for structural displacement and acoustic pressure are shown in **Table 13-17**. Historically, MSC.Nastran is a unitless program. For this reason, the reference pressure is defaulted to a value $p_{\text{ref}} = 1$. For airborne sound in

MKS units the effective (root-mean-square) reference pressure rounded off is,
 $p_{\text{ref}} = 20. \mu Pa$. The reference pressure can be changed via the PREFDB command on
the FLSTCNT Case Control command.

Table 13-17

POINT-ID =		COMPLEX DISPLACEMENT VECTOR (MAGNITUDE/PHASE)						
FREQUENCY	TYPE	T1	T2	T3	R1	R2	R3	
2.500000E+00	G	0.0	0.0	3.847206E+01	0.0	0.0	0.0	
		0.0	0.0	351.9399	0.0	0.0	0.0	
5.000000E+00	G	0.0	0.0	1.744332E+02	0.0	0.0	0.0	
		0.0	0.0	4.3973	0.0	0.0	0.0	
7.500000E+00	G	0.0	0.0	9.960500E+01	0.0	0.0	0.0	
		0.0	0.0	357.3768	0.0	0.0	0.0	
1.000000E+01	G	0.0	0.0	1.593794E+02	0.0	0.0	0.0	
		0.0	0.0	344.4384	0.0	0.0	0.0	
1.250000E+01	G	0.0	0.0	1.258859E+02	0.0	0.0	0.0	
		0.0	0.0	353.4156	0.0	0.0	0.0	
POINT-ID = 100		COMPLEX ACOUSTIC RESULTS (MAGNITUDE/PHASE)						
FREQUENCY	TYPE	P	P(RMS)	DB	DB(A)			
2.500000E+00	S	6.993792E+02	4.945357E+02	1.508737E+02	5.947366E+01			
		19.9694						
5.000000E+00	S	2.007701E+04	1.419659E+04	1.800334E+02	9.563338E+01			
		48.1003						
7.500000E+00	S	2.067182E+03	1.461719E+03	1.602870E+02	8.288697E+01			
		0.4811						
1.000000E+01	S	1.989231E+03	1.406599E+03	1.599531E+02	8.955311E+01			
		349.6160						
1.250000E+01	S	1.370231E+03	9.688995E+02	1.567153E+02	9.331527E+01			
		355.9947						

Note that sound pressure level L_p re $0 \mu Pa$ is 94 dB greater than L_p re $1 Pa$.

The DB(A) column in the above table stands for A-weighting relative response in dB based on (A) electrical weighting networks for sound level meter. The table assumes a flat, diffuse-field (random-incidence) response for the sound-level meter and microphone. *For this column to be meaningful, the frequency is required to be measured in Hertz.*

Typical self explanatory MSC.Nastran mode participation printed output is:

Table 13-18

RESIDUAL ONLY- DYNAMIC LOADING-MODAL FREQ. RESP. USING ACSRCE				SUBCASE 100		
MODAL PARTICIPATION FACTORS FOR SUBCASE 100 (DLOAD ID = 100)						
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 2.50000E+00 HZ FOR MPF TYPE = FLUID						
*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	FLUID
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)
-1.0124E+04	-1.1711E+03	1.0192E+04	186.60	224	3	2.09053E+00
5.9970E+02	2.7852E+01	6.0034E+02	2.66	224	2	1.74037E+00
-3.1339E+01	2.2636E+00	3.1421E+01	175.87	224	4	4.89727E+00
1.9555E+01	-2.9385E+00	1.9775E+01	351.45	224	5	8.73536E+00
-5.9933E-02	2.1114E-01	2.1948E-01	105.85	224	6	8.89319E+00
RESIDUAL ONLY- DYNAMIC LOADING-MODAL FREQ. RESP. USING ACSRCE				SUBCASE 100		
MODAL PARTICIPATION FACTORS FOR SUBCASE 100 (DLOAD ID = 100)						
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 2.50000E+00 HZ FOR MPF TYPE = STRUCTURE						
*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	STRUCTURE
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)
-2.3247E+02	-1.1209E+01	2.3274E+02	182.76	224	9	2.22105E+01
1.1281E+02	-1.6752E+00	1.1283E+02	359.15	224	3	3.24082E+00
6.8895E+00	6.4339E-01	6.9195E+00	5.34	224	5	1.03650E+01
-3.6600E+00	-1.0253E-01	3.6615E+00	181.60	224	7	1.80836E+01
3.2577E+00	6.0409E-01	3.3132E+00	10.51	224	4	4.26122E+00
4.7262E-02	-1.0367E-02	4.8386E-02	347.63	224	8	1.82966E+01
-1.0310E-02	7.0109E-04	1.0334E-02	176.11	224	6	1.64346E+01
RESIDUAL ONLY- DYNAMIC LOADING-MODAL FREQ. RESP. USING ACSRCE				SUBCASE 100		
MODAL PARTICIPATION FACTORS FOR SUBCASE 100 (DLOAD ID = 100)						
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 2.50000E+00 HZ FOR MPF TYPE = PANEL : FIRSTONE						
*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	STRUCTURE
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)
-2.3253E+02	-1.1206E+01	2.3280E+02	182.76	224	9	2.22105E+01
8.5777E+01	1.0465E+00	8.5783E+01	0.70	224	3	3.24082E+00
8.0208E+00	5.9707E-01	8.0430E+00	4.26	224	5	1.03650E+01
6.9714E+00	3.6833E-01	6.9811E+00	3.02	224	4	4.26122E+00
-3.6285E+00	-1.0473E-01	3.6300E+00	181.65	224	7	1.80836E+01
-5.5836E-02	-1.2668E-03	5.5850E-02	181.30	224	8	1.82966E+01
4.8712E-03	4.4001E-04	4.8911E-03	5.16	224	6	1.64346E+01

Guidelines and Limitations

The following guide lines are associated with the capability:

1. If superelements are used, there can be only one fluid superelement.
2. Residual vector calculations are done for the structure only.
3. The CHACAB and CHACBR elements are structural elements and all their grids are structure grids
4. The CAABSF element is a fluid element and must attach to fluid grids.
5. When initially modeling large car body in white structures, it is recommend that the ACMODL Bulk Data entry not be used. For these types of models, it is a tuning tool based on fluid “skin” selection. It is highly recommended that the .PCH file of the fluid “skin” be utilized to plot the fluid-structure boundary.

13.5.2 Example 1

The base line and restart input files below represents a simple fluid-structure problem for the purpose of showing salient input. The model consists of a CQUAD4 and four CELAS2 to represent structure, a CHEXA to represent the fluid, and a CHACAB structural element to represent an acoustic absorber. In the base line case control, note the use of the FLSFSEL to select lower limits on the fluid and structure frequency range modes. Also note the use of FLSTCNT to set the reference pressure. Some fictitious random entries are input to demonstrate the XYPLOT features for structural and fluid grids. On large models of automobiles and aircraft, the modal solutions, especially on the structure, can take significant time and disk space. Also modal participation can, on large models, generate gigabytes of data. Thus no participation information is requested for the base line run, as participation results will be requested on the restart run. Since a data base is required, the MSC.Nastran base line submittal contains SCR=NO.

In the restart file, note the use of the ASSIGN and RESTART commands. Notice, also, that Case Control commands such as FLSFSEL, FLSTCNT, FREQ, SPC, METHOD(STRUCT) and METHOD(FLUID) are *copied* over from the base line run. To change or omit these would cause the restart logic to attempt to do the modal analysis over again. To calculate mode participation information, the FLSPOUT command is added to case control along with three set definitions directly above.

To define random data appropriate to modal participation, a different RANDOM command is provided. Also, a different set of XYPLOT commands are provided that are to be used for plotting of participation results.

For the restart run, the original bulk data entries are removed and some PANEL and SET1 entries are added to define panels for the participation run. Also, new RANDPS and RANDT1 entries are added for the participation analysis. Note that no TABRND1 is included because the original one from the base line run is used. Since none of the bulk data from the base line run is changed or deleted, the restart file does not contain any (/) bulk data entries.

Input File For Base Line Run

```
SOL 111 $ MODAL FREQUENCY RESPONSE
CEND
TITLE= FLUID-STRUCTURE INTERACTION
$
FLSFSEL LFREQFL=1.0,LFREQ=2.0
FLSTCNT PREFDB=20.-6
$
OLOAD=ALL
DISP(SORT2,PHASE)=ALL
STRESS(SORT2,PHASE)=ALL
RANDOM 10
```

```

$
SPC =1313
FREQ=200
METHOD(STRUCT)=30
METHOD(FLUID)=20
$
SUBCASE 100
DLOAD=100
$
SUBCASE 200
DLOAD=200
$
OUTPUT(XYPLOT)
XGRID=YES
YGRID=YES
XTITLE=frequency #14
YTITLE=psd grid 61
XYPLOT,XYPEAK DISP PSDF /61(T3)
XTITLE=frequency #15
YTITLE=auto grid 61
XYPLOT,XYPEAK DISP AUTO /61(T3)
XTITLE=frequency #16
YTITLE=psd grid 224
XYPLOT,XYPEAK DISP PSDF /224(T1)
XTITLE=frequency #17
YTITLE=auto grid 224
XYPLOT,XYPEAK DISP AUTO /224(T1)
$
BEGIN BULK
$ PRODUCE OUTPUT FORTRAN FILES
PARAM,POST,-1
$ DYNAMIC LOADING
$ ON STRUCTURE
DLOAD 100 1. 1. 1002 1. 1001
DLOAD 200 1. 1. 2002 1. 2001
RLOAD1 1001 19 1001
RLOAD1 2001 29 2001
DAREA 19 41 3 125.
DAREA 19 45 3 125.
DAREA 19 61 3 125.
DAREA 19 65 3 125.
DAREA 29 41 3 125.
DAREA 29 45 3 125.
DAREA 29 61 3 125.
DAREA 29 65 3 125.
$ ON FLUID
ACSRCE 1002 101 1001 1. 1.
ACSRCE 2002 102 1001 1. 1.
SLOAD 101 120 125.
SLOAD 101 124 125.
SLOAD 101 220 125.
SLOAD 101 224 125.
SLOAD 102 120 250.
SLOAD 102 124 250.
SLOAD 102 220 250.
SLOAD 102 224 250.
TABLED4 1001 0.0 1.0 0.0 1.E6
0.0 0.0 39.478420.0 ENDT
$ FRQUENCY RANGE

```

```

FREQ1  200      2.5      2.5      100
$ DAMPING
PARAM          G          0.05
PARAM          GFL        0.05
$ METHOD CARD FOR EIGEN VALUE
EIGRL          20
EIGRL          30
$THE STRUCTURAL POINTS
GRID   41          0.0      0.0      0.0
GRID   45          1.1      0.0      0.1
GRID   61          0.0      0.20     0.0
GRID   65          1.0      0.20     0.0
$THE STRUCTURAL POINTS FOR ABSORBER
GRID   261         0.0      0.2      0.001      12456
GRID   265         1.0      0.2      .001        12456
GRID   141         0.0      0.0      0.001      12456
GRID   145         1.1      0.0      .1001      12456
$THE STRUCTURAL ELEMENTS
CQUAD4  100      4444   41      45      65      61
CELAS2  1007     .0625  61      3
CELAS2  1008     .0625  65      3
CELAS2  1009     .0625  41      3
CELAS2  1010     .0625  45      3
$ STRUCTURAL PROPERTIES
PSHELL  4444     77      .05     77      1.0
MAT1    77      100.     .333    1.000
$
$ FLUID POINTS
GRID   200          0.0      0.2      0.001     -1
GRID   204          1.0      0.2      .001      -1
GRID   220          0.0      0.2      1.0      -1
GRID   224          1.0      0.2      1.0      -1
GRID   100          0.0      0.0      0.001     -1
GRID   104          1.1      0.0      .1001     -1
GRID   120          0.0      0.0      1.0      -1
GRID   124          1.0      0.0      1.0      -1
$ FLUID ELEMENTS
CHEXA   1000     15      100     104     124     120     200     204
        224     220
$ FLUID PROPERTIES
PSOLID  15      25
MAT10   25      1.      0.1
$
$ SPC-STRUCTURAL AND FLUID
SPC1    1313     1246   41      THRU    61
SPC1    1313     5      41      61     65     45
SPC1    1313     146     65
SPC1           1313     1      200
$ DEFINE ACOUSTIC ABSORBER
CHACAB  9      9      41      45     65     61     141     145
+       265     261
PACABS  9      YES     11     22     33     1.     1000.
TABLED1 11
+       2.      .15     50.     .15     100.    .15     150.    .15
+       200.    .15     250.    .15     300.    .15     ENDT
TABLED1 22
+       2.      -40.    50.     14.     100.    30.     150.    46.
+       200.    62.     250.    78.     300.    94.     ENDT
TABLED1 33

```

```

+      2.      1.      50.      1.      100.      1.      150.      1.
+      200.     1.      250.     1.      300.      1.      ENDT
$
$ RANDOM INPUT
RANDPS 10      100      100      4.          1
RANDPS 10      200      200      8.          1
RANDPS 10      100      200      2.          1
TABRND1 1
RANDT1 10      0.      100.      300.      100.      ENDT
RANDT1 10      4      0.      .004
$
ENDDATA

ASSIGN MASTER='1_a.MASTER'
RESTART VERSION=1 KEEP
SOL 111 $ MODAL FREQUENCY RESPONSE
CEND
SUBTITLE= FLUID-STRUCTURE INTERACTION
$
SET 29 = BNDY
SET 224 = 224
FLSPOUT FLUIDMP=ALL,GRIDFMP=224,SEPS=0.,ARS=0.,
        STRUCTMP=ALL,PANELMP=29,GRIDMP=ALL,
        OUTFMP=ALL,OUTSMP=ALL,PSORT=(ABSOLUTE,DESCENDING)
FLSFSEL LFREQFL=1.0,LFREQ=2.0
FLSTCNT PREFDB=20.-6, SKINOUT=ALL
$
RANDOM=50
$
SPC =1313
FREQ=200
METHOD(STRUCT)=30
METHOD(FLUID)=20
DISP(PLOT)=ALL
$
SUBCASE 100
DLOAD=100
$
SUBCASE 200
DLOAD=200
$
OUTPUT(XYPLOT)
XGRID=YES
YGRID=YES
$
$-1 plt 2nd sc
XTITLE=EXCITATION FREQUENCY FROM 2.5 to 250 hertz sc 200 #1
YTITLE=fluid mode pf at fluid point 224 for mode 2
XYPLOT,XYPEAK FMPF(2) MODE 200 /224
$-2,3 plt 1st & 2nd sc
XTITLE=EXCITATION FREQUENCY FROM 2.5 to 250 hertz default sc 100 200 #2 and #3
YTITLE= load pf at fluid point 224
XYPLOT,XYPEAK LMPF MODE /224
$-4 1st sc
XTITLE= natural modes from .4 to 2 hertz #4
YTITLE= panel mode pf at fluid point 224 for mode 9 sc 100

```

```

XYPLOT,XYPEAK PMPF(9,BNDY) MODE 100 /224
$-5 1st sc
XTITLE= natural modes from .4 to 2 hertz sc 100 #5
YTITLE=structure mode pf at fluid point 224 for mode 3
XYPLOT,XYPEAK SMPF(3) MODE 100 /224
$-6,7 1st & 2nd sc
XTITLE= natural modes from .4 to 2 hertz default sc 100 200 #6 and #7
YTITLE=grid panel mode pf at fluid point 224 for mode 9 grid 61
XYPLOT,XYPEAK GMPF(9,BNDY,61) MODE /224
$-8,9 1st & 2nd sc
XTITLE= natural modes from .4 to 2 hertz try 100,200 #8 and #9
YTITLE=grid panel mode pf at fluid point 224 for mode 9 grid 61
XYPLOT,XYPEAK GMPF(9,BNDY,61) MODE 100,200 /224
$
$-10
XTITLE=frequency #10
YTITLE=psd mode participation for grid 224
XYPLOT,XYPEAK FMPF(3) PSDF /224
$-11
XTITLE=frequency #11
YTITLE=auto mode participation for grid 224
XYPLOT,XYPEAK FMPF(3) AUTO /224
$-12
XTITLE=frequency #12
YTITLE=psd grid mode participation for grid 224 firstone at grid 61 first decend
mode
XYPLOT,XYPEAK GMPF(9,BNDY,61) PSDF /224
$-13
XTITLE=frequency #13
YTITLE=auto grid mode participation for grid 224 firstone at grid 61 first decend
mode
XYPLOT,XYPEAK GMPF(9,BNDY,61) AUTO /224
$ ----- expect 2 plots(4th group)
$
BEGIN BULK
$ DEFINE STRUCTURAL PANELS
PANEL          BNDY          777
SET1    777    SKIN
$ NEW RANDOM INPUT FOR PARTICIPATION
RANDPS  50      100      100      2.          1
RANDPS  50      200      200      5.          1
RANDPS  50      100      200      1.          1
RANDT1  50      6        0.        .009
$
ENDDATA

```

BW Algorithm: Discussion and Example

A new method (BW) of calculating fluid structure interface has been implemented. Speed improvements are such that an all automobile model that consists of 120,000 - 150,000 shell elements and 15,000 - 20,000 fluid elements now takes only about 60 seconds for calculating the fluid/structure interface matrix.

You can request a .pch file that represent fluid "skin" (param,skinout,punch) in order to check interface accuracy. A utility program is provided for viewing and inspecting the completeness and accuracy of the "skin" coverage (see the following example).

The format for the new BW method is:

1	2	3	4	5	6	7	8	9	10
ACMODL	INTER	INFOR	FSET	SSET	NORMAL	METHOD	SKNEPS	DSKNEPS	
	INTOL	ALLSSET	SRCHUNIT						

Example

ACMODL							.25		

For the BW method, no ACMODL entry is necessary if defaults are used (suggested for first run). A special license is required for this entry.

In most cases, the BW (Body in White) method provides a good fluid/structure interface matrix. However, changes in the interface can be made by either adding or deleting structural elements in the interface calculation.

A set of utility programs are provided to aid in viewing of the elements in the interface.

The purpose of the BW algorithm is to overcome the difficulties of many structural elements of various sizes overlapping each other. Also, this algorithm should be somewhat insensitive to small inadvertent holes in the structure.

The algorithm proceeds as follows:

1. Use the current search algorithm to locate the fluid free faces and the corresponding structural element faces.
2. For a fluid free face and its list of structural element faces (that were determined by boxing normal to the fluid element) do as follows:
 - a. For each fluid free face establish a face coordinate system.
 - b. Determine the resultant pressure force for each grid on the fluid element by the relationship

$$R_i = \int_S [N_f] dS \{p_i = 1; p_j = 0\} = 1, N \text{ grid/elem}$$

- c. Resolve this resultant pressure force for a unit grid pressure to the grids of the fluid element by the expression (determined by virtual work)

$$\{F_i\} = \int_S \{N_f\}^T [N_f] dS \{p_i\}$$

- d. Using the origin of the free fluid face, determine the center of pressure (X_{P_i}, Y_{P_i}) . The relationship will be of the form

$$X_{P_i} = \sum_j^{\text{grids}} \frac{F_i}{R_i} (X_j - X_0)$$

$$Y_{P_i} = \sum_j^{\text{grids}} \frac{F_i}{R_i} (Y_j - X_0)$$

- e. Using rigid relations to consider only a unit motion normal to the fluid face with the appropriate moment relationships, determine the resulting load distribution at the grids of each of the structural elements. The area of each structural element projected normal to the fluid element will be used as a weighting function. The expression is of the form:

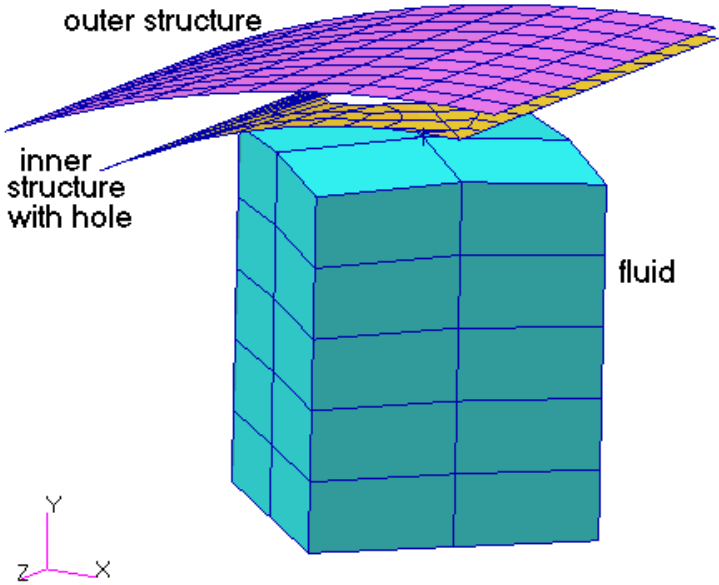
$$\{F_j\} = [W][R]([R]^T[W][R])^{-1} \begin{Bmatrix} R_i \\ 0 \\ 0 \end{Bmatrix}$$

where $\{F_j\}$ is the vector of resulting load distribution at the grids of each of the j structural elements. $[W]$ is a diagonal weighting matrix. $[R]$ is the rigid transformation matrix.

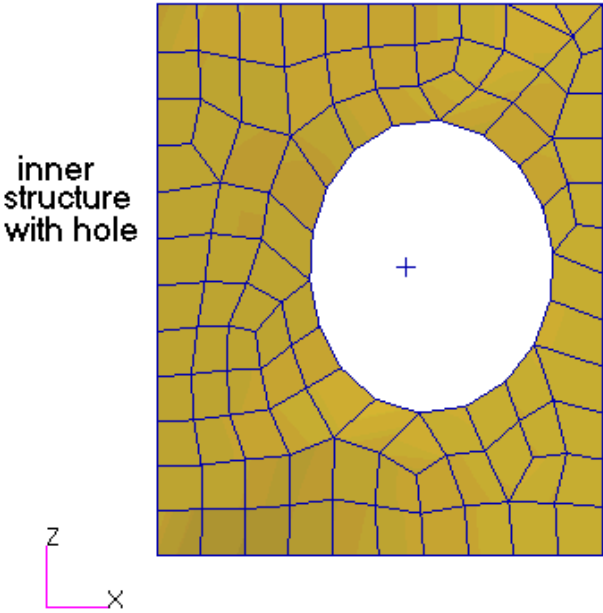
- f. Loop over each grid of the fluid element and accumulate the forces at the structural grids.
- g. Note, with this algorithm, we do not worry if a structural element is only partially wetted by the fluid. We always require rigid body equilibrium.
3. Repeat for the next fluid element and its associated group of structural elements. Accumulate the forces at the structural grids.

13.5.3 Example 2

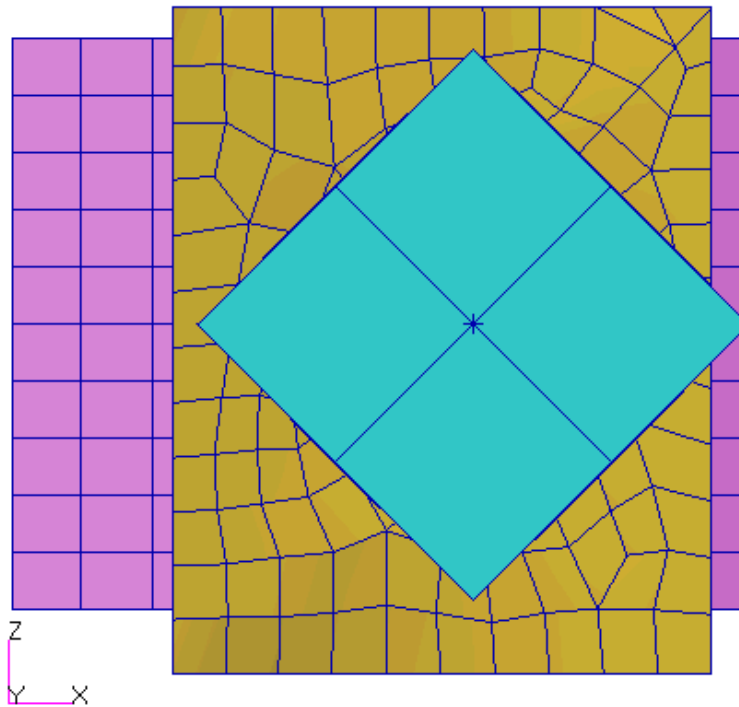
The model shown below, "acoustic1.dat", is a square column of fluid with a round top which interacts with an inner structural plate with a hole in it and an outer structural plate:



The inner plate has a hole as shown:



The relative location of the fluid face and the structural plates is shown:



When the model is run, and `PARAM,SKINOUT,PUNCH` is used, the fluid and structure interface elements and grids found by the search are written to the `acoustic1.pch` file shown:

```

$ *****
$ FSI - IDS OF FLUID ELEMENTS AT INTERFACE
$ *****
$ This bulk data entry may be referenced from the FSET field on the
$ ACMODL bulk data entry (with the ELEMENTS option) to remove unwanted
$ fluid faces from the fluid-structure interface.
SET1    1          5001    5001    5001    5002    5002    5003    5003
          5004
$ *****
$ FSI - IDS OF STRUCTURE ELEMENTS AT INTERFACE
$ *****
$ This bulk data entry may be referenced from the SSET field on the
$ ACMODL bulk data entry (with the ELEMENTS option) to remove unwanted
$ structure faces from the fluid-structure interface.
SET1    2          1041    1037    1038    1040    1039    1083    1094
          1092    1091    1090    1082    1095    1093    1063    1081
          1088    1064    1065    1062    1066    1053    1054    1055
          1087    1052    1056    1057    1086    1051    1096    1058
          1019    1028    1085    1073    1050    1072    1076    1084
          1020    1070    1059    1013    1077    1071    1060    1012
          1010    1061    1009    1008    1046    1047    1007    1006
          1044    1045
$ *****
$ FSI - IDS OF FLUID GRIDS AT INTERFACE
$ ***** $
This bulk data entry may be referenced from the FSET field on the
$ ACMODL bulk data entry (with the GRIDS option) to remove unwanted
$ fluid faces from the fluid-structure interface.
SET1    3          5001    5002    5003    5004    5005    5006    5007
          5008    5009    5010    5011    5012    5013    5016
$ *****
$ FSI - IDS OF STRUCTURE GRIDS AT INTERFACE
$ *****
$ This bulk data entry may be referenced from the SSET field on the
$ ACMODL bulk data entry (with the GRIDS option) to remove unwanted
$ structural faces from the fluid-structure interface.
SET1    4          1004    1005    1006    1007    1008    1009    1010
          1017    1018    1027    1028    1029    1030    1031    1032
          1040    1041    1042    1045    1047    1048    1049    1050
          1051    1051    1053    1054    1055    1056    1057    1058
          1059    1060    1061    1062    1063    1064    1065    1066
          1071    1072    1073    1074    1075    1078    1079    1080
          1081    1082    1084    1085    1086    1087    1088    1093
          1094    1095    1096    1097    1098    1099    1101    1102
          1103    1104    1105    1106    1107    1108    1109    1110
          1111    1112    1113    1114    1115    1116    1117    1118
          1119    1120    1121    1122    1123    1125    1126    1127
          1128    1129

```

These element and grid SET1 Bulk Data entry lists serve two purposes. First, they can be used in a finite element preprocessor as an accuracy check by graphically displaying which elements are used in the analysis, and second they can limit or expand what elements or grids are used in fluid/structure interface.

To check the fluid/structure interface, MSC supplies a perl program named "acnaspat.pl" which reads the above .pch file and writes a MSC.Patran session file. The perl script is located at "/msc2004/util". To run the script on UNIX, copy it to your local directory and type, for instance:

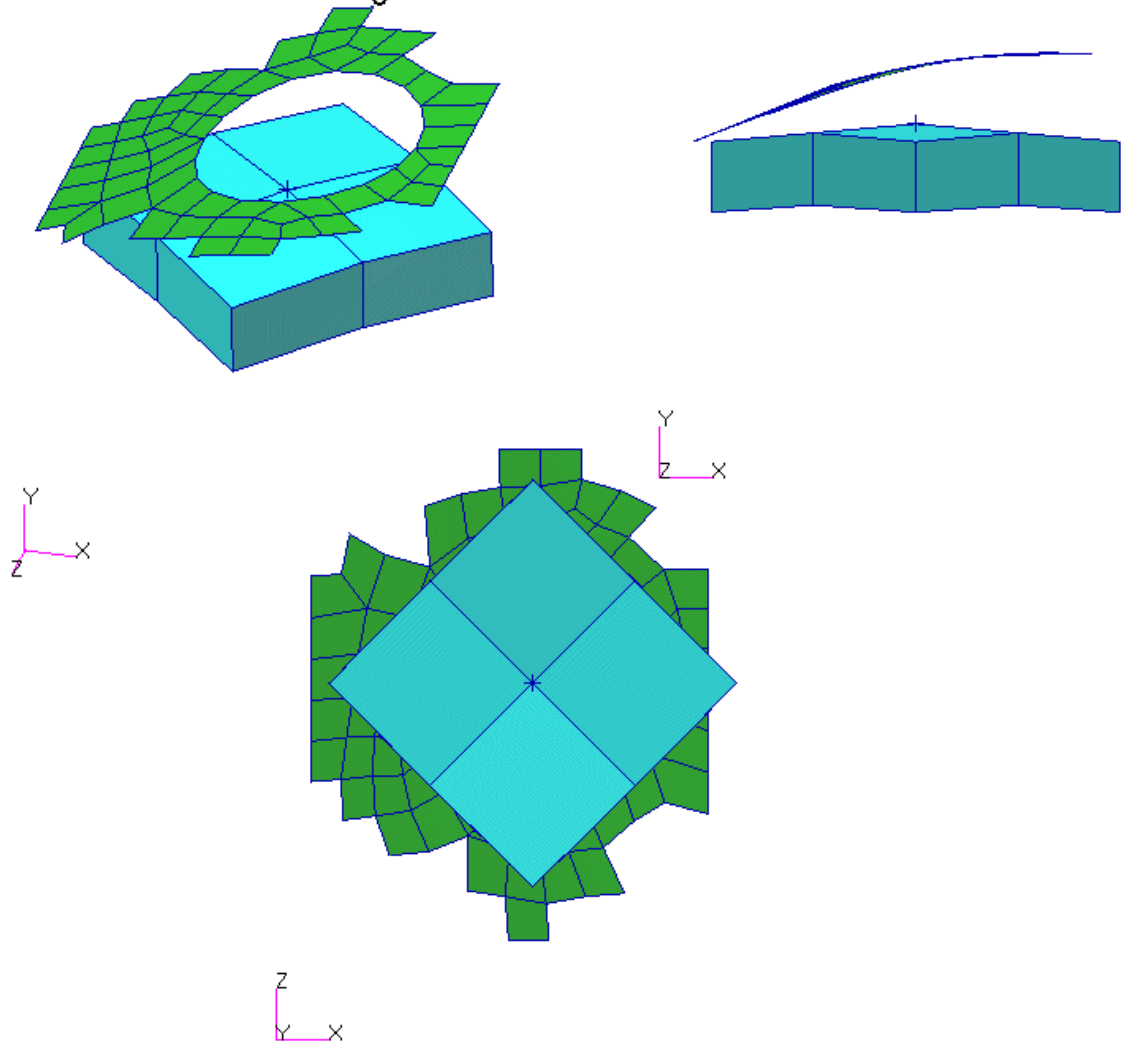
```
/msc2004/util/acnaspat.pl acoustic1.pch
```

which creates a MSC.Patran session file named "acoustic1_flstgrp.ses", which when run in MSC.Patran will create two groups in MSC.Patran:

acoustic1_fluid	the fluid interface elements and grids
acoustic1_structure	structure interface elements and grids

These two groups are shown below:

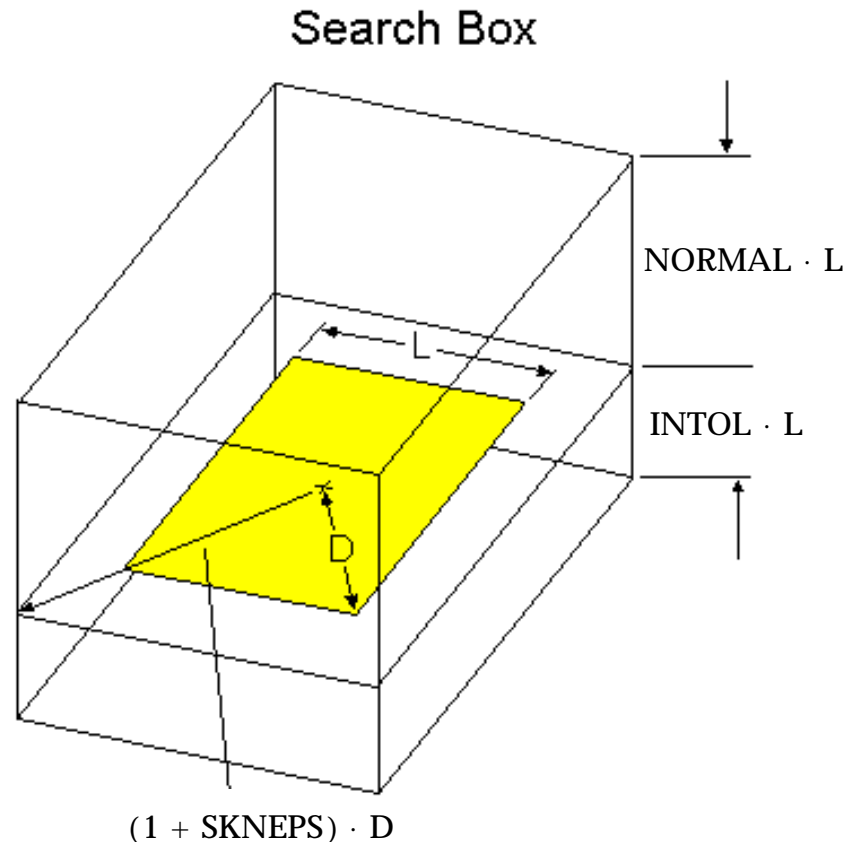
interface elements and grids



Two things are indicated:

- extra structure elements were selected that do not project directly onto the fluid faces,
- only elements from the inner structural plate were selected.

Which structural elements to be included in the fluid/structure interface can be adjusted in one of two ways. The first is by increasing the size of the search box by changing the defaults on the NORMAL, INTOL, and SKNEPS fields of the ACMODL Bulk Data entry.



- $\text{NORMAL} \cdot L$ -- Search box height in positive normal direction from fluid face, where L is smallest fluid face side length,
- $\text{NTOL} \cdot L$ -- height in negative normal direction,
- $(1 + \text{SKNEPS}) \cdot D$ -- in-plane width added to fluid face dimensions where D is the distance from the center of the fluid face to the grid point.

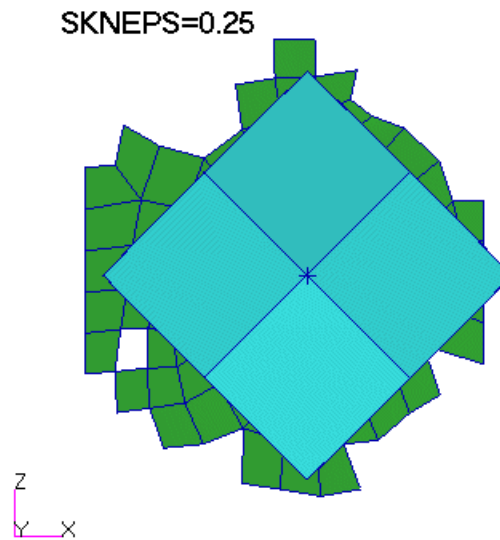
If the search cannot find any structural faces within the search box, it will widen the search box by using the value of the `DSKNEPS` field on the `ACMODL` Bulk Data entry.

Structural faces, whose centroids fall within the search box, are included in the fluid/structure interface. The effect of the structural face in the fluid/structure interface matrix is weighed according to areas, angles and distances between it and the fluid face that it found.

In this case, looking at the interface elements, the normal distance of the search box appears good, but too many structural elements on the sides of the fluid faces are being selected. By adding an `ACMODL` Bulk Data entry with `SKNEPS` decreased from the default of 0.5 to 0.25:

```
acmod1 , , , , , , , 0.25
```

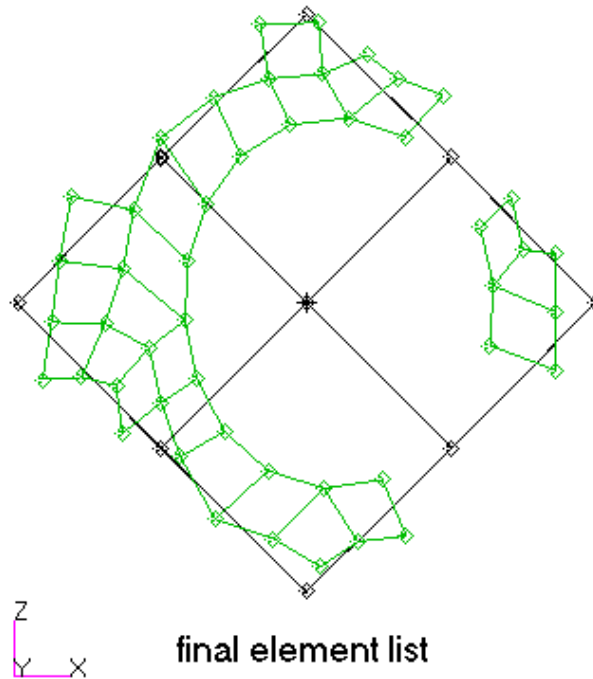
you get the following interface elements:



The structural elements are closer to what is wanted, but there are still too many.

The second way to control which elements are used in the fluid/structure interface is to use the INFOR="elements" or "grids" and the FSET and SSET fields on the ACMODL Bulk Data entry.

First the elements not wanted are removed from the "acoustic2_struct" group in MSC.Patran. For this example, any structure element with less than 50% of it's area projected onto the fluid faces is removed. The final structural interface elements are shown:



In MSC.Patran, the "acoustic2_struct" group is then displayed alone, its elements selected with the Erase tool, the element list highlighted, and then copied and pasted into a file, "element1.txt" in this case. The file's contents as shown below:

```
Elm 1008 1013 1044 1045 1050:1059 1062:1064 1070 1072 1082 1085 1090 1091
```

Another Perl program from the same install directory is now used to change the above MSC.Patran element or node list format into a MSC.Nastran SET1 format. To run the script type:

```
/msc2004/util/acpatnas.pl element1.txt
```

which creates a file named "element1.set1" in the following form:

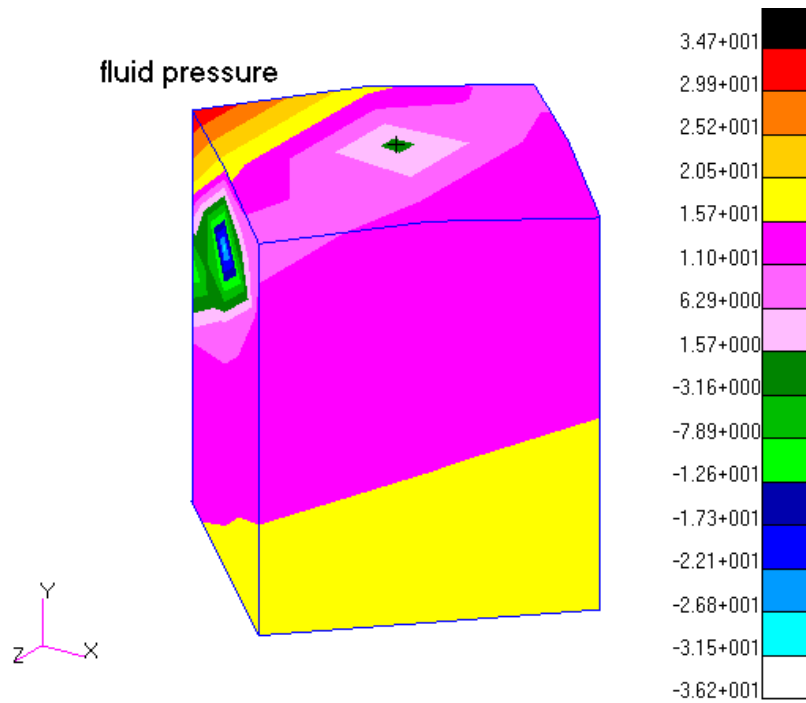
```
SET1          1    1008    1013    1044    1045    1050    1051    1052
              1053    1054    1055    1056    1057    1058    1059    1062
              1063    1064    1070    1072    1082    1085    1090    1091
```

The SET1 SID is changed from "1" to "2" so as not to conflict with the fluid element SET1 from the "acoustic1.pch" file. When these two SET1 Bulk Data entries are put in the MSC.Nastran file along with:

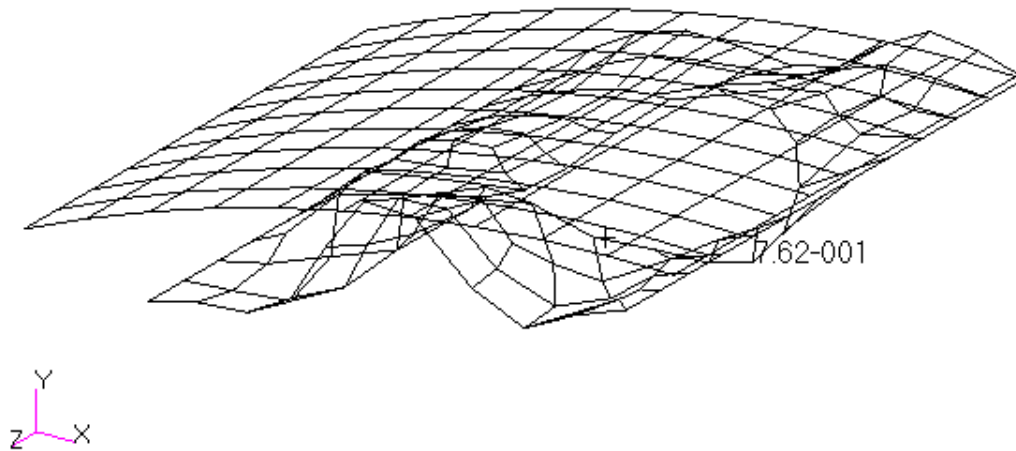
```
acmod1,,elements,1,2
```

then only the above structural elements are used in the fluid/structure interface.

The following plots show the fluid pressure distribution and the structural deflection.



structural deformation



The model is loaded with acoustic power on the upper left hand corner of the fluid, thus the high pressure in that area. The effect of the hole in the inner structure can be seen as a slight reduction in fluid pressure at the hole.

The structural deformation shows that the outer structure is unaffected by the pressure, as expected.

This concludes the example.

Parallel Structural Element Meshes in the Search Box:

As noted above, the outer parallel structural surface did not have any elements selected for inclusion in the interface, even though some were in the search box. This is because parallel surfaces are ignored by using a connectivity technique. The first structural element face found in the search box is the one closest to the searching fluid face. All elements connected to this first element are checked whether they are in the search box. Once an element is found to be outside of the box, it is no longer used for the connectivity check. This tends to eliminate parallel element meshes unless the connection is also in the search box, in which case the parallel elements must be manually removed using the SET1 technique above.

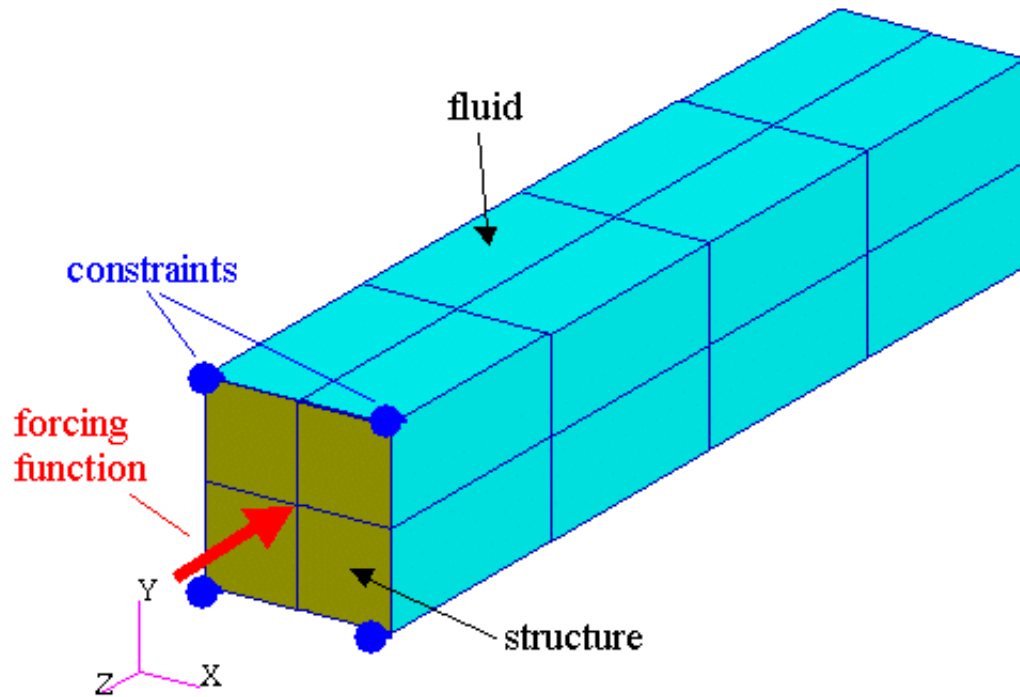
A problem occurs with this technique if a connectivity discontinuity occurs in the search box. Only one side of the discontinuity will be selected, causing elements to be missed in the interface search. The work-around for this is to use the ALLSET="YES" field on the ACMODL entry in which case all elements or grids in the SET1 entries are forced to be included in the fluid/structure interface whether the search finds them or not. You must also set INTER="DIFF", INFOR="ELEMENTS" or "GRIDS" and FSET and SSET must be properly filled out. This method forces an interface between the listed structural elements and the nearest fluid faces.

Absolute Versus Relative Search Box Dimensions:

If SRCHUNIT="ABS", the box dimensions are equal to the value of NORMAL, INTOL, and SKNEPS, rather than times a fluid face dimension. This is useful when the distance between the structural interface and the fluid interface is relatively constant instead of relative to the fluid element size.

13.5.4 Modal Participation Example

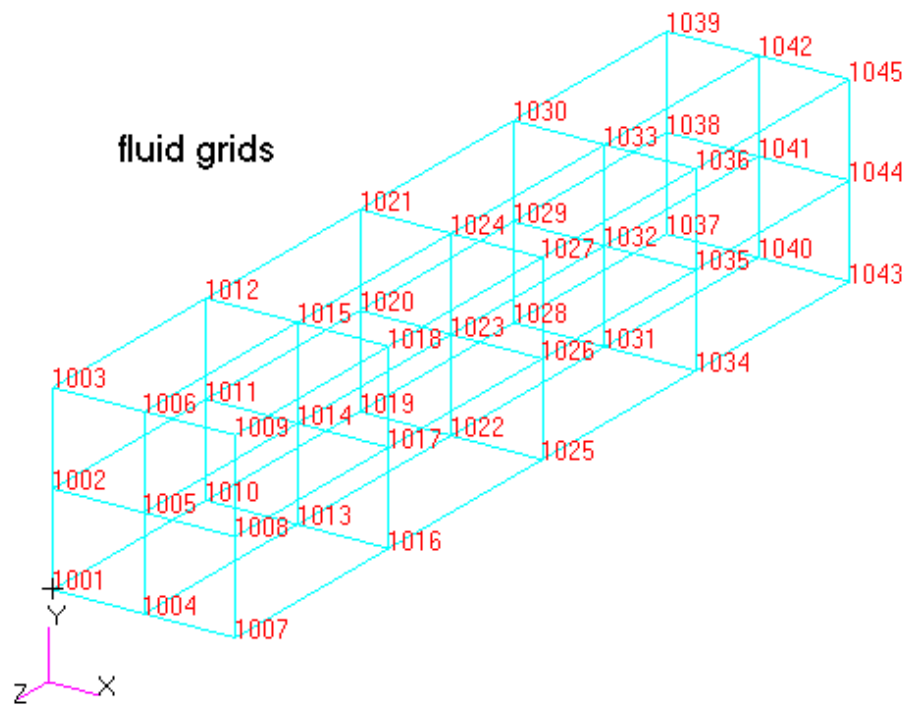
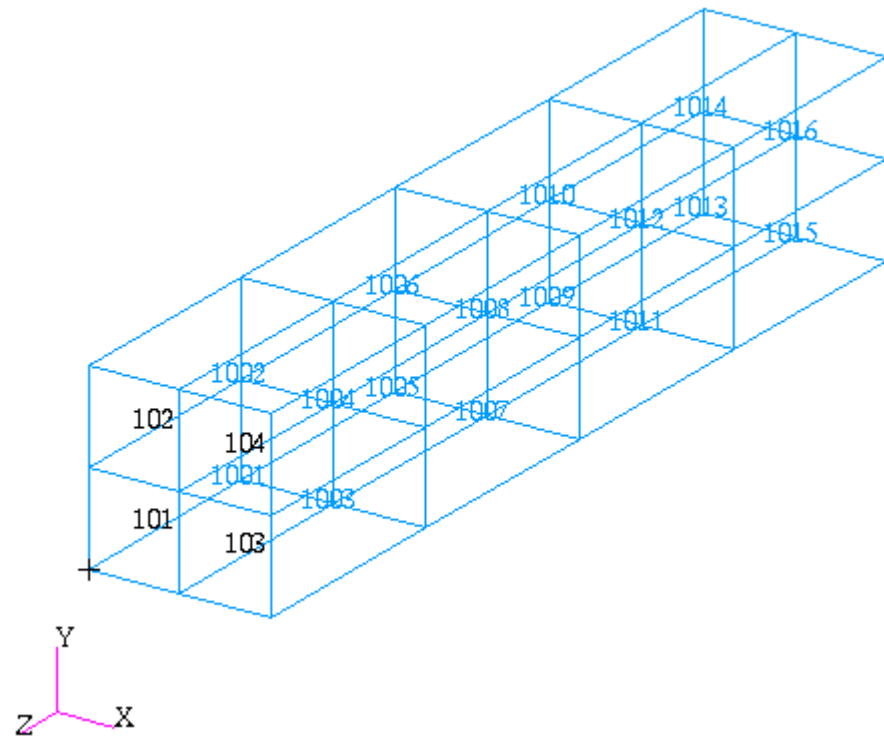
The problem is to find the modal participation factors in a damped frequency response analysis of a 20 in. x 20 in. x 100 in. air volume that has a 20 in. x 20 in. x 0.2 in. thick aluminum plate at one end loaded with a sinusoidal force at its center.

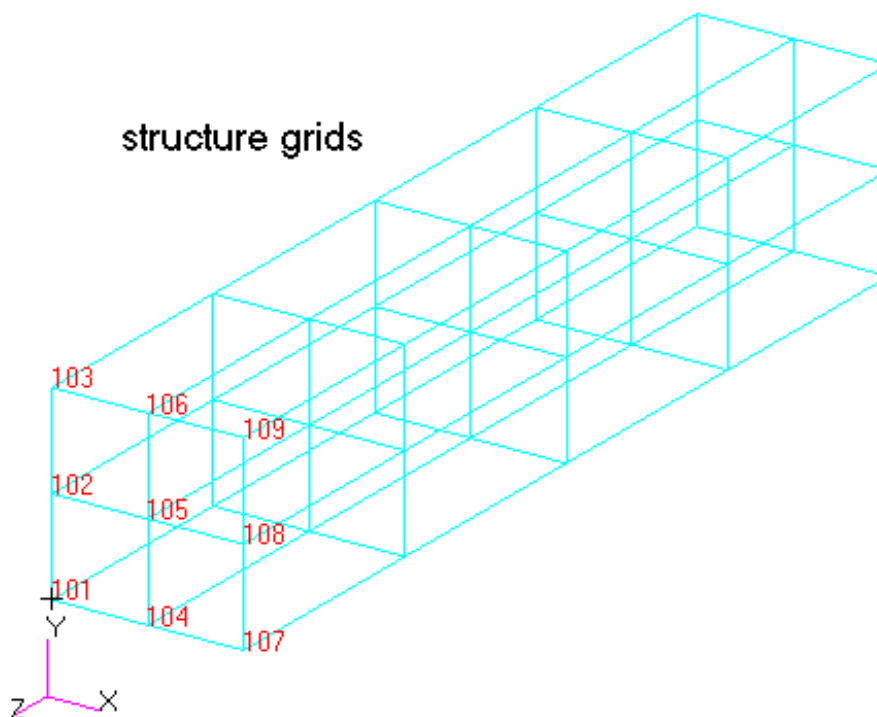


The following information will be found:

- fluid participation factors -- these are the effects of the fluid mode on fluid grid pressures
- structural participation factors -- these are the effect of the structural modes on fluid grid pressures
- panel participation factors -- these are the effect of a group of structural grids on fluid grid pressures
- panel grid participation factors -- these are the individual effects of each panel grid on fluid grid pressures

The model grid and elements are shown:





The modal frequency analysis uses five structural modes and 30 fluid modes. The case control commands to select fluid, structural, panel, and panel grid modal participation factors are:

```
flspout fluidmp=30 structmp=5 gridmp=all
```

Panel participation factors are automatically calculated when a STRUCTMP parameter is present for all the panels defined in the bulk data. This model has a panel defined with the following bulk data entries:

```
$
panel, leftside, 1
set1, 1, 101, 102, 103, 104, 105, 106
$
```

which are the grids that surround structural elements 101 and 102. The GRIDMP=all parameter on the FLSPOUT Case Control command requests participation factors for all of these panel grids.

A portion of the model is shown:

```
SOL      111
CEND
TITLE = Modal frequency response - fluid participation factors
DISP(sort2) = ALL
dload=10
spc=1
freq=15
```

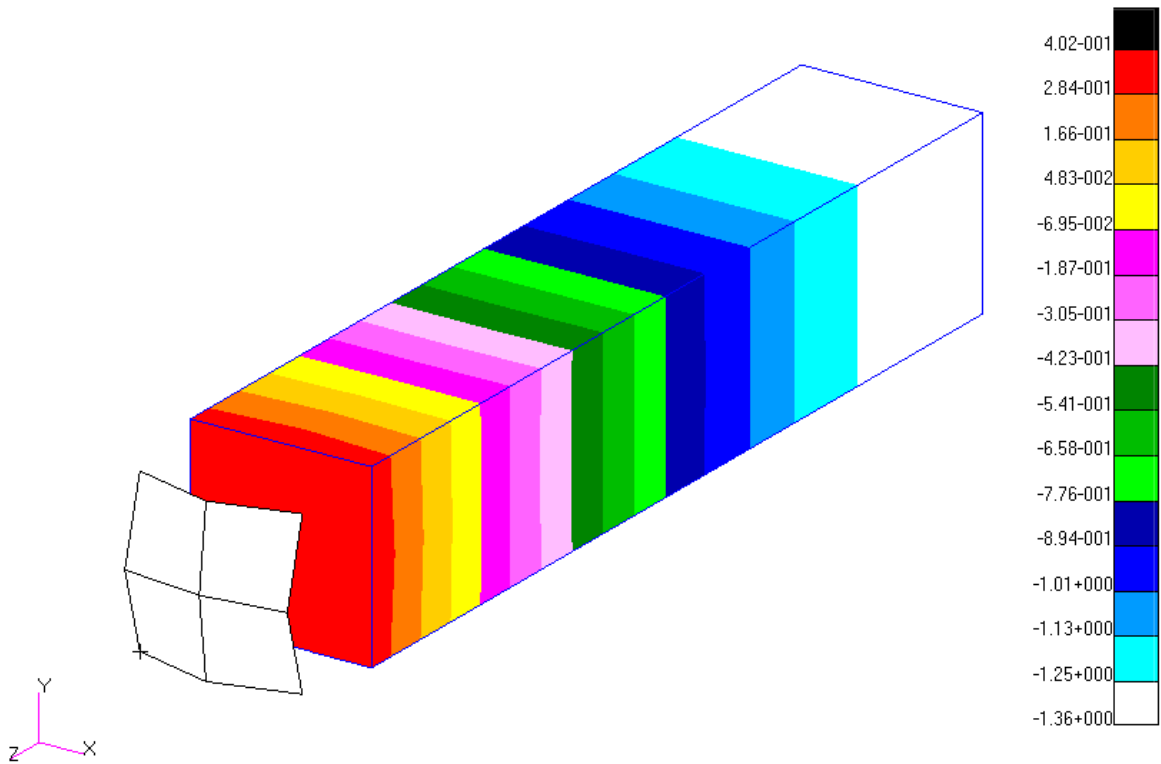
```

method(structure)=98
method(fluid)=99
flstcnt prefdb=2.9e-9
flspout fluidmp=30 structmp=5 gridmp=all
BEGIN BULK
acmodl ident
PARAM,POST,-1
PARAM AUTOSPC YES
$
rload1,10,100,,,100100
darea,100,105,3,15.0
tabled1,100100
      0.      1.0      1000.  1.0      endt
freq1,15,10.,1.0,99
eigr1,98,,,5
eigr1,99,,,30
$
panel,leftside,1
set1,1,101,102,103,104,105,106
$
$ Structural model
$
$
PSHELL 1      1      .2      1      1
MAT1,1,1.+7,,.3,2.54-4, 0.01
CQUAD4  101      1      101      102      105      104      0.      0.
CQUAD4  102      1      102      103      106      105      0.      0.
CQUAD4  103      1      104      105      108      107      0.      0.
CQUAD4  104      1      105      106      109      108      0.      0.
GRID    101      0.      0.      0.
GRID    102      0.      10.     0.
GRID    103      0.      20.     0.
GRID    104      10.     0.      0.
GRID    105      10.     10.     0.
GRID    106      10.     20.     0.
GRID    107      20.     0.      0.
GRID    108      20.     10.     0.
GRID    109      20.     20.     0.
$
SPC     1      109      3      0.0
SPC     1      103      3      0.0
SPC     1      101      123     0.0
SPC     1      107      23      0.0
$
$ Fluid model
$
PSOLID  1001      1001      PFLUID
MAT10,1001,,1.21e-7,1.3e4,.02
CHEXA   1001      1001      1001      1002      1005      1004      1010      1011      +      A
+       A 1014      1013
.
.
.
GRID    1044      20.     10.     -100.     -1
GRID    1045      20.     20.     -100.     -1
$
ENDDATA

```

Participation factors are available at all of the frequencies request, in this case 10 through 110 by 1. For illustrational purposes only the 39 hz results will be shown. This frequency is close to the first structural natural frequency.

A plot of the 39 hz response is shown below which is a composite of two pictures; a deformation plot of the structure and a pressure contour plot on the fluid elements, slightly offset from the structure for visual purposes:



The fluid modal participation factors for 39 hz are partially shown below:

MODAL PARTICIPATION FACTORS FOR SUBCASE 1 (DLOAD ID = 10)
 DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = FLUID

*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	FLUID
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1042	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1008	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1002	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1005	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1039	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1003	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1038	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1009	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1043	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1045	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1004	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1040	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1007	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1037	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1041	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1006	2	6.66808E+01
7.8079E-01	-6.6175E-01	1.0235E+00	319.72	1001	2	6.66808E+01
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1044	2	6.66808E+01
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1010	1	6.30946E-06
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1038	1	6.30946E-06
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1011	1	6.30946E-06
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1009	1	6.30946E-06
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1025	1	6.30946E-06
.
-5.9512E-04	4.9336E-04	7.7303E-04	140.34	1024	29	7.71939E+02
-4.6513E-04	3.8560E-04	6.0418E-04	140.34	1004	28	7.53632E+02
-4.6513E-04	3.8560E-04	6.0418E-04	140.34	1044	28	7.53632E+02
4.6513E-04	-3.8560E-04	6.0418E-04	320.34	1008	28	7.53632E+02
4.6513E-04	-3.8560E-04	6.0418E-04	320.34	1002	28	7.53632E+02
-4.6513E-04	3.8560E-04	6.0418E-04	140.34	1038	28	7.53632E+02
-4.6513E-04	3.8560E-04	6.0418E-04	140.34	1006	28	7.53632E+02
4.6513E-04	-3.8560E-04	6.0418E-04	320.34	1042	28	7.53632E+02
4.6513E-04	-3.8560E-04	6.0418E-04	320.34	1040	28	7.53632E+02

The factors are ordered by magnitude. For the fluid participation factors, the ARF value on the FLSPOUT Case Control command is used to determine which factors will be printed. The default is ARF=0.95. This means only values above 95% of the highest value for each mode are printed. 328 factors were printed for 39 hz above, and also a similar number for each other 100 frequencies, totaling approximately 33,000 lines.

For the fluid modes, since participation factors are the modal solutions times the mode shapes, and since the first fluid mode is a rigid body mode, all of the participation factors for the first mode have the same value and thus are unaffected by ARF so they are all printed.

Also for this example, because of the simple shape of the model and the symmetry of the loading and constraints, various parts of the fluid model tend to have the same eigenvector value, so more than usual are printed.

If only the first maximum value is wanted from each of the 30 fluid modes, use ARF = 0.999:

*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	FLUID
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1042	2	6.66808E+01
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1010	1	6.30946E-06
1.6373E-01	-1.3631E-01	2.1304E-01	320.22	1008	3	1.43346E+02
-9.1451E-02	7.5979E-02	1.1890E-01	140.28	1045	4	2.32942E+02
3.8657E-02	-3.2104E-02	5.0249E-02	320.29	1024	5	2.86691E+02
7.1325E-03	-5.9130E-03	9.2648E-03	320.34	1045	27	7.53632E+02
-4.3311E-03	3.5905E-03	5.6258E-03	140.34	1032	30	7.71939E+02
2.5097E-03	-2.0806E-03	3.2600E-03	320.34	1019	25	7.30922E+02
2.4613E-03	-2.0404E-03	3.1971E-03	320.34	1004	26	7.30922E+02
-2.0102E-03	1.6665E-03	2.6112E-03	140.34	1041	23	7.19823E+02
-1.8705E-03	1.5507E-03	2.4297E-03	140.34	1008	24	7.19823E+02
-1.4622E-03	1.2122E-03	1.8994E-03	140.34	1007	22	7.16728E+02
-5.9823E-04	4.9595E-04	7.7708E-04	140.34	1024	21	7.16728E+02
-5.9512E-04	4.9336E-04	7.7303E-04	140.34	1011	29	7.71939E+02
-4.6513E-04	3.8560E-04	6.0418E-04	140.34	1004	28	7.53632E+02

Only 15 fluid mode values are shown because the other 15 have no values above the FEPS=1.0e-11 cutoff on the FLSPOUT Case Control command.

To list only fluid grid 1041 fluid participation values, use GRIDFMP:

*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	FLUID
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)
-7.8079E-01	6.6175E-01	1.0235E+00	139.72	1041	2	6.66808E+01
-6.9499E-01	5.5365E-01	8.8856E-01	141.46	1041	1	6.30946E-06
1.6373E-01	-1.3631E-01	2.1304E-01	320.22	1041	3	1.43346E+02
-9.1451E-02	7.5979E-02	1.1890E-01	140.28	1041	4	2.32942E+02
3.8657E-02	-3.2104E-02	5.0249E-02	320.29	1041	5	2.86691E+02
-7.1325E-03	5.9130E-03	9.2648E-03	140.34	1041	27	7.53632E+02
4.3311E-03	-3.5905E-03	5.6258E-03	320.34	1041	30	7.71939E+02
2.5097E-03	-2.0806E-03	3.2600E-03	320.34	1041	25	7.30922E+02
2.4138E-03	-2.0011E-03	3.1354E-03	320.34	1041	26	7.30922E+02
-2.0102E-03	1.6665E-03	2.6112E-03	140.34	1041	23	7.19823E+02
1.4622E-03	-1.2122E-03	1.8994E-03	320.34	1041	22	7.16728E+02

The participations above show a wide variation of fluid mode inputs into the fluid solution. This is because 39 hz is a structural natural frequency. The single structural mode is translated at the fluid/structure interface into an equivalent set of fluid modes.

The structural fluid participation factors should show just one structural mode contributing to the majority of the fluid response because 39 hz is close to a structural natural frequency:

MODAL PARTICIPATION FACTORS FOR SUBCASE 1 (DLOAD ID = 10)				DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = STRUCTURE			
*** MODE PARTICIPATION FACTOR (MPF) ***				FLUID	MODE	STRUCTURE	
REAL	IMAGINARY	MAGNITUDE	PHASE	POINT	NUMBER	NATURAL FREQ. (HZ)	
-1.3650E+00	1.1229E+00	1.7675E+00	140.56	1037	1	3.96034E+01	
-1.3650E+00	1.1229E+00	1.7675E+00	140.56	1043	1	3.96034E+01	
-1.3650E+00	1.1229E+00	1.7675E+00	140.56	1039	1	3.96034E+01	
-1.3650E+00	1.1229E+00	1.7675E+00	140.56	1045	1	3.96034E+01	
-1.3648E+00	1.1228E+00	1.7673E+00	140.56	1040	1	3.96034E+01	
-1.3648E+00	1.1228E+00	1.7673E+00	140.56	1044	1	3.96034E+01	
-1.3648E+00	1.1228E+00	1.7673E+00	140.56	1038	1	3.96034E+01	
-1.3648E+00	1.1228E+00	1.7673E+00	140.56	1042	1	3.96034E+01	
-1.3647E+00	1.1226E+00	1.7671E+00	140.56	1041	1	3.96034E+01	
-1.5697E-05	2.0986E-04	2.1045E-04	94.28	1041	5	9.82372E+01	
-1.5689E-05	2.0976E-04	2.1034E-04	94.28	1040	5	9.82372E+01	
-1.5689E-05	2.0976E-04	2.1034E-04	94.28	1042	5	9.82372E+01	
-1.5689E-05	2.0976E-04	2.1034E-04	94.28	1038	5	9.82372E+01	
-1.5689E-05	2.0976E-04	2.1034E-04	94.28	1044	5	9.82372E+01	
-1.5682E-05	2.0965E-04	2.1024E-04	94.28	1039	5	9.82372E+01	
-1.5682E-05	2.0965E-04	2.1024E-04	94.28	1043	5	9.82372E+01	
-1.5682E-05	2.0965E-04	2.1024E-04	94.28	1037	5	9.82372E+01	
-1.5682E-05	2.0965E-04	2.1024E-04	94.28	1045	5	9.82372E+01	

Note that the only other mode besides structural mode 1 is mode 5 which has insignificant values.

The response for fluid grid 1041 at 39 hz is 1.76 psi:

POINT-ID = 1041		COMPLEX ACOUSTIC PRESSURE RESULTS			
(MAGNITUDE/PHASE)					
FREQUENCY	TYPE	P	P(RMS)	DB	DB(A)
3.90000E+01	S	1.767275E+00 140.5534	1.249652E+00 140.5534	1.756981E+02 140.5534	1.405334E+02 140.5534

Note that this is the value of the single structural modal participation factor, and is also the sum of the fluid factors (after accounting for phase).

The panel is made of the six grids that bound the left side of the structure. The participations for this panel are shown:

```

MODAL PARTICIPATION FACTORS FOR SUBCASE          1 (DLOAD ID =      10)
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = PANEL :
LEFTSIDE

      *** MODE PARTICIPATION FACTOR (MPF) ***
      REAL      IMAGINARY      MAGNITUDE      PHASE      FLUID      MODE      STRUCTURE
      POINT     NUMBER     NATURAL FREQ. (HZ)
-1.1672E+00  9.6021E-01    1.5114E+00  140.56    1045      1      3.96034E+01
-1.1672E+00  9.6021E-01    1.5114E+00  140.56    1043      1      3.96034E+01
-1.1672E+00  9.6017E-01    1.5114E+00  140.56    1037      1      3.96034E+01
-1.1672E+00  9.6017E-01    1.5114E+00  140.56    1039      1      3.96034E+01
-1.1672E+00  9.6014E-01    1.5113E+00  140.56    1044      1      3.96034E+01
-1.1671E+00  9.6009E-01    1.5113E+00  140.56    1038      1      3.96034E+01
-1.1670E+00  9.6004E-01    1.5112E+00  140.56    1042      1      3.96034E+01
-1.1670E+00  9.6004E-01    1.5112E+00  140.56    1040      1      3.96034E+01
-1.1670E+00  9.5997E-01    1.5111E+00  140.56    1041      1      3.96034E+01
 2.9833E-06 -5.1878E-05    5.1964E-05  273.29    1003      5      9.82372E+01
 2.9833E-06 -5.1878E-05    5.1964E-05  273.29    1001      5      9.82372E+01
    
```

Note that panel "leftside" has participations equaling 1.511 psi of the total 1.766 psi for the fluid grid 1041 response. This panel factor is the sum of the grid factors in the panel. They are shown below for 39 hz. Only structural mode 1 is shown since it only is significant. Also only fluid grid 1041 is shown:

```

MODAL PARTICIPATION FACTORS FOR SUBCASE          1 (DLOAD ID =      10)
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = PANEL :
LEFTSIDE AT GRID 102

      *** MODE PARTICIPATION FACTOR (MPF) ***
      REAL      IMAGINARY      MAGNITUDE      PHASE      FLUID      MODE      STRUCTURE
      POINT     NUMBER     NATURAL FREQ. (HZ)
 1.9775E-01 -1.6267E-01    2.5606E-01  320.56    1041      1      3.96034E+01

MODAL PARTICIPATION FACTORS FOR SUBCASE          1 (DLOAD ID =      10)
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = PANEL :
LEFTSIDE AT GRID 104

      *** MODE PARTICIPATION FACTOR (MPF) ***
      REAL      IMAGINARY      MAGNITUDE      PHASE      FLUID      MODE      STRUCTURE
      POINT     NUMBER     NATURAL FREQ. (HZ)
 1.9775E-01 -1.6267E-01    2.5606E-01  320.56    1041      1      3.96034E+01

MODAL PARTICIPATION FACTORS FOR SUBCASE          1 (DLOAD ID =      10)
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = PANEL :
LEFTSIDE AT GRID 105

      *** MODE PARTICIPATION FACTOR (MPF) ***
      REAL      IMAGINARY      MAGNITUDE      PHASE      FLUID      MODE      STRUCTURE
      POINT     NUMBER     NATURAL FREQ. (HZ)
 5.7371E-01 -4.7195E-01    7.4288E-01  320.56    1041      1      3.96034E+01

MODAL PARTICIPATION FACTORS FOR SUBCASE          1 (DLOAD ID =      10)
DESCENDING ORDER SORT ON MAGN VALUE AT EXCITATION FREQUENCY OF 3.90000E+01 HZ FOR MPF TYPE = PANEL :
LEFTSIDE AT GRID 106

      *** MODE PARTICIPATION FACTOR (MPF) ***
      REAL      IMAGINARY      MAGNITUDE      PHASE      FLUID      MODE      STRUCTURE
      POINT     NUMBER     NATURAL FREQ. (HZ)
 1.9775E-01 -1.6267E-01    2.5606E-01  320.56    1041      1      3.96034E+01
    
```

When the real and imaginary terms are summed for each panel grid factor, and the magnitude calculated, it will equal the panel factor of 1.511 psi.

13.5.5 Hydroelastic Modeling

See the *MSC.Nastran Advanced Dynamic Analysis User's Guide*.

13.5.6 The Virtual Mass Method

See the *MSC.Nastran Advanced Dynamic Analysis User's Guide*.

13.6 Acoustic Cavity Modeling

The Bulk Data entries used specifically for acoustic cavity analysis are described below, and the entry formats are exhibited in the [MSC.Nastran Quick Reference Guide](#). Their purposes are analogous to the use of structural data entries. A mesh is defined over the longitudinal cross section of an acoustic cavity, and finite elements are connected between these points to define the enclosed volume.

The points are defined by GRIDF entries for the axisymmetric central fluid cavity and by GRIDS for the radial slots. The GRIDF points are interconnected by finite elements via the CAXIF2, CAXIF3, and CAXIF4 entries to define a cross-sectional area of the body of rotation. The CAXIF2 element defines the area of the cross section between the axis and two points off the axis (the GRIDF points may not have a zero radius). The CAXIF3 and CAXIF4 entries define triangular or quadrilateral cross sections and connect three or four GRIDF points, respectively. The density and/or bulk modulus at each location of the enclosed fluid may also be defined on these entries.

The GRIDS points in the slot region are interconnected by finite elements via the CSLOT3 and CSLOT4 entries. These define finite elements with triangular and quadrilateral cross-sectional shapes, respectively. The width of the slot and the number of slots may be defined by default values on the AXSLOT entry. If the width of the slots is a variable, the value is specified on the GRIDS entries at each point. The number of slots, the density, and/or the bulk modulus of the fluid may also be defined individually for each element on the CSLOT3 and CSLOT4 entries.

The AXSLOT Bulk Data entry is used to define the overall parameters for the system. Some of these parameters are called the “default” values and may be selectively changed at particular cross sections of the structure. The values specified on the AXSLOT entry will be used if the corresponding fields on the GRIDS, CAXIF_i, or CSLOT_i entries are left blank. The fields RHOD and RHO (density) and BD and B (bulk modulus) are properties of the fluid. If the value given for bulk modulus is zero, the fluid is considered incompressible by the program. The fields MD and M (number of slots) and WD (slot width) are properties of the geometry. The fields MD and M define the number of equally spaced slots around the circumference, with the first slot located at $\phi = 0^\circ$. The field N (harmonic number) is selected by the user to analyze a particular set of acoustic modes. The pressure is assumed to have the following distribution:

$$p(r, z, \phi) = (r, z) \cos N\phi$$

If $N = 0$, the breathing and longitudinal modes will result. If $N = 1$, the pressure at $\phi = 180^\circ$ will be the negative of the pressure at $\phi = 0^\circ$. If $N = 2$, the pressures at $\phi = 90^\circ$ and $\phi = 270^\circ$ will be the negative of that at $\phi = 0^\circ$. Values of N larger than $M/2$ have no significance.

The interface between the central cavity and the slots is defined with the SLBDY Bulk Data entries. The data for each entry consist of the density of the fluid at the interface, the number of radial slots around the circumference, and a list of GRIDS points that are listed in the sequence in which they occur as the boundary is traversed. In order to ensure continuity between GRIDF and GRIDS points at the interface, the GRIDF points on the boundary between the cylindrical cavity and the slots are identified on the corresponding GRIDS entries rather than on GRIDF entries. Thus, the locations of the GRIDF points will be exactly the same as the locations of the corresponding GRIDS points.

Various standard bulk data entries may be used for special purposes in acoustic analysis. The SPCi Bulk Data entry may be used to constrain the pressures to zero at specified points, such as at a free boundary. The formats for these entries are included in the *MSC.Nastran Quick Reference Guide*. Dynamic load entries, direct input matrices, and scalar elements may be introduced to account for special effects.

Assumptions and Limitations

The accuracy of the acoustic model will be dependent on the selection of the mesh of finite elements. The assumption for each element is that the pressure field has a linear variation over the cross section and a sinusoidal variation around the axis in the circumferential direction. In areas where the pressure gradient changes are large, such as near a sharp corner, the points in the mesh should be placed closer together so that large changes in flow may be defined accurately by the finite elements.

The shape of the finite elements play an important part in the accuracy of the results. It has been observed that long, narrow elements produce disproportionate errors. Cutting a large square into two rectangles will not improve the results, whereas dividing the square into four smaller squares may decrease the local error by as much as a factor of ten.

The slot portion of the cavity is limited to certain shapes because of basic assumptions in the algorithms. The cross section of the cavity normal to the axis must have a shape that is reasonably well defined by a central circular cavity having equally spaced, narrow slots. Various shapes are shown in [Figure 13-48](#) in the order of increasing expected error.

It is recommended that shapes such as the cloverleaf and square cross section be analyzed with a full three-dimensional technique. The assumption of negligible pressure gradient in the circumferential direction within a slot is not valid in these cases.

The harmonic orders of the solutions are also limited by the width of the slots. The harmonic number, N , should be no greater than the number of slots divided by two. The response of the higher harmonics is approximated by the slot width correction terms discussed in Section 17.1 of *The NASTRAN Theoretical Manual*.

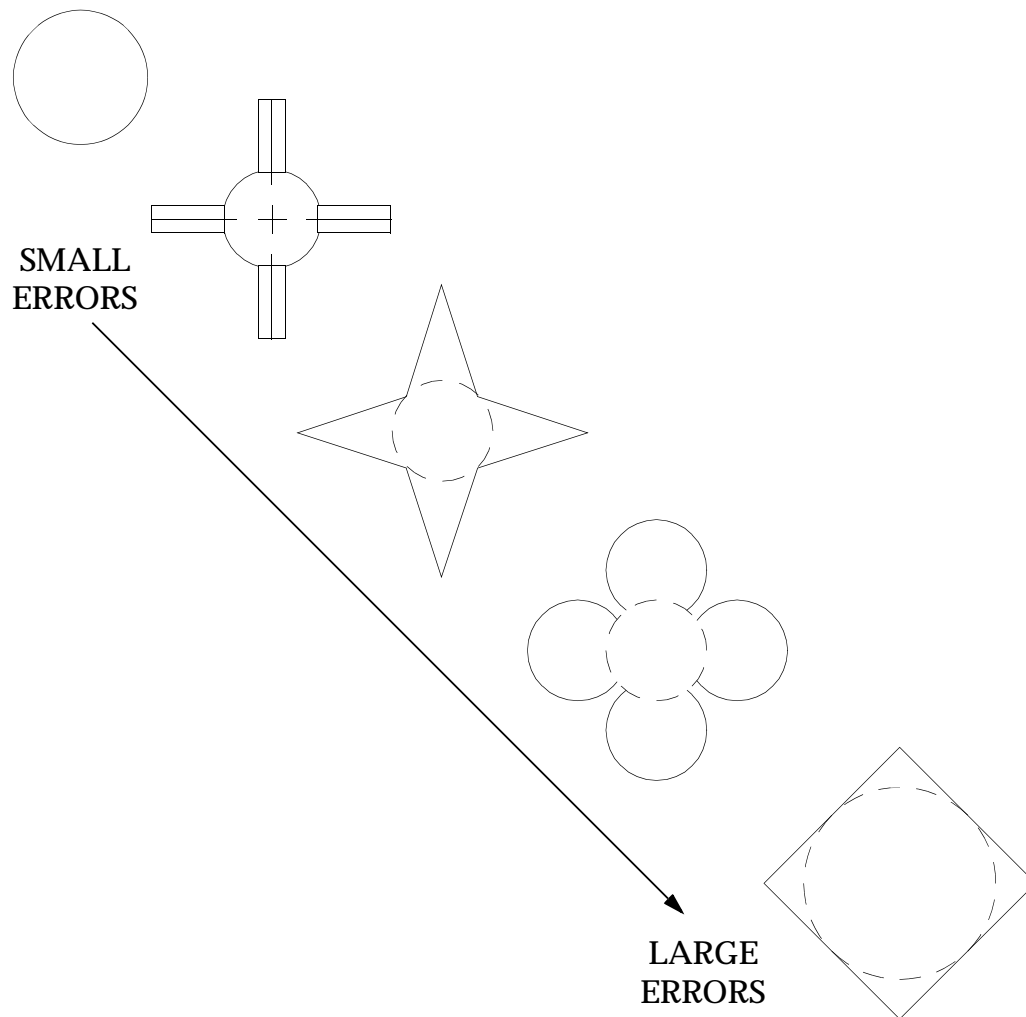


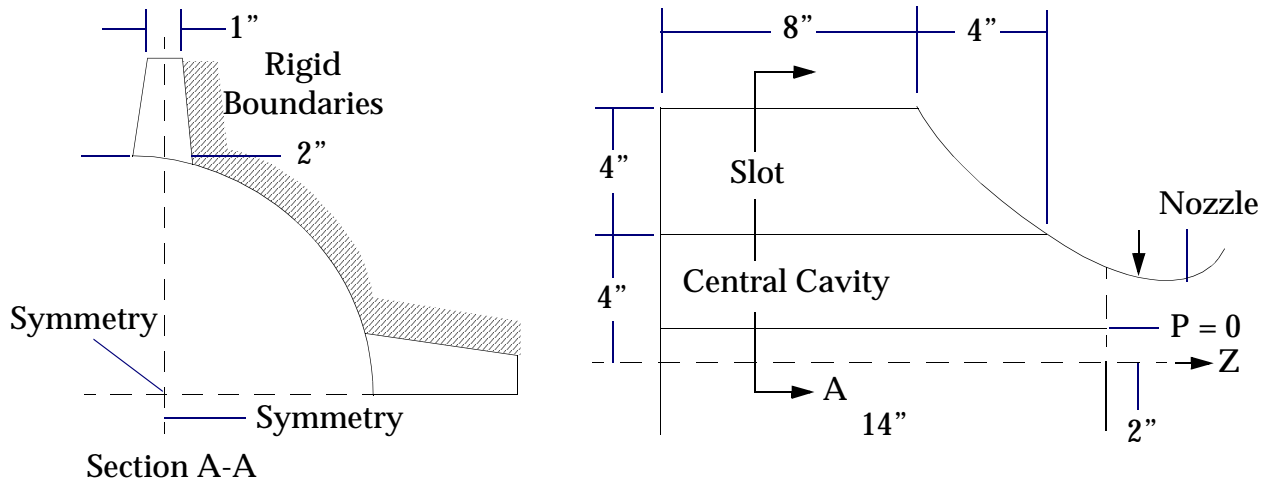
Figure 13-48 Modeling Errors for Various Shapes

The output data for the acoustic analysis consist of the values of pressure in the displacement vector selected via the Case Control command `PRESSURE = n`. The velocity vector components corresponding to each mode may be optionally requested by the Case Control command `STRESS = n` where n is the set number indicating the element numbers to be used for output, or by the words `STRESS = ALL`.

Plots of the finite element model and/or of the pressure field may be requested with the MSC.Nastran plot commands. The central cavity cross section will be positioned in the XY plane of the basic coordinate system. The slot elements are offset from the XY plane by the width of the slot in the +Z direction. The radial direction corresponds to X and the axial direction corresponds to the Y direction. Pressures will be plotted in the Z direction for both the slot points and the central cavity points. The PLOTTEL elements are used for plotting the acoustic cavity shape. The plot request entry SET n INCLUDE PLOTTEL must be used where n is a set number.

Acoustic Cavity Example Problem

Table 13-19 contains a listing of the data entries used as a simple example of acoustic cavity analysis. The problem to be solved is illustrated in **Figure 13-49**. The model was subdivided into only ten finite elements in order to limit the number of data entries. For reasonable engineering accuracy, this model should be subdivided into at least four times that number of elements.



<u>Parameters:</u>	Density:	RHOD	=	$1.2 \times 10^{-7} \text{ lb-sec}^2/\text{in}^4$
	Bulk Modulus:	BD	=	$ra^2 = gRT = 21.0 \text{ lb/in}^2$
	Harmonic:	N	=	1
	Number of Slots:	MD	=	4

Finite Element Model:

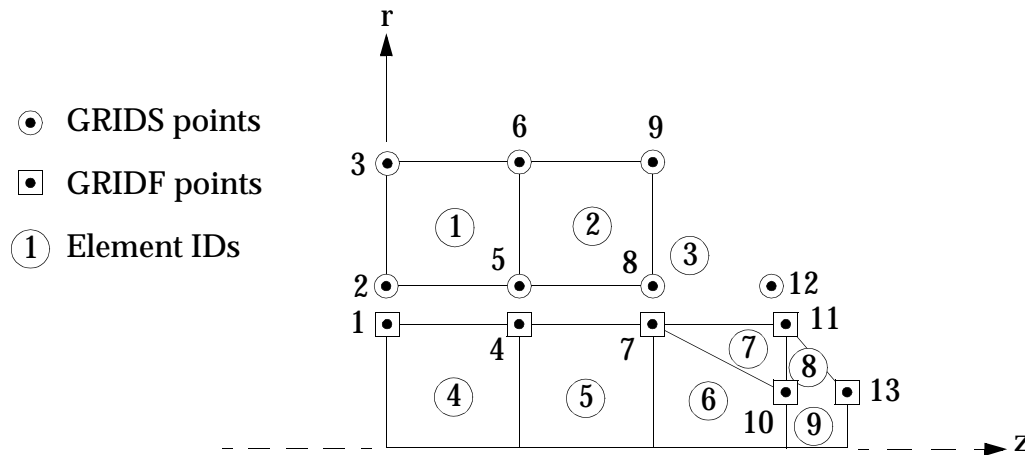


Figure 13-49 Description of Acoustic Cavity Example Problem

Each data entry in [Table 13-19](#) is given a number on the left side. The format for each type of Bulk Data entry is given in parentheses above the group of that type. The following is a brief description of each part of the input data:

Lines

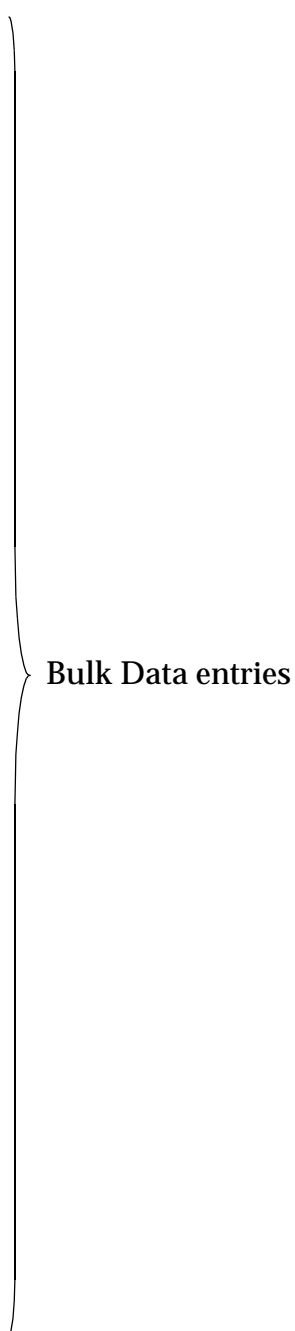
- 1-4 Each statement in the Executive Control Section has the format of a request word and a selection separated by blanks or a comma. The CEND statement is last, but the statements above may appear in any order. Solution Sequence 103, normal modes analysis, was chosen. A limit of 2 minutes CPU time was set (TIME2).
- 5-7 The TITLE, SUBTITLE, and LABEL commands may contain any list of letters and numbers following the (=) sign. This list will appear on the first three lines of each output page.
- 8 The method of eigenvalue extraction is selected with the METHOD command. The number 11 refers to the identification number of an EIGRL Bulk Data entry which appears below as line 32.
- 9-11 A simple output request is illustrated with these commands. PRES = ALL will result in the printout of all pressures at the GRIDF and GRIDS points. STRESS = ALL will result in the printout of all velocities in the elements. This printout will occur for all extracted eigenvectors. Selected points or elements can be printed via the SET command.
- 12 The BEGIN BULK command denotes the beginning of the Bulk Data Section. The Bulk Data entries may occur in any order. Putting these entries in alphabetic sort will save sorting time in large problems, however.
- 13 In this problem, all the parameters except slot width WD are constant throughout the volume. The values on AXSLOT will be used whenever a corresponding field in the following entries is blank.
- 14-20 The location of points on the slot are defined with these entries. Entries 14, 16, 18, and 20 serve a dual purpose by defining a GRIDS point identification number in field 2 and a GRIDF point identification number in field 6. The two types of points thereby are forced to have the same locations at the interface.
- 21-22 The location of points within the axisymmetric fluid cavity are described by the GRIDF entry. No points are allowed to have a zero or negative radius.
- 23-31 These entries describe the elements shown in [Figure 13-49](#). Each element is given a unique identification number and a list of the connected GRIDS or GRIDF points. Since the parameters p and B are constants, these fields are left blank, so the values on the AXSLOT entry will be used.

- 32 The ElGRL entry is used to define parameters for eigenvalue extraction (resonant frequencies). More than one of these entries may appear. The method to be used is selected with the METHOD command (command 8). With this particular command, the Lanczos method was selected with three (ND = 3) output mode shapes specified.
- 33 The SLBDY entry defines the boundary between the slot and the central cavity. Both the density (RHO) and the number of radial slots (M) are blank, so the AXSLOT defaults are used; i.e., RHOD = 1.2×10^{-7} and M = 4. Only four GRIDS points are on the boundary, so a continuation entry is not necessary. Field 8 is left blank to signify the last entry.
- 34 The ENDDATA entry is required to denote the end of the Bulk Data Section. Any following entries will be ignored.

Table 13-19 Acoustic Cavity Example Problem Data

Line No.		
1	\$DEMO PROBLEM D03D92 WITH SOL 103 AND LANCZOS	}
2	SOL 103	
3	TIME 2	
4	CEND	
5	TITLE = ACOUSTIC CAVITY EXAMPLE PROBLEM	}
6	SUBTITLE = FIRST HARMONIC	
7	LABEL = NORMAL MODES (LANCZOS)	
8	METHOD = 11	
9	OUTPUT	
10	PRES = ALL	
11	STRESS = ALL	
12	BEGIN BULK	

Table 13-19 Acoustic Cavity Example Problem Data

Line No.		
13	(AXSLOT, (RHOD, BD, N, WD, MD) AXSLOT,1.2-7,21.0,1,,4 (GRIDS, ID,R,Z,W, ID	
14	GRIDS, 2,4.0,0.0,0.23 01,1	
15	GRIDS, 3,8.0, .0,1.0	
16	GRIDS, 5,4.0,4.0,2.0,4	
17	GRIDS, 6,8.0,4.0,1.0	
18	GRIDS, 8,4.0,8.0,2.0,7	
19	GRIDS, 9,8.0,8.0,1.0	
20	GRIDS, 12,4.0,1.2+1,2.0,11 (GRIDF, ID,R,Z)	
21	GRIDF, 10,2.0,12.0	
22	GRIDF, 13,2.0,1.4E1 (CSLOT4, 1D,P ₁ ,P ₂ ,P ₃ ,P ₄ ,RHO,,M)	
23	CSLOT4, 1 ,2,3,6,5	
24	CSLOT4, 2,5,6,9,8 (CSLOT3, 1D,P ₁ ,P ₂ ,P ₃ ,,RHO,,)	
25	CSLOT3, 3,8,9,12 (CAXIF2, 1D,P ₁ ,P ₂ ,,,RHO)	
26	CAXIF2, 4,1 ,4	
27	CAXIF2, 5,4,7	
28	CAXIF2, 6,7,10	
29	CAXIF2, 9,10,13 (CAXIF3, ID,P ₁ ,P ₂ ,P ₃ ,,RHO,B)	
30	CAXIF3, 7,7,10,11	
31	CAXIF3, 8,10,11,13 (EIGRL, SID,V1,V2,ND)	
32	EIGRL, 11,,,3 (SLBDY, RHO,M, ID1, ID2, ID4,etc.)	
33	SLBDY, ,,12,8,5,2	
34	ENDDATA	

13.7 Cyclic Symmetry

Introduction

Symmetry has long been a subject on which philosophers and mathematicians have focused their attention. Their inquiries into the concepts of symmetry have led to powerful results that have, in turn, led to the formulation of analysis procedures of general applicability in applied mechanics. A familiar example is provided by the routine application of reflective or bilateral symmetry in many engineering disciplines. The principles of reflective symmetry allow the analyst to reduce the size of his analytical model by one-half for each plane of reflective symmetry in the system under investigation.

Reflective symmetry is only one of the types of symmetry that exist in nature. Among the other types is cyclic or rotational symmetry in which the properties of a body are repeated at equal intervals about an axis of symmetry. Mathematicians have developed a general theory of cyclic groups to deal with such problems, which electrical engineers have applied, in the method of symmetrical components originated by Fortescue in 1918, to the analysis of polyphase electrical networks.

An original method for the application of the concepts of cyclic symmetry to structural analysis is available in MSC.Nastran. The method introduces no sacrifice in theoretical rigor or generality and provides substantial reductions in computer time for the analysis of linear systems that possess rotational and/or reflective planes of symmetry. These techniques are currently available in the following solution sequences:

SOLution 14	Statics
SOLution 15	Modes
SOLution 16	Buckling
SOLution 114	Superelement Static Analysis
SOLution 115	Superelement Normal Modes Analysis
SOLution 116	Cyclic Buckling
SOLution 118	Superelement Direct Frequency Response

In addition to significant savings in computer time, the analyst who grasps the concept of cyclic symmetry can realize a considerable reduction in problem preparation tasks. For example, if a body consists of N identical segments in the sense that the entire body can be generated through reflections and/or rotations of one (the fundamental region) of these N segments, the analyst is only required to define a finite

element model for the fundamental region and to provide a list of the boundary grid points for this fundamental region. The finite element model may include any members of the MSC.Nastran element library other than axisymmetric elements. All boundary conditions are automatically handled internally by MSC.Nastran.

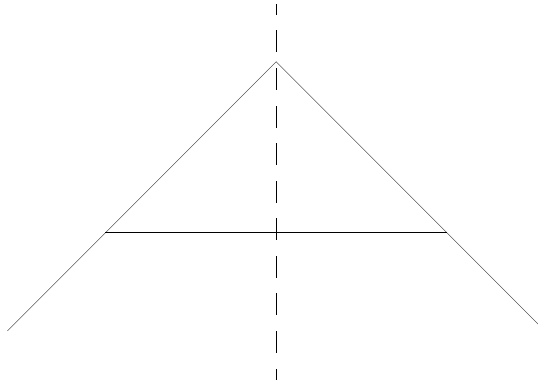
Complete generality in the definition of loading conditions is achieved by allowing the user to independently specify loads, enforced motions, and/or temperatures on each of the N segments, even though the fundamental region is the only segment explicitly represented in the analysis. Output requests are handled with the same degree of generality in that solutions are provided for each of the N segments.

13.7.1 Symmetry

Prior to specific consideration of the cyclic symmetry capability in MSC.Nastran, it is useful to briefly review the aspects of symmetry that concerns the analyst. Specifically, the concern is with those symmetry operations that involve the permutations of a fundamental region of a body that leave the body unchanged. Several examples are presented to illustrate the points of interest.

A-Frame (Dihedral Symmetry)

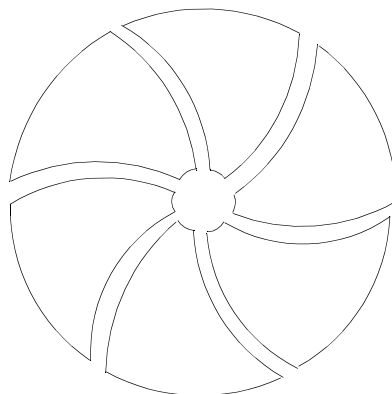
An A-frame has a plane of reflective symmetry as indicated by the dashed line in the figure below.



A reflection of the left half of the symmetric frame through the plane of symmetry results in the right half of the frame. This familiar reflective type of symmetry is known as dihedral (DIH) symmetry. Either symmetric half of the frame may be taken as the fundamental region.

Pinwheel (Rotational Symmetry)

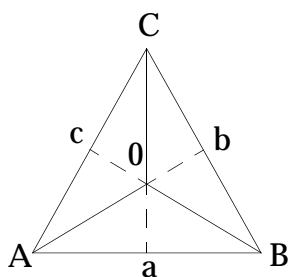
An example of a rotationally symmetric body is provided by the following example in which the “leaves” of a pinwheel have curved boundaries.



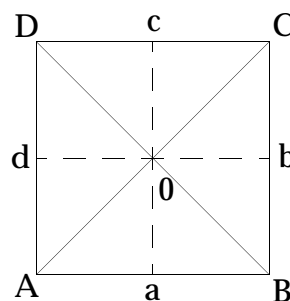
In this case, there are no planes of dihedral symmetry, but it is observed that each of the six “leaves” of the figure are identical and that rotations of the fundamental region, e.g., segment 1, through the effectively distinct multiples of $360^\circ / 6$, will “generate” the complete pinwheel. This type of symmetry is known as rotational symmetry. In general, rotational symmetry includes N symmetry operations corresponding to the effectively distinct multiples $360^\circ / N$ where N is the number of identical segments in the body.

N-Gons

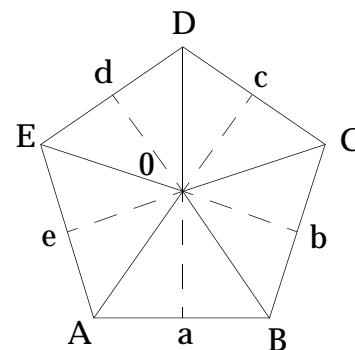
In more general cases, a body may be susceptible to both DIH and ROT symmetry operations. For example, consider the three regular N -gons: the equilateral triangle, the square, and the pentagon.



$N = 3$



$N = 4$



$N = 5$

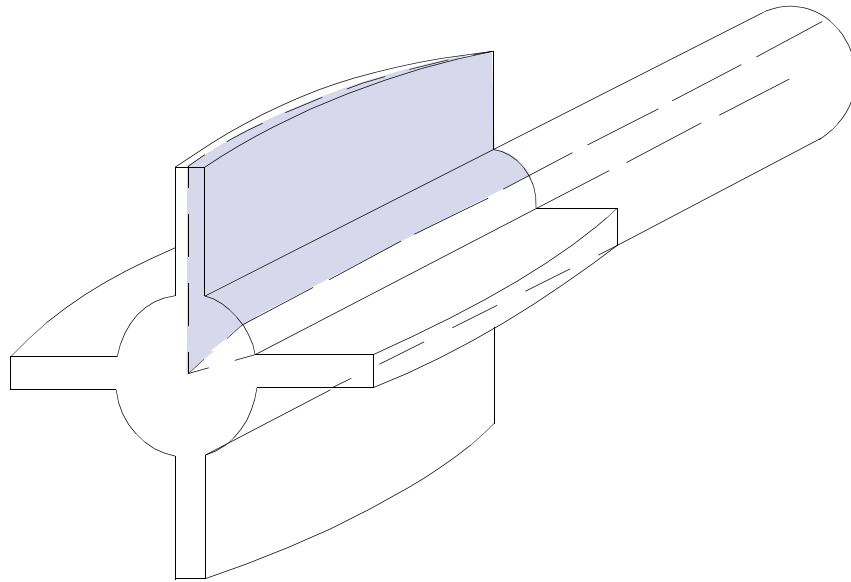
In these figures, the internal solid lines in conjunction with the edges define fundamental regions, e.g., $A0B$, that may be permuted through the ROT symmetry operations. The axis of rotation is normal to the plane of the page and passes through the point 0 .

In addition, the dotted internal lines such as a_0 may be used to, for example, reflect A_0a into a_0B . Also, a_0B may be reflected through B_0 to obtain B_0b , and so on around each N -gon until the identity of A_0a is reached. Clearly, the fundamental region A_0a for the DIH symmetry operations is smaller than the fundamental region A_0B for ROT symmetry operations, and both of these fundamental regions are substantially smaller than the fundamental region required by traditional reflective symmetry techniques.

Three-Dimensional Bodies

The reader has undoubtedly noted that the previous examples involved planar bodies. However, the previous discussions would not in any way be altered if these examples were assumed to be three-dimensional bodies, i.e., if the illustrations represent a cross-section of a uniform body perpendicular to the plane of the page. In fact, there are no restrictions with regard to changes in cross-sectional geometry as a function of distance along the axis of rotational symmetry.

In the example, the fundamental region is indicated by the shaded volume.



The symmetry operations are of the type DIH.

Cyclic Symmetry Versus Traditional Reflective Symmetry Techniques

Engineers are generally familiar with the traditional techniques of reflective symmetry that could be employed in the analysis of any of the previous examples other than the pinwheel. The following list compares the major steps in the analysis procedures required with traditional reflective techniques and with cyclic symmetry techniques. The advantages of the cyclic symmetry techniques are most vivid.

1. Analytical Model

a. Traditional Techniques

The size of the analytical model may be reduced by one-half for each plane of reflective symmetry.

b. Cyclic Symmetry

The size of the analytical model is at most equal to the size of the model required by traditional analysis techniques (A-frame). In general, the analytical model required for cyclic symmetry analysis techniques is substantially smaller.

2. Boundary Conditions

a. Traditional Techniques

In the general case of p planes of reflective symmetry, the analyst must explicitly define 2^p sets of boundary conditions. These boundary conditions are defined through SPCi Bulk Data entries.

b. Cyclic Symmetry

The user supplies a list of the grid points on the boundary of the fundamental region. Boundary conditions are automatically imposed by the program.

3. Applied Loads

a. Traditional Techniques

The analyst must decompose the various loading conditions into symmetric and antisymmetric components.

b. Cyclic Symmetry

Applied loads are defined to act at arbitrary locations throughout the entire structure respective of whether any or all of these locations are in the explicitly defined fundamental region.

4. Computer Solution

a. Traditional Techniques

For each set of defined boundary conditions, a lengthy matrix decomposition and equation solution are required. Static condensations are generally not economically feasible in static analysis.

b. Cyclic Symmetry

A static condensation is performed once to reduce the size of the analysis set. In static analysis, all but the boundary degrees-of-freedom should be omitted. In real eigenvalue analysis, buckling analysis, and dynamic analysis, the boundary degrees-of-freedom and selected internal points are kept in the analysis set. The static condensation involves a lengthy decomposition which, however, should be less lengthy than any one of the decompositions required by traditional techniques because the fundamental region for cyclic symmetry is generally substantially smaller than the region required for other analysis techniques. Several additional decompositions of relatively trivial magnitude are required after the static condensation.

5. Data Recovery

a. Traditional Techniques

The results for each boundary condition must be combined to obtain the complete response of the structure. The user must define such combinations through SUBCOM or SYMCOM Case Control commands.

b. Cyclic Symmetry

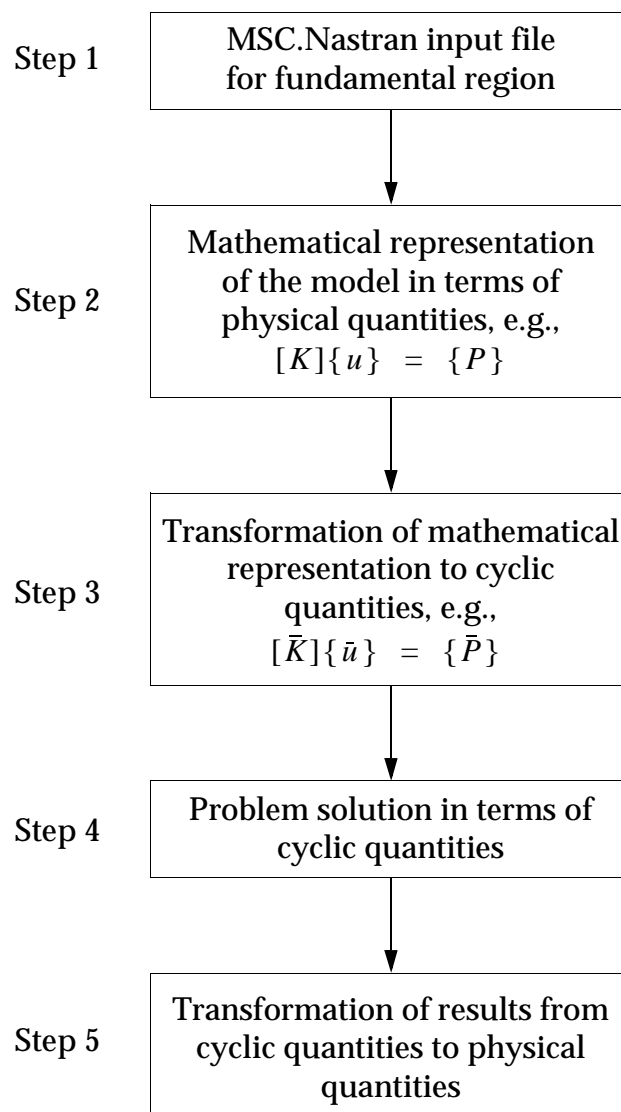
The response for the entire structure is automatically provided without any special user action. SUBCOMs and SYMCOMs are not necessary and must not be used.

In summary, cyclic symmetry provides a uniform treatment of symmetry with a potential for substantial savings in data preparation and solution cost.

Summary

The examples that have been discussed provide heuristic evidence that cyclic symmetry may be of significant value in structural analysis. One advantage is that the analyst will be required to provide an analytical model for only the fundamental regional of a structure, and this fundamental region is generally substantially smaller than the analytical model required by more traditional analysis techniques. In addition, the comparison between cyclic symmetry and more traditional analysis procedures provided in the previous section indicates that both user convenience and reductions in computer time requirements are uniquely available through cyclic symmetry procedures.

The validity of these statements relies upon the development of transformation relations that allow the analyst to manipulate the mathematical representation of the fundamental region in accordance with the appropriate symmetry operations. In graphical terms, the following operations are performed.



Steps 3 and 5 are of particular interest because they depend on the existence of the necessary transformations. Step 4 is also of interest in that it must be demonstrated that problem solution procedures in terms of cyclic quantities are valid and require less computer time than more traditional solution procedures. These matters are addressed in the next section.

13.7.2 Theory

The theory of cyclic symmetry is presented in some detail for the ROT case, and the corresponding results for the DIH case are summarized. This form of presentation will provide the interested user with sufficient information to independently investigate many of the theoretical aspects of cyclic symmetry that cannot be adequately covered within the limits of this discussion.

Symmetrical Components

Consider an arrangement of N identical objects symmetrically disposed about a normal. The angle between these objects is

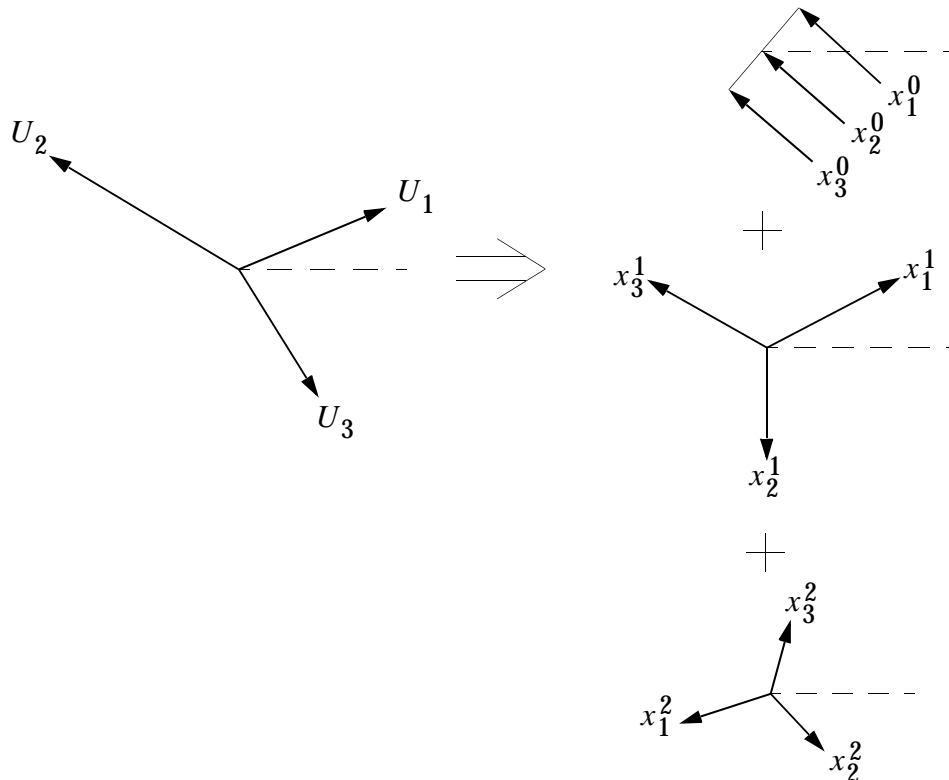
$$a = \frac{2\pi}{N} \quad \text{Eq. 13-77}$$

and, in addition, let $U_1(r, t)$ be any variable displacement, force, stress, temperature, etc. in object 1, the fundamental region. The variables r and t denote distance and time respectively. The corresponding quantity in the n -th object is $U_n(r, t)$ and as a single valued solution is required

$$U_n = U_{n+N} \quad n = 1, 2, \dots, N \quad \text{Eq. 13-78}$$

The various U_n 's are, in general, unsymmetric quantities that are not related in a simple manner.

The work by Fortescue on symmetrical component theory was referenced in the introduction of this section. This theory states that a general vector field on N -vectors with arbitrary phase relationships may be resolved into N -sets of N -symmetrically phased vectors of equal magnitude. For $N = 3$, the vectors may be illustrated by the following diagram.



Now, in accordance with the theory of symmetrical components, the following vector sums may be stated.

$$\begin{aligned} U_1 &= x_1^0 + x_1^1 + x_1^2 \\ U_2 &= x_2^0 + x_2^1 + x_2^2 \\ U_3 &= x_3^0 + x_3^1 + x_3^2 \end{aligned} \quad \text{Eq. 13-79}$$

Here, the U_n 's represent physical quantities and the x_n^k terms represent symmetrical components. The superscripts k denote the order of the symmetric components. The subscripts n are taken to be the number of a particular vector in the vector field. In the general context of cyclic symmetry, the subscripts n denote the number applied to each of the N identical objects under consideration. The object $n = 1$ is arbitrarily taken as the fundamental region.

Transformation from Physical Components to Cyclic Components (ROT)

These transformations for the symmetric components in terms of the general field vectors may be written in summation form for the general case of N vectors.

$$x_1^k = \frac{1}{N} \sum_{n=1}^N U_n e^{i(n-1)ka} \quad \text{Eq. 13-80}$$

where a is given by [Eq. 13-77](#).

Transformation from Cyclic Components to Physical Components (ROT)

After the solution phase of the problem, it will normally be necessary to transform the results from cyclic components back into physical components.

[Eq. 13-79](#) may be written in the following form.

$$\begin{aligned}
U_n &= \sum_{k=0}^{N-1} x_1^k e^{-1(n-1)ka} \\
&= x_1^0 + [x_1^1 e^{-1(n-1)a} + x_1^{(N-1)} e^{i(n-1)a}] \\
&\quad + [x_1^2 e^{-1(n-1)a} + x_1^{(N-2)} e^{i(n-1)2a}] \\
&\quad + x_1^3 e^{-1(n-1)a} + x_1^{(N-3)} e^{i(n-1)3a} \\
&\quad + \dots, \\
&\quad + x_1^{N/2} (-1)^{(n-1)}
\end{aligned} \tag{Eq. 13-81}$$

Here,

$$e^{-1(n-1)(N-k)a} = e^{-1(n-1)ka} e^{-1(n-1)Na} = e^{-1(n-1)ka}$$

and

$$e^{-1(n-1)(N/2)a} = (-1)^{n-1}$$

If the exponentials in the bracketed terms in the above summation are replaced by the relations $e^{\pm ix} = \cos x \pm i \sin x$, the typical bracketed term takes the following form:

$$[(x_1^k + x_1^{(N-k)}) \cos(n-1)ka - i(x_1^k - x_1^{(N-k)}) \sin(n-1)ka]$$

The purpose for the somewhat arbitrary introduction of **Eq. 13-81** is now recognized, and the typical bracketed term assumes the following form:

$$[U^{kc} \cos(n-1)ka + U^{ks} \sin(n-1)ka]$$

The final form of the transformation from cyclic components to physical components is therefore,

$$\begin{aligned}
U_n &= x_1^0 + \sum_{k=1}^{K_L} [U^{kc} \cos(n-1)ka + U^{ks} \sin(n-1)ka] + (-1)^{n-1} X_1^{N/2} \\
&\quad n = 1, 2, 3, \dots, N
\end{aligned} \tag{Eq. 13-82}$$

In **Eq. 13-82**, U_n represents physical quantities such as displacement force, stress, temperature in the n -th segment in terms of the cyclic components in the fundamental region. The following points should be noted:

1. $K_L = \frac{(N-1)}{2}$ and $x_1^{N/2}$ is nonexistent when N is odd.
2. $K_L = \frac{(N-2)}{2}$ when N is even.
3. The notation for cyclic components of order $k = 0$ and $k = N/2$ is distinct from the notation for cyclic components of order $1 \leq k \leq k_L$ because of the somewhat peculiar nature of these two components.

Boundary Conditions and Problem Solution (ROT)

The transformation equations of “**Theory**” on page 809 are assumed to allow one to transform a problem that is mathematically defined in terms of physical components into a set of mathematical statements written in terms of cyclic components.

Specifically, the mathematical description of the n replications of the fundamental region may be written in the form

$$\{F_n\} = [Y_n]\{U_n\} \quad \text{Eq. 13-83}$$

This expression represents the mathematical description of the system in terms of physical coordinates.

The application of previously defined transform equations may now be introduced to obtain the desired mathematical description of the system in terms of cyclic components. First, however, consider the matrix representation of the transform **Eq. 13-80** and **Eq. 13-82**, respectively. **Eq. 13-80** assumes the form

$$\begin{Bmatrix} x_1^0 \\ U^{1c} \\ U^{1s} \\ U^{2c} \\ \vdots \\ U^{K_L s} \\ x_1^{N/2} \end{Bmatrix} = \begin{bmatrix} \frac{1}{N} & \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \frac{2}{N} & \frac{2}{N} \cos a & \frac{2}{N} \cos 2a & \cdots & \frac{2}{N} \cos(N-1)a \\ 0 & \frac{2}{N} \sin a & \frac{2}{N} \sin 2a & \cdots & \frac{2}{N} \sin(N-1)a \\ \frac{2}{N} & \frac{2}{N} \cos 2a & \frac{2}{N} \cos 4a & \cdots & \frac{2}{N} \cos(N-1)2a \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \frac{2}{N} \sin K_L a & \frac{2}{N} \sin 2K_L a & \cdots & \frac{2}{N} \sin(N-1)K_L a \\ \frac{1}{N} & \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} (-1)^{N-1} \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ \vdots \\ \vdots \\ U_N \end{Bmatrix}$$

$$\{U^k\} = [T_{xU}]\{U_n\}$$

$$\text{Eq. 13-84}$$

Eq. 13-82 assumes the form

$$\begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ \vdots \\ U_N \end{Bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & \dots & 0 & 1 \\ 1 & \cos a & \sin a & \cos 2a & \dots & \sin K_L a & -1 \\ 1 & \cos 2a & \sin 2a & \cos 4a & \dots & \sin 2K_L a & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cos(N-1)a & \sin(N-1)a & \cos(N-1)2a & \dots & \sin(N-1)K_L a & (-1)^{(N-1)} \end{bmatrix} \begin{Bmatrix} x_0^1 \\ U^{1c} \\ U^{1s} \\ \vdots \\ U^{K_L S} \\ x_1^{N/2} \end{Bmatrix}$$

$$\{U_n\} = [T_{Ux}]\{U^k\} \quad \text{Eq. 13-85}$$

It is easily demonstrated that

$$[T_{xU}][T_{Ux}] = [I]$$

Return now to Eq. 13-83, the mathematical representation of the system in terms of physical coordinates, and represent the energy of the system in the form

$$U = [U_n]^T \{F_n\} \quad \text{Eq. 13-86}$$

From Eq. 13-83, Eq. 13-85, and Eq. 13-86

$$U = \begin{cases} \{U_n\}^T \{Y_n\} \{U_n\} \\ \{U^k\}^T [T_{Ux}]^T [Y_n] [T_{Ux}] \{U^k\} \end{cases} \quad \text{Eq. 13-87}$$

The system energy in terms of cyclic components may be written as

$$U = \{U^k\}^T [Y_k] \{U^k\} \quad \text{Eq. 13-88}$$

Since the energy must be invariant to the transformations, Eq. 13-88 and Eq. 13-89 may be equated

$$[Y_k] = [T_{Ux}]^T [Y_n] [T_{Ux}] \quad \text{Eq. 13-89}$$

That is, the equations of the system in terms of physical components are transformed into a set of equations in terms of cyclic components. As there is no coupling between the N identical objects of the system, the matrix Y_n may be written so that it consists of N identical blocks along the diagonal. These blocks represent, in general, the equations of motion of each of the N identical objects of the system.

As an example, assume $N = 4$ ($K_L = 1$), and expand Eq. 13-89. Note that $Y_1 = Y_2 = Y_3 = Y_4$.

$$\begin{aligned}
 [Y_k] &= Y_1 \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \cos a & \cos 2a & \cos 3a \\ 0 & \sin a & \sin 2a & \sin 3a \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} \\ \\ I \\ \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & \cos a & \sin a & -1 \\ 1 & \cos 2a & \sin 2a & 1 \\ 1 & \cos 3a & \sin 3a & -1 \end{bmatrix} \\
 &= Y_1 \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}
 \end{aligned}$$

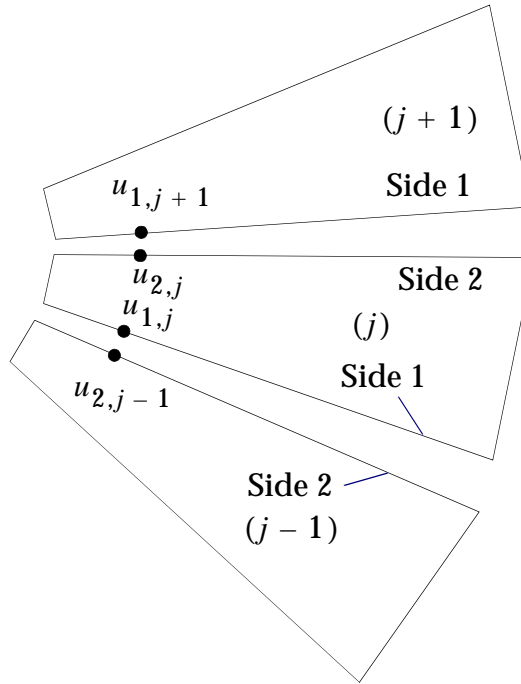
In general, for N objects the above equation may be written

$$\begin{aligned}
 Y_k &= Y_1 N \text{ for } k = 0, N/2 \\
 Y_k &= \frac{N}{2} Y_1 \text{ for } 1 \leq k \leq K_L
 \end{aligned}$$

These equations imply that the equations of motion in terms of cyclic components are either N or $N/2$ times the corresponding equations written in terms of physical components. This argument provides an explanation for the presence of the coefficients that appear in the terms of.

To this point, it has been found that except for boundary conditions, the equations for motion in terms of cyclic components ($k = 0, 1, \dots, N/2$) may be developed from the equations of motion of the fundamental region ($n = 1$). In essence, an arbitrarily large physical system may be investigated in terms of a fundamental region which has been transformed into a number of models defined in terms of cyclic components. The boundary conditions between cyclic models must now be defined.

The only connection between the n identical objects in question exists along the boundaries between adjacent segments. Specifically, consider three of the N objects as indicated in the following figure.



These objects are displaced from one another in order to more clearly define the boundary conditions between the objects. Note the designation of the two sides of each of the elements as side 1 and side 2 as one proceeds in the counterclockwise direction. From the requirement of displacement compatibility, it is necessary that the typical boundary degrees-of-freedom, u , satisfy the equations

$$\begin{aligned} u_{2,j-1} &= u_{1,j} \\ u_{2,j} &= u_{1,j+1} \end{aligned} \quad \text{Eq. 13-90}$$

These physical degrees-of-freedom may be transformed into cyclic components of order K through [Eq. 13-80](#).

$$\begin{aligned} (x^k)_{\text{side 1}} &= \frac{1}{N} u_{1,j} e^{i(j-1)ka} \\ (x^k)_{\text{side 2}} &= \frac{1}{N} u_{1,j+1} e^{ijka} \end{aligned} \quad \text{Eq. 13-91}$$

A comparison of [Eq. 13-91](#) provides the result

$$(x^k)_{\text{side 2}} = (x^k)_{\text{side 1}} e^{ika} \quad \text{Eq. 13-92}$$

for $1 \leq k \leq k_L$. For $k = 0$,

$$(x^k)_{\text{side 2}} = (x^k)_{\text{side 1}} \quad \text{Eq. 13-93}$$

and for $k = \frac{N}{2}$,

$$(x^k)_{\text{side 2}} = -(x^k)_{\text{side 1}} \quad \text{Eq. 13-94}$$

The cyclic component of **Eq. 13-94** may be written in terms of the variables U^{kc} and U^{ks} in accordance with the definitions.

$$\begin{aligned} (U^{kc})_{\text{side 1}} &= [(x^k)_{\text{side 1}} + (x^{N-k})_{\text{side 1}}] \\ (U^{ks})_{\text{side 1}} &= -i[(x^k)_{\text{side 1}} - (x^{N-k})_{\text{side 1}}] \\ (U^{kc})_{\text{side 2}} &= [(x^k)_{\text{side 2}} + (x^{N-k})_{\text{side 2}}] \\ (U^{ks})_{\text{side 2}} &= -i[(x^k)_{\text{side 2}} - (x^{N-k})_{\text{side 2}}] \end{aligned}$$

Eq. 13-92 may now be introduced to yield the equations

$$\begin{aligned} (U^{kc})_{\text{side 2}} &= [(x^k)_{\text{side 1}} e^{ika} + (x^{N-k})_{\text{side 1}} e^{-ika}] \\ (U^{ks})_{\text{side 2}} &= -i[(x^k)_{\text{side 1}} e^{ika} - (x^{N-k})_{\text{side 1}} e^{-ika}] \end{aligned}$$

Upon the replacement of the exponentials with Euler's equation, the appropriate boundary conditions may be represented as follow:

$$\begin{Bmatrix} (U^{kc})_{\text{side 2}} \\ (U^{ks})_{\text{side 2}} \end{Bmatrix} = \begin{bmatrix} \cos ka & -\sin ka \\ \sin ka & \cos ka \end{bmatrix} \begin{Bmatrix} (U^{kc})_{\text{side 1}} \\ (U^{ks})_{\text{side 1}} \end{Bmatrix} \quad \text{Eq. 13-95}$$

These boundary conditions are of the form of constraint equations in which the variables on side 2 are dependent and the variables on side 1 are independent. Indeed, these equations may be viewed as MPC equations that couple kc and ks terms of side 1 to the corresponding terms of side 2.

The equations of motion in terms of cyclic components may be written in the following form.

- The DIH option can be utilized for analyses that would normally be handled with traditional reflective symmetry techniques. The A-frame that was discussed above is an example of this situation.

The form of the transformations between cyclic and physical components and the boundary conditions between the segments of the object in question are, to some extent, dependent on the notation system. The [Figure 13-49](#) defines the notation system used in the DIH option in MSC.Nastran.

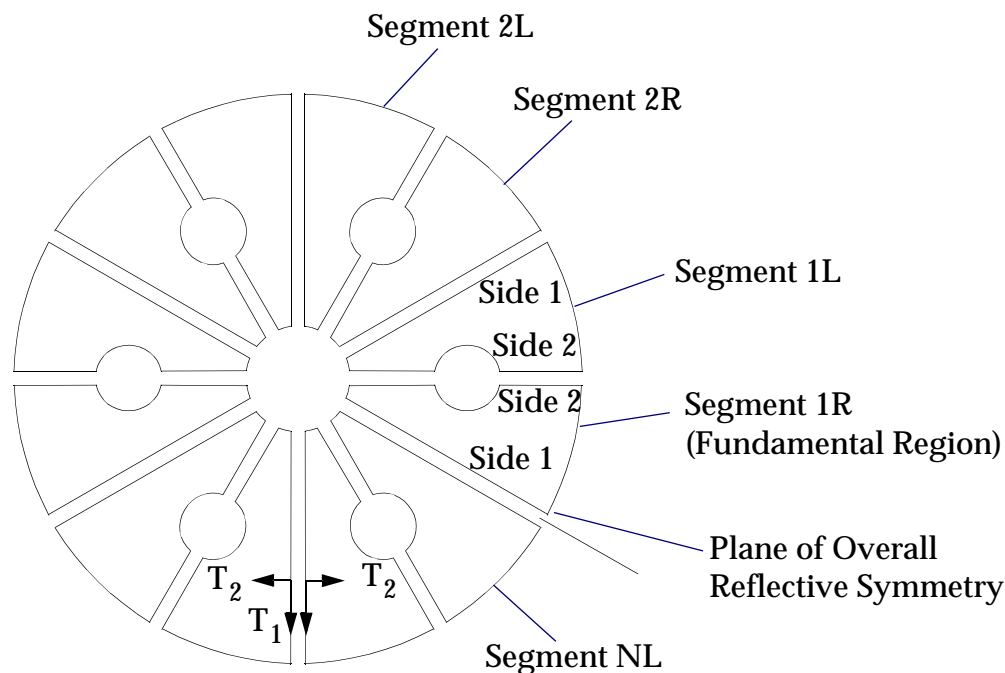


Figure 13-49 Example of Dihedral Option

Note that:

- The midplanes or relative planes are always denoted as side 2. The other boundary, which must also be planar, is always denoted as side 1. In the specific case of a structure with a single overall plane of reflective symmetry (i.e., the complete structure is represented by segments 1R and 1L) the reflective plane may be defined as side 1 or side 2.
- The two halves of a segment that join at a reflective plane are denoted as the right, "R", and left, "L", halves.
- The fundamental region is taken as 1R.

4. R halves use a right-handed coordinate system and L halves use a left-handed coordinate system. Therefore, at each boundary a degree-of-freedom is associated with its mirror image on the adjacent segment so that compatibility always involves a left- and a right-hand coordinate system. The degrees-of-freedom on the boundaries are further categorized into odd and even components in accordance with the following table.

Component	Degrees-of-Freedom
Even	Displacement vectors parallel to the boundary and rotation vectors normal to the boundary.
Odd	Displacement vectors normal to the boundary and rotation vectors parallel to the boundary.

The several equations of interest for the DIH option are indicated below along with the equation number for the corresponding expression under the ROT option. In the following equations, the starred (*) terms donate anti-symmetric motions and unstarred terms denote symmetric motions. The subscripts R and L refer to the right and left halves respectively.

Transformation from Physical Components to Cyclic Components (DIH)

(See [Eq. 13-82](#) for the corresponding equations for the ROT option.)

$$\left. \begin{array}{l} U^{kc} \\ U^{kc*} \end{array} \right\} = \frac{\delta}{N} \sum_n [U_{n,R} \cos(n-1)ka \pm U_{n,L} \cos(N-n)ka] \quad \text{Eq. 13-98}$$

$$\left. \begin{array}{l} U^{ks} \\ U^{ks*} \end{array} \right\} = \frac{\delta}{N} \sum_n [U_{n,R} \sin(n-1)ka \pm U_{n,L} \sin(N-n)ka]$$

Here,

$$\delta = \begin{cases} 1/2 & \text{for } k = 0, N/2 \\ 1 & \text{for } 1 \leq k \leq K_L \end{cases} \quad \text{Eq. 13-99}$$

and the positive sign is associated with the unstarred terms.

Transformation from Cyclic Components to Physical Components (DIH)

(See [Eq. 13-95](#) for the corresponding relation for the ROT option.)

$$\begin{aligned}
 U_{n,R} &= \sum_k (U^{kc} + U^{kc*}) \cos(n-1)ka + (U^{ks} + U^{ks*}) \sin(n-1)ka \\
 U_{n,L} &= \sum_k (U^{kc} + U^{kc*}) \cos(N-n)ka + (U^{ks} + U^{ks*}) \sin(N-n)ka \quad \text{Eq. 13-100} \\
 0 &\leq k \leq \frac{N}{2}
 \end{aligned}$$

Boundary Conditions and Problem Solution (DIH)

(See [Eq. 13-82](#) for the corresponding boundary conditions for the ROT option.)

Side 1, EVEN

$$\begin{aligned}
 U^{kc*} &= 0 \\
 U^{ks} &= 0
 \end{aligned} \quad \text{Eq. 13-101}$$

Side 1, ODD

$$\begin{aligned}
 U^{kc} &= 0 \\
 U^{ks*} &= 0
 \end{aligned} \quad \text{Eq. 13-102}$$

Side 2, EVEN

$$\begin{aligned}
 U^{kc} \sin \frac{ka}{2} + U^{ks} \cos \frac{ka}{2} &= 0 \\
 U^{kc*} \cos \frac{ka}{2} - U^{ks*} \sin \frac{ka}{2} &= 0
 \end{aligned} \quad \text{Eq. 13-103}$$

Side 2, ODD

$$\begin{aligned}
 U^{kc} \cos \frac{ka}{2} - U^{ks} \sin \frac{ka}{2} &= 0 \\
 U^{kc*} \sin \frac{ka}{2} + \cos \frac{ka}{2} &= 0
 \end{aligned} \quad \text{Eq. 13-104}$$

In static analysis with cyclic symmetry, the equation

$$[\bar{K}_{aa}]^K \{\bar{u}_a\}^K = \{\bar{P}_a\}^K$$

is solved in SSG3 (Static Solution Generator – Part 3) for $\{\bar{u}_a\}^K$

$$[\bar{K}_{aa}]^K = [G_{ck}^T K_{aa} G_{ck}] + [G_{sk}^T K_{aa} G_{sk}]$$

and

$$\{\bar{P}_a\}^K = [G_{ck}^T]\{\bar{P}\}^{Kc} + [G_{sk}^T]\{\bar{P}\}^{ks}$$

are formed in CYCLIC3 (Cyclic Symmetry Transformation – Module 3).

The symmetrical components of the loads $\{\bar{P}\}^{kc}$ and $\{\bar{P}\}^{ks}$ are subsets of $\{\bar{P}\}^x$ which is obtained in CYCLIC3 by multiplying the load vector formed in SSG1 by the GFORE matrix formed in CYCLIC1 module

$$\{\bar{P}\}^x = \{P_l\}\{GFORE\}$$

where $\{P_l\}$ is formed in SSG2 (Static Solution Generator – Part 2) and the transformation matrix [GFORE] is formed in CYCLIC1 module.

The cyclic component constraint data (CYCD) used in CYCLIC3 for the formulation of the transformation matrices $[G_{ck}]$ and $[G_{sk}]$ are generated in CYCLIC2 (Cyclic Symmetry – Module 2).

The solution $\{u_a\}^K$ is transformed to symmetric components in CYCLIC4 as follows:

$$\begin{aligned}\{u_x\}^{kc} &= [G_{ck}]\{\bar{u}_a\}^K \\ \{u_x\}^{ks} &= [G_{sk}]\{\bar{u}_a\}^K\end{aligned}$$

The symmetric components for each harmonic index are appended to form $\{\bar{u}\}^x$ in the CYCLIC4 module. The physical segment displacements are obtained as follows:

$$\{u_a\} = \{\bar{u}\}^x [GBACK]$$

where the transformation matrix is formed in the CYCLIC1 module.

In the case of vibration analysis with cyclic symmetry the following equation is used in READ to determine the eigenvalues and eigenvectors,

$$[\bar{K}_{aa} - \lambda \bar{M}_{aa}]\{\bar{u}\}^K = 0$$

where $[\bar{M}_{aa}]^K$ is formed in a manner similar to $[\bar{K}_{aa}]$ in CYCLIC3. The symmetrical components of the eigenvectors are recovered in CYCLIC4 and the physical segment data is recovered using the GBACK1 matrix created in CYCLIC4 module.

Since all of the transformation matrices are extremely sparse, none of the matrix multiplications indicated above will require large amounts of computer time. The most significant operation is the triple product associated with the determination of $[\bar{K}_{aa}]^K$. In the usual application this matrix is dense and approximately a-size. The computer time for the triple product can be estimated as ten matrix packing operations on a full matrix of a-size.

Using Cyclic Symmetry

Many structures, including pressure vessels, rotating machines and antennas for space communications, are made up of virtually identical segments that are symmetrically arranged with respect to an axis. As shown in [Figure 13-50](#) and [Figure 13-51](#), there are two types of cyclic symmetry: simple rotational symmetry, in which the segments do not have planes of reflective symmetry and the boundaries between segments may be general doubly-curved surfaces; and dihedral symmetry, in which each segment has a plane of reflective symmetry and the boundaries between segments are planar. The use of cyclic symmetry allows the user to model only one of the identical substructures. There will also be a large saving of computer time for most problems.

The total model consists of N identical segments which are numbered consecutively from 1 to N. The user generates the model for one segment, using regular elements and standard modeling techniques. All other segments and their coordinate systems are automatically rotated to equally spaced positions about the polar axis by MSC.Nastran. The boundaries must be conformable; i.e., when the segments are put together, the grid points and the displacement coordinate systems of adjacent segments must coincide. This is easiest to ensure if a cylindrical or spherical coordinate system is used, but such is not required.

The two boundaries are called Sides 1 and 2. In the case of rotational symmetry, Side 2 of segment n is connected to Side 1 of segment n+1, as shown in [Figure 13-49](#). In the case of dihedral symmetry, Side 1 is on the boundary of the plane of overall reflective symmetry, and Side 2 is on the midplane or reflective plane of the segment, as shown in [Figure 13-51](#). In either case, the grid point numbers on Sides 1 and 2 must be specified on the Bulk Data entry, CYJOIN.

The type of symmetry (rotational, dihedral, or axisymmetric) and the number of segments in the model are specified on the Bulk Data entry CYSYM. Grid points that lie on the axis of symmetry must be listed on the CYAX Bulk Data entry. The harmonics to be used in the cyclic symmetry analysis are specified on the HARMONICS Case Control command. The HARMONICS command must be

present in Case Control and it must appear above the subcase level. In buckling analysis the program limits the static preload to the zero harmonic and the information on the HARMONICS command is used only for the eigenvalue analysis.

No restrictions are placed on the use of the single-point constraint, the multipoint constraint, or the OMIT feature. Constraints between segments are automatically applied to the degrees-of-freedom at grid points specified on CYJOIN Bulk Data entries which are not otherwise constrained. The SPCD Bulk Data entry may be used to vary the magnitude of enforced displacements for each of the segments.

In the case of static analysis, the OMIT feature may be used to remove all degrees-of-freedom at internal grid points without any loss of accuracy. Since this reduction is applied to a single segment prior to the symmetry transformations, it can greatly reduce the amount of subsequent calculation. In the case of vibration analysis, the OMIT feature is used in the usual way to reduce the size of the analysis set and involves the usual approximations. The SUPORTi entries for the free bodies cannot be used with cyclic symmetry.

Provision is made for the use of rigid body supports with the use of the CYSUP Bulk Data entry. All supported degrees-of-freedom for a rigid body must be specified at a single grid point. Additional restrictions on the location of the support point and the uses of coordinate systems are given on the CYSUP Bulk Data entry.

Special provision is made in DIH-type problems for reflective symmetry about one or two planes. Symmetrical or antisymmetrical boundary conditions are specified on the DSYM Case Control command. A single plane of reflective symmetry must be with respect to Side 1, and if a second plane of reflective symmetry exists, it must be perpendicular to Side 1. In the case of two planes of symmetry, the model must contain an even number of whole segments.

Static loads are applied to the model with a LOADCYH (harmonic load), LOADCYN (physical segment load), or LOADCYT (AXI option) Bulk Data entry. Only a single subcase is needed for each loading condition and it must contain the keyword LOAD, TEMP(LOAD), or DEFORM to select a LOADCYH, LOADCYN, or LOADCYT Bulk Data entry. The LOADCYH and LOADCYN entries in turn reference FORCE, MOMENT, PLOAD, SLOAD, SPCD, DEFORM, and TEMP entries. The LOADCYH entry may also reference an RFORCE or GRAV entry in order to specify centrifugal or gravity loading, respectively. The LOADCYT entry is used only with the AXI option to specify loads as a function of azimuth angle.

The NOUTPUT Case Control command is used to request physical output within a subcase for all output quantities, such as forces and displacements, requested in the subcase. The HOUTPUT Case Control command is used to request harmonic output

within a subcase for all output quantities requested in the subcase. In the case of thermal or deformation loading, element force and stress output can only be requested consistent with the form of the loading, i.e., physical output if the loads are defined with a LOADCYN Bulk Data entry and harmonic output if the loads are defined with a LOADCYH Bulk Data entry.

The use of a single subcase for each loading condition introduces the need to refer to “coded” subcases for SORT2 output and plotting purposes. Since a single subcase now refers to NSEG segments (in the case of ROT and AXI options) and two times NSEG segments (including Right and Left halves, in the case of DIH option), a unique coded subcase ID is necessary to refer to a particular segment of the structure for plotting purposes in PLOT commands wherever a subcase ID is to be supplied. This coded ID is obtained as follows:

Coded Subcase ID for a Particular Segment

$$= \text{SUBCASE ID} \cdot 10000 + \text{SEGMENT ID} \cdot 10 + \text{zero (for ROT or AXI option)}$$

$$= \text{SUBCASE ID} \cdot 10000 + \text{SEGMENT ID} \cdot 10 + \begin{cases} 1 & \text{(if R segment)} \\ 2 & \text{(if L segment)} \end{cases}$$

Coded Subcase ID for a Particular Harmonic

$$= \left[\text{SUBCASE ID} \cdot 10000 + \text{HARMONIC ID} \cdot 10 + \begin{cases} 1 & \text{(if component = C)} \\ 2 & \text{(if component = S)} \\ 3 & \text{(if component = C*)} \\ 4 & \text{(if component = S*)} \end{cases} \right]$$

Solution of Axisymmetric Problems

The Cyclic Symmetric Solution Sequences provide a convenient and effective procedure for solving axisymmetric problems without the need for special axisymmetric elements. The following procedure is recommended for the solution of static load problems:

1. Select the AXI symmetry option on the CYSYM Bulk Data entry.
2. Select a large number of segments (for example, 180) on the CYSYM Bulk Data entry.
3. Construct a model with two rows of grid points and a single row of elements in the meridional direction.

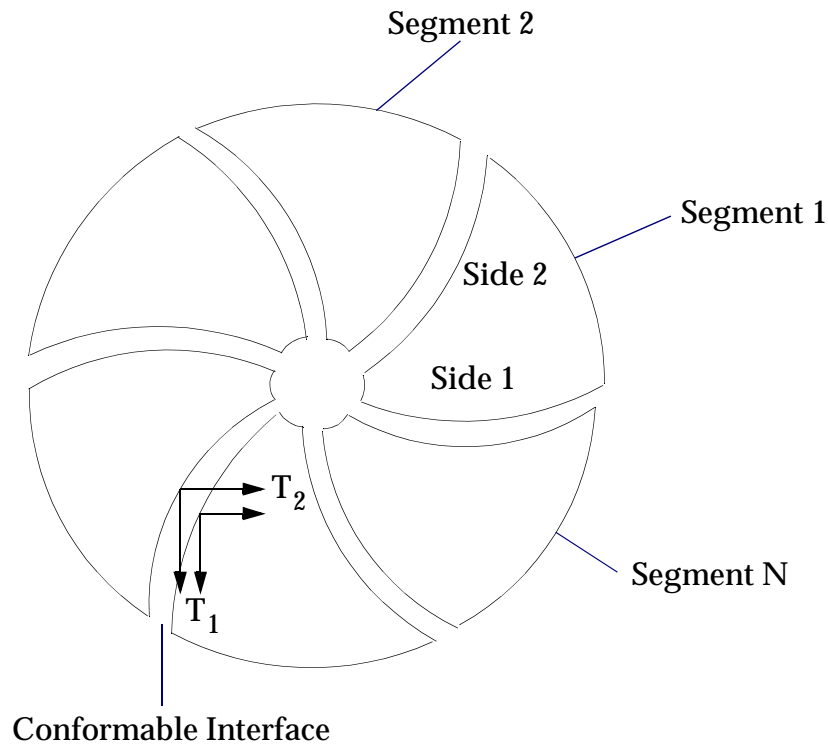
4. Define a set of harmonics on a SET Case Control command. The set of harmonics should be reasonably small for efficiency, but sufficient in size to yield reasonably accurate results.
5. Select the defined set of harmonics on the HARMONICS Case Control command. Convergence of the harmonic output towards zero for the higher harmonics will verify that the number of harmonics specified on the HARMONICS Case Control command and used in the analysis is sufficient.
6. Select a static loading condition with the LOAD, TEMP(LOAD), and DEFORM Case Control commands which reference LOADCYN, LOADCYH, or LOADCYT Bulk Data entries.
7. Place static loads only on Side 1 or on elements. The boundary conditions will take care of loading on Side 2.
8. Prepare the referenced TABLED1 entries if LOADCYT entries are used to specify loads as a function of azimuth angle. The number of x-y pairs in a given table should be sufficient to produce reasonably accurate results when linearly interpolated.
9. Select the physical output for a reasonably small number of segments with the NOUTPUT Case Control command or ALL of the harmonic output with the HOUTPUT Case Control command.

The following procedure is recommended to obtain the vibration modes of an axisymmetric structure:

1. Select the AXI symmetry option on the CYSYM Bulk Data entry.
2. Select a large number of segments (for example, 90) on the CYSYM Bulk Data entry.
3. Construct a model with two rows of grid points and a single row of elements in the meridional direction.
4. Define a set of harmonics on a SET Case Control command where the number of harmonics is reasonably small for efficiency, but sufficient to cover the frequency range of interest.
5. Select the defined set of harmonics on the HARMONICS Case Control command.
6. Select ALL of the harmonic output with the HOUTPUT Case Control command. This will give the eigenvectors, in a concise form, which may be interpreted by the formula

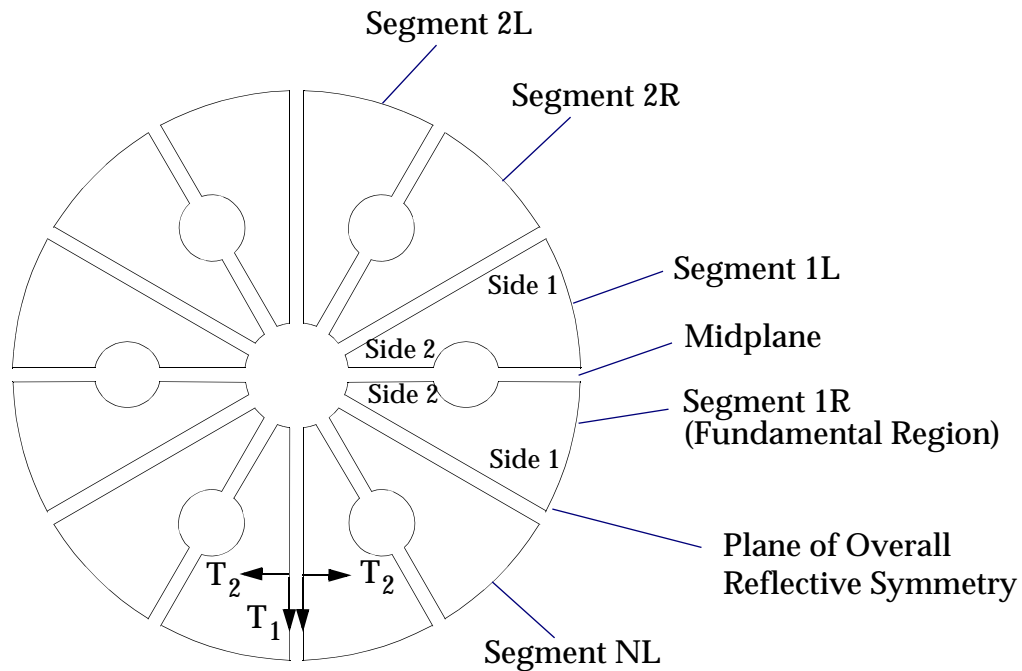
$$U(\phi) = U^{kc} \cos k\phi + U^{ks} \sin k\phi$$

where ϕ is the azimuth angle measured from Side 1 of the fundamental region and κ is the harmonic index.



1. The user models one segment.
2. Each segment has its own coordinate system which rotates with the segment.
3. Segment boundaries may be curved surfaces. The local displacement coordinate system must conform at the joining points. The user gives paired list of points on Side 1 and Side 2 which are to be joined.

Figure 13-50 Rotational Symmetry



1. The user models one-half segment (an R segment). The L half segments are mirror images of the R half segments.
2. Each half segment has its own coordinate system which rotates with the segment. The L half segments use left-hand coordinate systems.
3. Segment boundaries must be planar. Local displacement systems axes, associated with the inter-segment boundaries, must be in the plane or normal to the plane. The user lists the points on Side 1 and Side 2 which are to be joined.

Figure 13-51 Dihedral Symmetry

13.8 p-Elements

13.8.1 Introduction to p-Elements

Applications of p-Elements

One of the main applications of p-version elements is detailed stress analysis. The p-elements have higher-order polynomials, which provide better representation of complex stress fields. For these complex stress fields, the geometry, loads, and boundary conditions must be represented accurately. This includes modeling fillets instead of sharp corners, distributed loads and constraints instead of point loads and constraints, etc. A more detailed model leads to more detailed results.

p-Elements have several modeling advantages. With the higher-order polynomials, the geometry and loads can be represented more accurately. Generally, fewer p-version elements with curvature than h-version elements with straight edges are required.

The accuracy of the analysis is controlled primarily by the polynomial level, not by the element size. Generally the user needs to only use the minimum number of elements necessary to adequately model the geometry, independent of expected result characteristics. Polynomial levels can then be assigned based on areas of the model in which the user is interested, and areas of the model in which the user is not. (Of course, the same polynomial level can be assigned everywhere for a minimum of user involvement but this is not as efficient.) If a more accurate answer is necessary, the polynomial levels can be increased, either manually by the user or automatically by the program, until the answers reach the specified accuracy, without changing the mesh. Adding degrees-of-freedom until the error decreases to a specified level is known as adaptivity. By adding higher-order polynomials instead of refining the mesh, a faster rate of convergence, and therefore fewer iterations, can be achieved.

p-Elements in MSC.Nastran

In MSC.Nastran, p-elements have been added to complement the existing h-elements. These elements can be used in linear static (SOL 101) and normal modes (SOL 103) analysis only. They allow different polynomial orders in the three coordinate directions, and use bubble functions at $p = 1$ and optimum integration algorithms at $p \geq 2$ to achieve better accuracy. The elements will also pass the patch tests and rigid body motion at any allowable p-level. These additional features improve the accuracy over conventional p-elements.

Compatibility with elements of different p-orders, and with the existing solid elements, is automatically enforced. The user does not have to write constraint equations or use special elements to provide displacement continuity. Loads and boundary conditions can be applied to the elements themselves, or to the geometry on which they are defined.

The definition of geometry has been added to MSC.Nastran. This geometry can be defined with midside points, as in the existing elements; with rational parametric cubics or equations; or even with any external geometry. A facility has been set up to invoke remote procedure calls to any independent program and database that provides a toolkit for geometry queries, which would allow proprietary geometries of other pre-processors or CAD packages to be used. Associativity has been added so that the elements are defined on the geometry, and the element geometry calculated accordingly.

The error estimator requires only a single analysis, not the difference between two analyses. For linear elements, it is based on the grid-point stress discontinuity; for higher polynomial orders, it considers the contribution of the additional terms for the individual edge, face, and body functions. Therefore it provides efficient information because of the directional sensitivity within the elements.

The adaptivity method is very flexible, working on an element-by-element basis. Using the error estimator, the polynomial level within each element can be varied independently in the three directions. At the most basic level, the edges are varied, since they determine the face and body functions. Different error criteria may be used in different regions, allowing areas of primary interest to have lower error tolerances. Multiple subcases may be used.

Combining p- and h-Elements

The main advantage of implementing p-elements in MSC.Nastran is the ability to combine them with the existing h-elements for global/local analysis. The mesh could consist mostly of h-version elements, with p-version elements in the areas of interest. This method of global/local interface is the most accurate, because it provides both stiffness and loads information, rather than transferring just boundary displacements or tractions into the local model.

h-version elements still have several advantages. They are better for global behavior, such as loads analysis, where the exact geometry is not used. They are better for strongly singular problems, such as nonlinear problems. Approximations, such as sharp corners, point loads, point constraints, MPC's, etc. may be used. For a given iteration, they tend to have shorter solution times and use less disk space. Finally, h-elements are a very mature technology.

Putting both approaches in the same program combines the advantages for general problems. The p-elements may be connected directly to the existing h-elements, and continuity is automatically enforced. This provides global/local analysis in a single run.

Most general problems have both some global and some local aspects; i.e., regions where h-elements and p-elements, respectively, are appropriate. Therefore the best approach is combined h- and p-adaptivity, with different criteria in different parts of the model.

13.8.2 p-Version Capabilities

Following is a limited description of the Bulk Data entries and Case Control commands for the p-version elements. [Table 13-20](#) contains a comparison with the conventional h-version elements. A full description of the entries may be found in the *MSC.Nastran Quick Reference Guide*.

Solution Sequences Supported

Solution Sequences 101, 103, 107, 108, 109, 110, 111, 112, and 101 (Heat Transfer).

Geometry

With the h-version elements, geometry was determined by the addition of midside grid points to the element definition. With the p-version elements, geometrical entities have been introduced into MSC.Nastran. The edges and faces of elements can be defined to lie on curves and surfaces, which can span an arbitrary number of elements, and midside grid points do not have to be defined.

The Bulk Data entries to define the associativity between the finite element model and the geometrical model are the FEEDGE and FEFACE entries for the edges and faces, respectively. These entries reference the grids that make up the edges and faces, not the elements themselves, and therefore do not have to be repeated for each element that shares the edge or face. The old method of midside grid points is still available for the FEEDGE entry, which would reference one or two GRIDs or POINTs. The POINT entry is a new entry, very similar to the GRID entry, except that it defines a location in space with no associated degrees-of-freedom.

The more useful way of defining the FEEDGES and FEFACES is to have them reference geometrical curves and surfaces, defined by the new GMCURV and GMSURF entries. GMCURVs and GMSURFs are functions of one or two parametric variables, respectively, and can be defined as rational parametric cubics or general equations. For instance, a hole could be defined with the parametric equations for a

cylinder, which all the element faces on the hole would then reference. The GMCURVs and GMSURFs could also reference external geometry from a CAD or other database, using remote procedure calls with an external programming interface. When geometry has been specified, it is approximated as cubics for the element-level computations. Using cubics ensures that the elements pass the rigid-body and patch tests.

The two different methods of defining the geometry are shown in [Figure 13-52](#).

Convective coordinate systems may be defined on the FEEDGEs, FEFACEs, GMCURVs, and GMSURFs. The directions of the systems are normal and tangent to the entities at every point. These coordinate systems can then be used for material properties, loads, boundary conditions, etc.

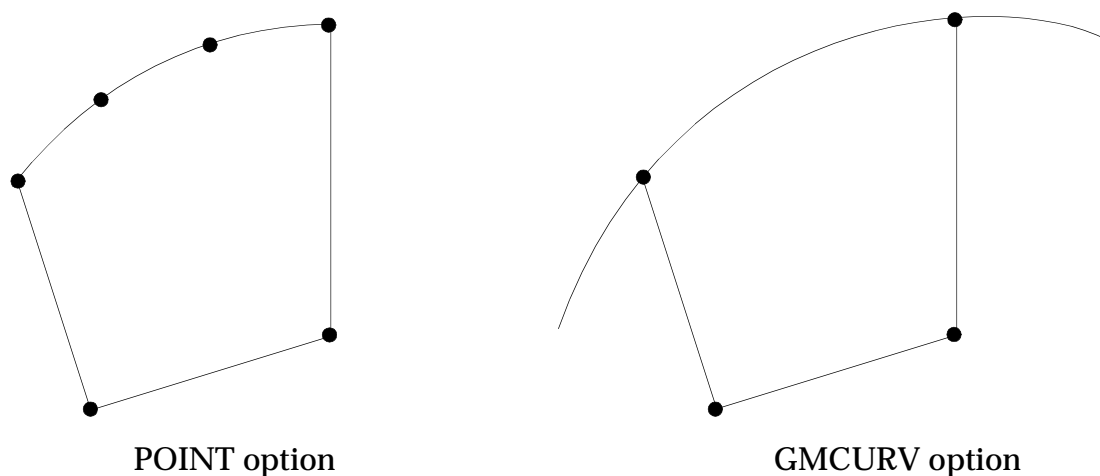


Figure 13-52 Methods of Defining Element Geometry

Elements

Only the CHEXA, CPENTA, CTETRA, CQUAD, CTRIA, and CBEAM elements may be defined as p-elements. Midside grid points may not be specified on p-version elements; curvature is defined by the FEEDGE or FEFACE entries, as previously described.

The standard Bulk Data entries, PSOLID, PSHELL, and PBEAM are used to define the element properties. The p-version properties are defined in the PVAL and ADAPT entries, which are described later.

The p-version elements may be joined to existing h-version elements with a few considerations. FEEDGEs and FEFACEs may not be defined on the interface, since the h-elements can not accept them. Midside grid points may not be defined on the interface, since the p-elements can not accept them. The polynomial levels of the p-elements will be lowered automatically so that displacement continuity is enforced.

Materials

The standard isotropic, orthotropic, and anisotropic materials are available for p-elements. Orthotropic and anisotropic material properties may be defined in a curvilinear material system.

Loads

For p-version elements in detailed stress analysis, it is important to use distributed loads. Concentrated forces cause singularities in the stress field, and therefore should be avoided. The GMLOAD Bulk Data entry has been defined for the p-version elements. It can apply a distributed load to FEEDGEs, FEFACEs, GMCURVs, or GMSURFs. The load can be distributed as constant, linear, quadratic, cubic, equation, or table, and can be given in any coordinate system. The PLOAD4 and GRAV entries can also be applied to p-elements. The FORCE entry can be used, but should only be used where the adjacent elements have a fixed p-level and the results are not of interest.

For thermal loads, the TEMPF entry allows specification of an equation or table to describe the variation of the temperature field over the elements. The TEMP and TEMPD entries may also be used, but will result in linear interpolation of the temperature from the grids. The TEMPP1 entry defines a constant temperature and temperature gradient distribution in a shell element. The TEMPRB entry defines a linear temperature and temperature gradient distribution in a beam element.

Boundary Conditions

As with the loads, it is important to use distributed boundary conditions for p-version elements in detailed stress analysis. Concentrated boundary conditions also cause singularities in the stress field and should be avoided. The GMSPC and GMBC Bulk Data entries have been defined for the p-version elements. These can apply distributed boundary conditions to GRIDs, FEEDGEs, FEFACEs, GMCURVs, or GMSURFs. Note that they should be applied to GRIDs only to prevent rigid-body motion. The imposed boundary conditions can be distributed as constant, linear, quadratic, cubic, equation, or table, and can be given in any coordinate system. The

current SPC and SPC1 entries can be used, but should only be used to prevent rigid-body motion or used where the adjacent elements have a fixed p-level and the results are not of interest.

p-Version Adaptivity

In order for an element to be recognized as a p-version element, it must be assigned the polynomial values in the new PVAL Bulk Data entry, which is referenced in the ADAPT Bulk Data entry. In order for the p-element to be adaptive, the adaptivity parameters must be assigned in the ADAPT entry.

The new PVAL Bulk Data entry assigns the polynomial levels for the three directions to a single element or a set of elements, and is referenced in the ADAPT Bulk Data entry. By default, the three directions are along the element edges. If a coordinate system is specified, the polynomial level of each edge is set to a weighted combination of the coordinate directions. There may be multiple PVALs with the same ID; all of the elements do not have to use the same entry. Each p-version element must have a starting, minimum, and maximum PVAL ID; however, these do not need to be unique. Since the polynomial levels can not decrease, the starting and minimum PVAL IDs can be the same. If the starting and maximum PVAL IDs are the same, the element will have a fixed p-level.

After the p-value distribution has been defined, all of the values will be resolved to ensure displacement continuity. This includes selecting the highest p-level specified for elements with common edges, decreasing the p-level for p-elements adjacent to h-elements, and increasing the p-level on curved edges to adequately map them.

The new ADAPT Bulk Data entry controls the p-adaptivity. The ADAPT Bulk Data entry is referenced with the ADAPT Case Control command, and there may only be one unique ADAPT command in the Bulk Data file. The ADAPT entry contains the IDs for the starting, minimum, and maximum p-value distributions and the maximum number of iterations. It also contains the adaptivity parameters, which may be assigned differently to different sets of elements. These sets of elements do not have to correspond with any sets used in the PVAL entries. The adaptivity type may be element-by-element p-adaptive, uniform p-adaptive, no change, or list of p-distributions. For each set in the ADAPT entry, the error estimator method, error tolerance, and stress and strain tolerances are specified. If a p-version element is not included in one of the sets, it is not adaptive.

For a non-adaptive p-version solution, the ADAPT Bulk Data entry may specify a maximum of one iteration, or may specify an adaptivity type of no change for all the elements. A third way is to use the PSET Bulk Data entries, which have the same format as the PVAL entry, and reference them with the ADAPT Case Control command.

Data Recovery

Since the p-version analysis may have multiple iterations, the output control must have the capability of differing among the iterations. For this reason, the OUTPUT Bulk Data entry was added. For a given set of elements, the conventional displacements, stresses, and strains, as well as the new element errors and polynomial values, may be printed, plotted, or punched for the given iteration. The OUTPUT entry is referenced by the DATAREC Case Control command. The conventional Case Control commands still apply to all iterations, but the OUTPUT entry has precedence.

The p-version elements also tend to be larger than h-version elements, and have higher-order distributions of displacements, stresses, and strains. Therefore results only at the grid points are not adequate, especially for most plotting packages, which use linear interpolation. For data recovery, the p-version elements are divided into view-grids and view-elements in order to better visualize the results. The new OUTRCV Bulk Data entry defines the number of view-elements in each element direction and the output coordinate system for the view-grids and view-elements. The OUTRCV entry is referenced by the OUTRCV Case Control command. The coordinates of the view-grids may be output using the VUGRID Case Control command.

Table 13-20 Bulk Data Entries for h-Elements and p-Elements

	h-Version Elements	p-Version Elements
Geometry		GMCURV GMSURF
Geometrical Associativity		FEEDGE FEFACE
Coordinate Systems	CORDij	CORDij GMCORD
Grids Points	GRID	GRID POINT

Table 13-20 Bulk Data Entries for h-Elements and p-Elements (continued)

	h-Version Elements	p-Version Elements
Elements	CHEXA CPENTA CTETRA CQUAD CTRIA CBEAM	CHEXA CPENTA CTETRA CQUAD CTRIA CBEAM
Element Properties	PSOLID PSHELL PBEAM	PSOLID PSHELL PBEAM
Material Properties	MATi	MATi
Loads		GMLOAD
Thermal Loads	PLOAD4 FORCE TEMP, TEMPD TEMPP1 TEMPRB	PLOAD4 FORCE* TEMPF TEMP, TEMPD TEMPP1 TEMPRB
Gravity Loads	GRAV	GRAV
Constraints		GMSPC
Boundary Conditions	SPC, SPC1 SPCD	SPC, SPC1* GMBC SPCD*
Polynomial Values		PVAL PSET
Adaptivity Control		ADAPT
Data Recovery		OUTPUT OUTRCV
Miscellaneous		DEQATN TABLE3D
Sets Definition		SETS DEFINITION

* Should only be used under specific conditions.

13.8.3 p-Element Example Problems

This section demonstrates the p-element and p-adaptivity capabilities. Several examples are included in this section to show how p-elements are used. Most of the input files of these examples have been annotated and are included in the Test Problem Library (TLP) directory. The detailed descriptions of the input commands and the interpretation of the results can be found in the [MSC.Nastran Quick Reference Guide](#), the [MSC.Nastran 2005 Release Guide](#), and [Section 3.13](#) and [Section 5.11](#) of this manual.

Examples

Each example contains the following parts:

Objectives	Outlines the problem objectives.
Keywords	Serves as index and show what p-element related Bulk Data entries being used.
Problem Description	Describes the geometry and data of the problem.
Theoretical Results	Shows the closed form solution, if available.
Setup	Describes what and how to use the Bulk Data entries.
Results	Compares the calculated results with theory.
References	

13.8.4 Square Plate with a Circular Hole

Objectives

This example computes the stress concentration factor of a square plate with a circular hole under axial loads. The objective of this problem is to validate the stress concentration factor for a square plate of 2.73 around the hole.

Problem Description

Plate size	= 10 in x 10 in
Thickness	= 0.1 in
Hole radius	= 2.0 in
Modulus of Elasticity	= 1.0E7 psi
Poisson's ratio	= 0.3

Figure 13-53 shows the quarter model that requires only two CHEXA solid p-elements.

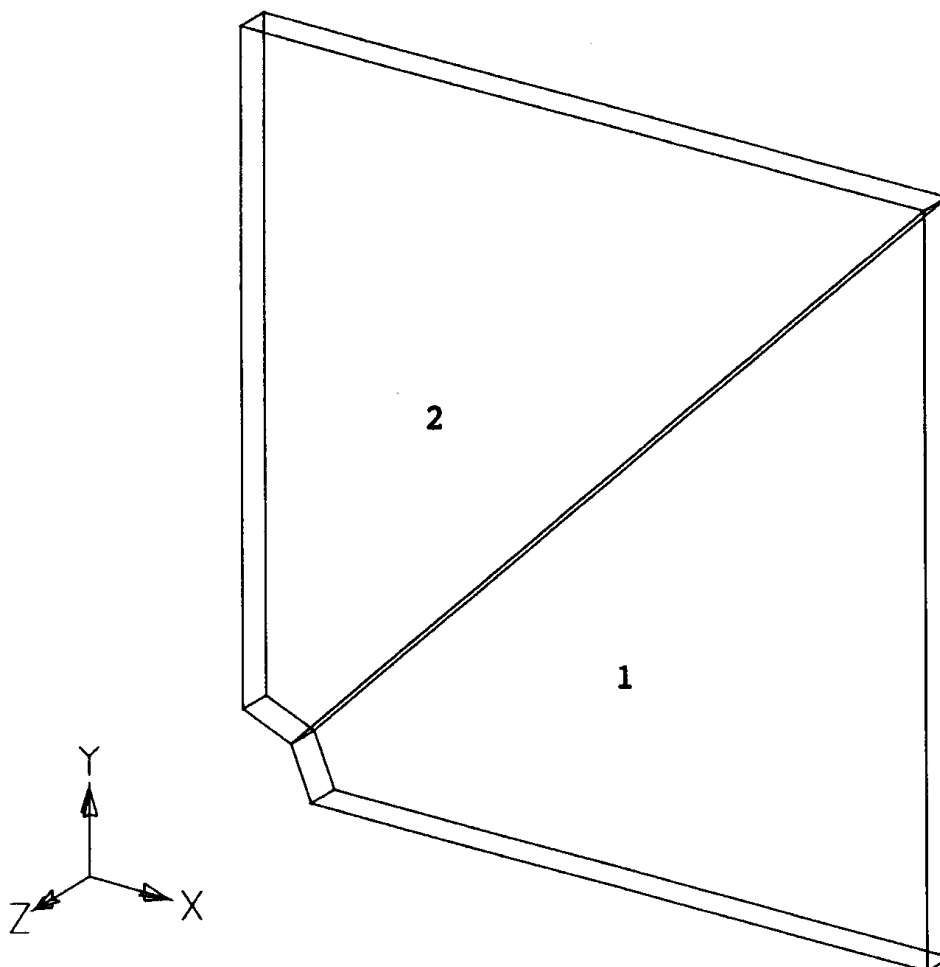


Figure 13-53 Geometry of a Square Plate with a Circular Hole

A portion of the file which has been annotated is shown in **Listing 13-14**. The complete input file, PEV0101.dat, can be found in the TPL directory.

Theoretical Results

Stress concentration factor of 2.73 around the hole as described in Reference 6.

Setup

The Bulk Data entry FEEDGE is used to describe the circular arc of the hole. The quadratic order is used for demonstration purposes; a cubic order curve can be easily accommodated with an additional point for each curve. The load and boundary conditions are also specified using FEFACE.

Since the file has been heavily annotated, the description of the Bulk Data entries is not repeated here. Please refer either to the following partial list or the complete input file in the TPL directory for detailed descriptions.

Listing 13-14 Partial Bulk Data Entries

```

TITLE = PLATE WITH HOLE -- QUARTER MODEL (p-element model; Point option)
SUBTITLE = p-adaptivity; maximum allowable p-value 5X5X3
$
$   p-elements Case Control commands
$
$   Select Bulk Data ADAPT ID for adaptivity control
$
ADAPT          = 150
$
$   Request output destination(s) of locations (i.e. coordinates)
$   for p-element data recovery points
$
VUGRID(PLOT,PUNCH)=all
$
$   Request output (STRESS, STRAIN, DISP, PVAL) for p-elements only
$   DATAREC = n, where n refers to a Bulk Data OUTPUT ID
$
DATAREC = 301
$
$   Select output options for p-elements
$   OUTRCV = n, where n defines an OUTRCV ID in Bulk Data
$
OUTRCV = 401
$
$   Select set(s) of elements for adaptivity and stress output
$   SETS DEFINITION must be defined at the end of the Case Control
$   SET 501 defines a set of all elements
$   referenced by ELSETID in ADAPT entry for ERRTOL of 10%
$
SETS DEFINITION
set 501 = all
$
BEGIN BULK
$
$ p-elements Bulk Data
$
$   Adaptivity control
$   PSTRTID specifies the PVAL ID 120 for the starting pvalues p1, p2 and
p3
$   for the element SET 501 (all elements)
$   PMINID specifies the PVAL ID 120 for the minimum pvalues p1, p2 and
p3
$   PMAXID specifies the PVAL ID 150 for the maximum pvalues p1, p2 and
p3
$
$ADAPT  SID                PSTRTID PMINID  PMAXID
$      PART          ELSETID

```

```

$
ADAPT 150 120 120 150
PART=ALL,ELSET=501

$
$ Specify p-levels for element SET 501 (all elements)
$ POLY1, POLY2 and POLY3 correspond to p1, p2 and p3 respectively and
$ can have different p-values
$ p-values specified by PMINID should be = PSTRTID < PMAXID
$ starting p-level p1=p2=p3=2
$ minimum p-level p1=p2=p3=2
$ maximum p-level p1=p2=5 p3=3
$
$PVAL ID POLY1 POLY2 POLY3 SETTYP ID
PVAL 120 2 2 2 SET 501
PVAL 150 5 5 3 SET 501
$
$ Select various output options;
$ Displacements, Stresses, Strains and p-values
$
$ SID
$ ELSET=n, DISP=(PRINT,PLOT,PUNCH),
$ STRESS=(PRINT,PLOT,PUNCH), STRAIN=(PRINT,PLOT,PUNCH),
$ PVAL=(PRINT,PLOT,PUNCH), BY=n
$
OUTPUT 301
ELSET=501, DISP=PRINT, STRESS=PRINT
STRAIN=PLOT, PVAL=PRINT, BY=1
$
$ Define various output options;
$ for Displacements, Stresses and Strains
$
$ SID SETID
$ CID= 0 or n (default 0; basic coordinate system)
$ VIEW=i*j*k (default 3*3*3)
$ For less output, reduce i,j and k to 2 or 1
$
OUTRCV 401 501
VIEW=2*2*2
$
CHEXA 1 1 1 25 28 4 101 125
+ 128 104
CHEXA 2 1 32 4 28 56 132 104
+ 128 156
$
$ The following geometry specifications demonstrate the use of
$ different options on specifying actual geometry
$ to achieve more accurate results
$
$ Define FEEDGES and FEFACES to specify
$ geometry, loading and constraints
$
$ Use POINT, FEEDGE to describe curves 104, 108, 204 and 208
$
POINT 3 1.7305341.0026220.0

```

```

POINT    30                1.0026221.7305340.0
POINT    103               1.7305341.0026221.0
POINT    130               1.0026221.7305341.0
$
$      EDGEID  G1      G2                GEOMIN  ID1
FEEDGE 104      4      1                POINT   3
FEEDGE 204      4      32               POINT   30
FEEDGE 108     104     101              POINT   103
FEEDGE 208     104     132              POINT   130
$
$      Specify FEFACE for BC and loads
$
$      FACEID  G1      G2      G3      G4
FEFACE 1001    1      25     125    101
FEFACE 1002    25     28     128    125
FEFACE 2002    56     156    132    32
$
$      Use GMSPC,GMLOAD and FEFACE to specify BC and loads
$      on geometry instead of element data
$
$      Boundary Conditions
$
$      SID      C      ENTITY  ID
GMSPC  999    2      FEFACE 1001
GMSPC  999    1      FEFACE 2002
$
$      LOADS
$
$      LID                N1      N2      N3      ENTITY  ID      METHOD
GMLOAD 998            1.0    0.0    0.0    FEFACE 1002  CONSTANT
          90.
$
ENDDATA

```

Results

Table 13-21 compares MSC.Nastran results with the theory, and they agree very well with only two p-elements. Closer results could have been obtained with more elements or higher p-values.

Table 13-21 Comparison of MSC.Nastran and Theoretical Results

Stress Concentration Factor		Percent Difference
Theory	MSC.Nastran	
2.73	2.77	1.5%

13.8.5 Spherical Structure Under Internal Pressure

Objectives

This example computes the effects of internal pressure on a spherical structure. The deflections at the inner and the outer surfaces are compared with the theoretical results.

Problem Description

Internal pressure (P)	= 3000.0 psi
Inner radius (b)	= 5.0 in
Outer radius (a)	= 10.0 in
Modulus of Elasticity (E)	= 1.0E7 psi
Poisson ratio (ν)	= 0.3

Theoretical Results

The theoretical results on radial deflections can be found in Reference 6 and are listed as follows:

$$\text{Inner radial deflection} = \frac{Pb}{E} \left[\frac{(1 - \nu)(a^3 + 2b^3)}{2(a^3 - b^3)} + \nu \right]$$

$$\text{Outer radial deflection} = \frac{Pa}{E} \frac{3(1 - \nu)b^3}{2(a^3 - b^3)}$$

Setup

This example demonstrates how the higher cubic order of the geometry for a spherical structure can be specified using the EQUATION option of the GMCURV and GMSURF Bulk Data entries. An octant model is analyzed with four CPENTA elements. The boundaries of the CPENTA elements are specified along the great circles. **Figure 13-54** shows the geometry and the element boundaries.

Partial Bulk Data entries are given in **Listing 13-15**. Equations of the inner circle along the sphere are described using Bulk Data entries GMCURV and DEQATN. The equations (IDs 1, 2, and 3) are the spatial locations of the curve along the great circle (circumference of a sphere) expressed in terms of the radius and an angle (U). The first

and second derivatives of the equations are described by DEQATN entries 4 to 9. The first and second derivatives are optional inputs. If they are missing, they will be calculated by the program.

The GMCURV entry specifies the great circle geometry on the x-y plane with equations 1 through 9 for the curve ID 1, and ranges from 0 to 90 degrees (0 to 1.57 radians). Similarly, the geometry on the y-z and x-z planes, curves 2 and 3, can be specified using the same equations with different orders. Note that the labels and arguments of the equations may be chosen for user convenience.

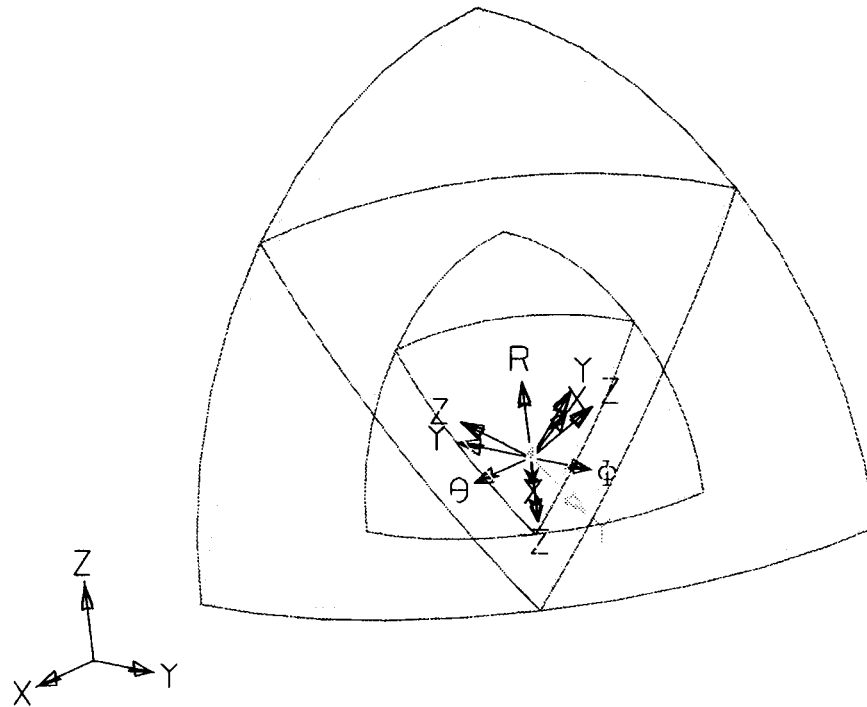


Figure 13-54 Spherical Structure

Listing 13-15 Partial Bulk Data Entries

```

$
$ ***** CURVE AND SURFACE GEOMETRY *****
$
$ Inner circle equations
$
$
DEQATN 1      X(U) = 5.0*COS(U)
DEQATN 2      Y(U) = 5.0*SIN(U)
DEQATN 3      Z(U) = 0.0
DEQATN 4      XP(U) = -5.0*SIN(U)
DEQATN 5      YP(U) = 5.0*COS(U)
DEQATN 6      ZP(U) = 0.0
DEQATN 7      X2P(U) = -5.0*COS(U)
DEQATN 8      Y2P(U) = -5.0*SIN(U)

```

```

DEQATN  9          Z2P(U) =  0.0
$
$ Inner circle on x-y plane
$
GMCURV  1          MSCGRP1  0          0          +
+      EQUATION  0.0    1.570796327    +
+      1 2 3    4 5 6    7 8 9
$
$ Inner circle on y-z plane
$
GMCURV  2          MSCGRP1  0          0          +
+      EQUATION  0.0    1.570796327    +
+      3 2 1    6 5 4    9 8 7
$
$ Inner circle on x-z plane
$
GMCURV  3          MSCGRP1  0          0          +
+      EQUATION  0.0    1.570796327    +
+      2 3 1    5 6 4    8 9 7
$

```

Geometry of the inner surface is described using GMSURF and DEQATN entries. The equations (IDs 41, 42, and 43) are the spatial locations of the surface using a spherical coordinate system expressed in terms of radius and two angles (U and V). The first and second derivatives of the equations are described by DEQATNs 44 to 59. Again, the first and second derivatives are optional inputs.

The GMSURF entry specifies the surface geometry with equations 41 through 59 for the surface ID 1, and ranges from 0 to 90 degrees in two directions. Besides describing geometry, the specification of this surface facilitates the application of internal pressure with the GMLOAD command. The GMLOAD specifies a constant pressure load of 3000 psi on the surface GMSURF 1. The commands are shown below.

```

$
$ Inner sphere
$
DEQATN  41          SX(U,V) =  5.0*SIN(U)*COS(V)
DEQATN  42          SY(U,V) =  5.0*SIN(U)*SIN(V)
DEQATN  43          SZ(U,V) =  5.0*COS(U)
DEQATN  44          SXU(U,V) =  5.0*COS(U)*COS(V)
DEQATN  45          SYU(U,V) =  5.0*COS(U)*SIN(V)
DEQATN  46          SZU(U,V) = -5.0*SIN(U)
DEQATN  47          SXV(U,V) = -5.0*SIN(U)*SIN(V)
DEQATN  48          SYV(U,V) =  5.0*SIN(U)*COS(V)
DEQATN  49          SZV(U,V) =  0.
DEQATN  51          SXUU(U,V) = -5.0*SIN(U)*COS(V)
DEQATN  52          SYUU(U,V) = -5.0*SIN(U)*SIN(V)
DEQATN  53          SZUU(U,V) = -5.0*COS(U)
DEQATN  54          SXVV(U,V) = -5.0*SIN(U)*COS(V)
DEQATN  55          SYVV(U,V) = -5.0*SIN(U)*SIN(V)
DEQATN  56          SZVV(U,V) =  0.

```

```

DEQATN  57      SXUV(U,V) = -5.0*COS(U)*SIN(V)
DEQATN  58      SYUV(U,V) =  5.0*COS(U)*COS(V)
DEQATN  59      SZUV(U,V) =  0.
$
GMSURF  1      MSCGRP1 0      0      +
+      EQUATION  0.0  1.570796327  0.0  1.570796327  +
+      41 42 43  44 45 46  47 48 49  +
+      51 52 53  54 55 56  57 58 59  +
$
$ THIS SECTION CONTAINS LOADS
$
GMLOAD  1      1      1.      0.      0.      GMSURF 1      CONST  +
+      3000.
$

```

Since only an octant model is analyzed, the symmetric boundary conditions are required. The following partial commands demonstrate how GMSURF, GMSURF, and DEQATN work together. The GMSURF 201 is expressed in terms of radius (R) and an angle (U), ranging from 5 to 10 inches and 0 to 1.57 radians. GMSURF specifies the constraint in the basic z-direction.

```

$
$ Surface on x-y plane
$
DEQATN  61      SX(R,U) =  R*COS(U)
DEQATN  62      SY(R,U) =  R*SIN(U)
DEQATN  63      SZ(R,U) =  0.
DEQATN  64      SXU(R,U) =   COS(U)
DEQATN  65      SYU(R,U) =   SIN(U)
DEQATN  66      SZU(R,U) =  0.
DEQATN  67      SXV(R,U) = -R*SIN(U)
DEQATN  68      SYV(R,U) =  R*COS(U)
DEQATN  69      SZV(R,U) =  0.
DEQATN  71      SXUU(R,U) =  0.
DEQATN  72      SYUU(R,U) =  0.
DEQATN  73      SZUU(R,U) =  0.
DEQATN  74      SXVV(R,U) = -R*COS(U)
DEQATN  75      SYVV(R,U) = -R*SIN(U)
DEQATN  76      SZVV(R,U) =  0.
DEQATN  77      SXUV(R,U) = - SIN(U)
DEQATN  78      SYUV(R,U) =  COS(U)
DEQATN  79      SZUV(R,U) =  0.
$
GMSURF  201     MSCGRP1 0      0      +
+      EQUATION  5.0 10.0  0.0  1.570796327  +
+      61 62 63  64 65 66  67 68 69  +
+      71 72 73  74 75 76  77 78 79  +
$
$ x-y plane in xyz basic
$
GMSURF  1      3      GMSURF 201
$

```


Other inputs, such as FEEDGE and FEFACE, are used to specify the geometry, the element boundaries, loads, and boundary conditions. For example, the following FEEDGE command is used to specify that the element edge between grids 3 and 4 will follow the geometry of GMCURV ID 1. The complete input file has been annotated and stored in the TPL directory for review.

```

$
$ Inner radius x-y plane
$
FEEDGE 1      3      4      0      GMCURV 1

```

Results

The p-element model requires two adaptivity cycles to converge to the desired error tolerance of 10% at p-value of 4. [Table 13-22](#) tabulates the results with the theory.

Table 13-22 Comparison of Displacements

Results			Percent Difference
Deflections	Theory	MSC.Nastran	
Inner	1.20E-03	1.19E-03	1.0%
Outer	4.50E-04	4.45E-04	1.1%

13.8.6 Motor Blade – Normal Modes

Objectives

A normal mode analysis of a simplified motor blade is analyzed and compared using the h- and p-elements. The natural frequencies between 0 and 2000 Hz are computed.

Problem Description

[Figure 13-55](#) shows the geometry of the motor blade consisting of a base and a blade. The whole assembly part is made of aluminum.

Modulus of Elasticity = 1.0E7 psi

Poisson's ratio = 0.3

Mass density = 2.59E-4 lb/in³

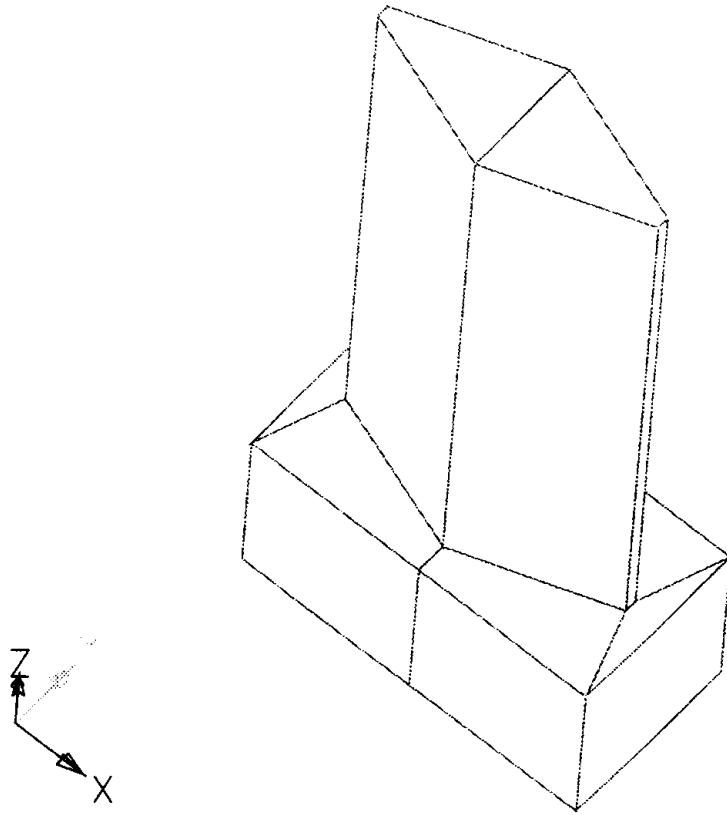


Figure 13-55 Motor Blade – p-Element Model

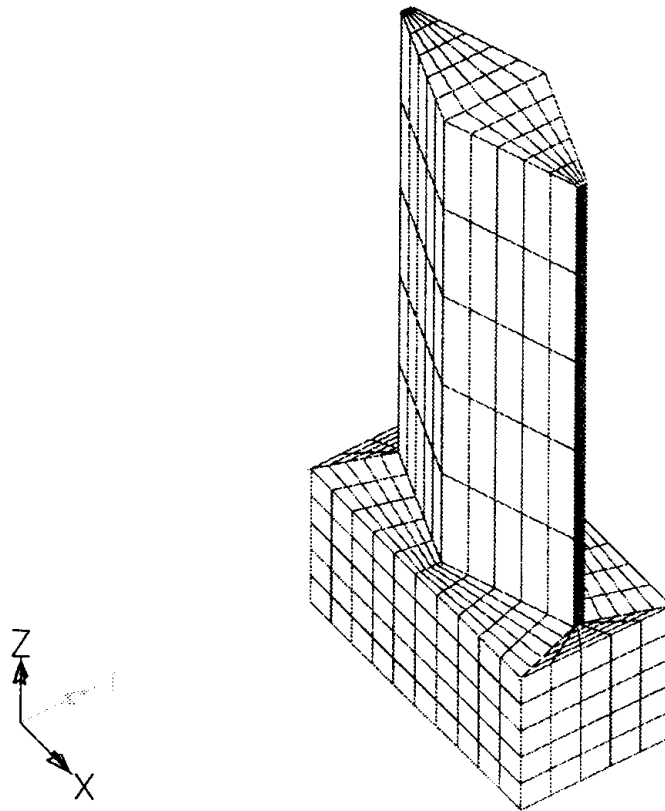


Figure 13-56 Motor Blade -- Fine h-Model

Theoretical Results

There is no theoretical result. Results of three different meshes of h-elements--coarse, medium, and fine--are compared with a p-element model.

Setup

Figure 13-55 and **Figure 13-56** show the p-model and the h-model with fine mesh respectively. Since only the natural frequencies are requested, the geometry of both the h- and p-meshes are simplified and the boundaries are modeled with straight edges. If a detailed stress analysis is required, the geometry should be specified more precisely with higher-order geometry.

Results

All models produce six rigid body modes, so they are not discussed here. **Table 13-23** tabulates the computed elastic natural frequencies between 0.0 to 2000.0 Hz, the number of elements, and the number of degrees-of-freedom with respect to different models. The p-element results agree very well with the fine meshed h-model showing the power and usefulness of the p-elements.

Table 13-23 Comparison of Natural Frequencies of a Motor Blade

Descriptions	Mesh	No. of Elements	No. of DOFs	Elastic Modes			
				1st	2nd	3rd	4th
h-model	Coarse	4	90	622.5	651.2	1204.9	–
	Medium	80	441	604.5	954.7	1289.6	1956.2
	Fine	1250	4878	659.7	1051.5	1404.8	1744.7
p-adaptive model	cycle=1	10	276	737.7	1064.8	1526.9	–
	cycle=2		1161	675.8	1070.8	1426.9	1796.6
	cycle=3		2112	674.6	1071.2	1427.8	1765.2

13.8.7 Torsion of Elliptical Bar

Objectives and Problem Description

This example illustrates torsion of an elliptical bar.

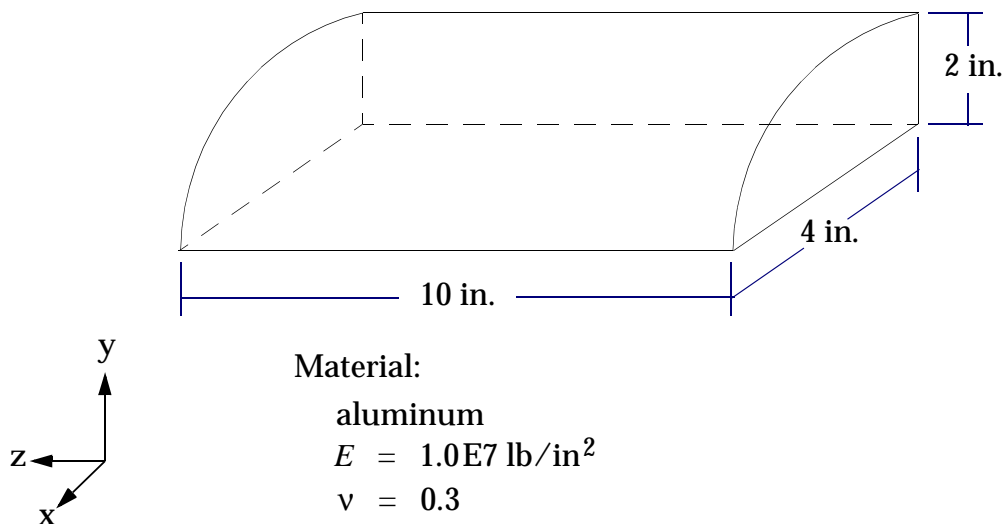


Figure 13-57 Octant of the Elliptical Bar Model

Setup

The model is constructed from four CPENTA elements. There are two coordinate systems used: the cylindrical coordinate system (CORD2C entry) to define grids of the model, and the convective coordinate system (GMCORD entry) to define loads applied to the bar.

The elliptical cross section of the model is described using GMCURV and GMSURF entries, both with the equation method. For the GMCURV entry:

The general ellipse equation:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

as shown in [Figure 13-58](#).

Define x and y in polar coordinates:

$$x = r \cos \theta$$

$$y = r \sin \theta$$

$$r = \frac{ab}{\sqrt{a^2 \sin^2 \theta + b^2 \cos^2 \theta}}$$

$$= \frac{a}{\sqrt{\cos^2 \theta + \frac{a^2}{b^2} \sin^2 \theta}}$$

$$x = \frac{a}{\sqrt{\cos^2 \theta + \frac{a^2}{b^2} \sin^2 \theta}} \cos \theta$$

$$y = \frac{a}{\sqrt{\cos^2 \theta + \frac{a^2}{b^2} \sin^2 \theta}} \sin \theta$$

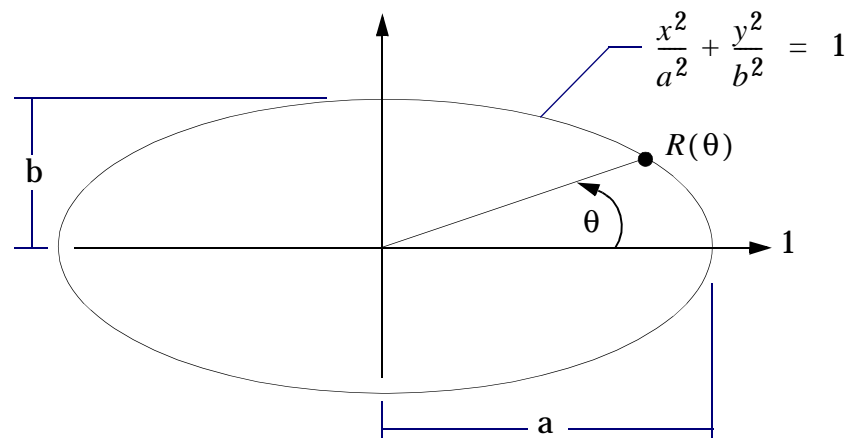


Figure 13-58

For the GMSURF entry, three different types of parameterization were used. In the first method,

$$x_s = \frac{v}{\sqrt{\cos^2 u + \frac{a^2}{b^2} \sin^2 u}} \cos u \quad 0.0 \leq u \leq \frac{\pi}{2}$$

$$y_s = \frac{v}{\sqrt{\cos^2 u + \frac{a^2}{b^2} \sin^2 u}} \sin u \quad 0.0 \leq v \leq a$$

as shown in **Figure 13-59**.

In the second method,

$$x_s = av \cos u \quad 0.0 \leq u \leq \frac{\pi}{2}$$

$$y_s = bv \sin u \quad 0.0 \leq v \leq 1.0$$

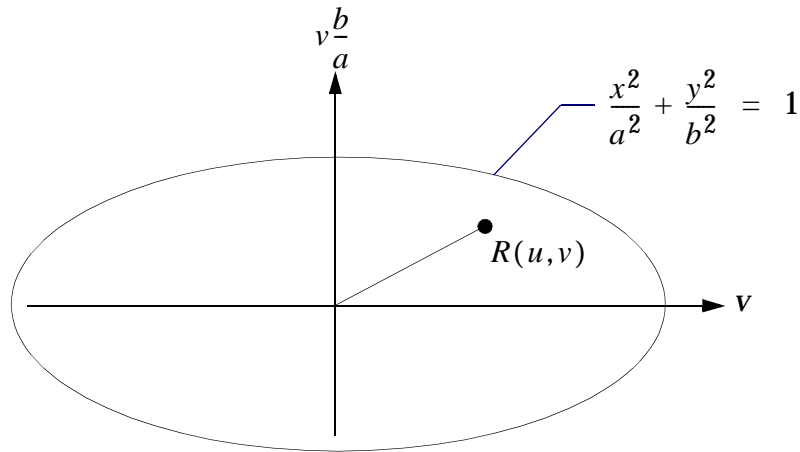


Figure 13-59

The third method, using hyperbolic functions,

$$x + iy = e(\cosh r \cos t + i \sinh r \sin t)$$

$$= e \cosh(r + it)$$

$$x = e \cosh r \cos t \quad 0.0 < r < R$$

$$y = e \sinh r \sin t \quad 0.0 < t < \frac{\pi}{2}$$

$$e = \sqrt{a^2 - b^2} \quad R = \tanh^{-1}\left(\frac{b}{a}\right)$$

as shown in **Figure 13-60**, with contours of t displayed.

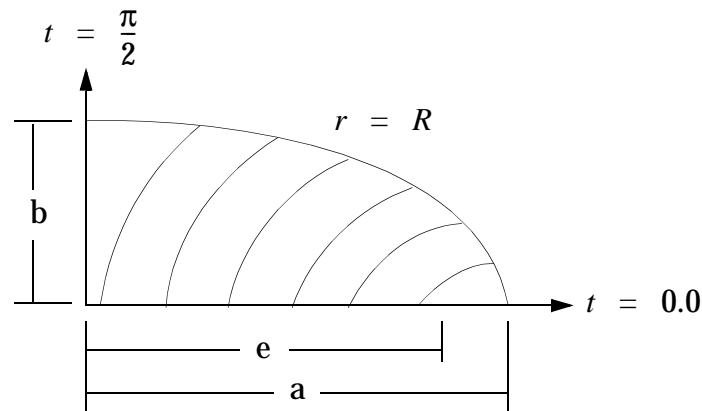


Figure 13-60

For the GMSURF entry, the order of the input IDs of DEQATN entries for the surface is important. First, we input the IDs of DEQATN entries providing equations for the x , y , and z coordinates of the surface. Then, the IDs of DEQATN entries of the first derivatives of x , y , and z functions with respect to the surface parameter u . Then, the IDs of DEQATN entries of the first derivatives of x , y , and z functions with respect to the surface parameter v . Then, the IDs of DEQATN entries of the second derivatives of x , y , and z functions with respect to the surface parameter u . Then, the IDs of DEQATN entries of the second derivatives of x , y , and z functions with respect to the surface parameter v . Finally, the IDs of DEQATN of the mixed second derivatives of the x , y , and z functions with respect to both surface parameters u and v .

The torsion in this example is defined by applying an edge load (GMLOAD entry) using the convective coordinate system (GMCORD entry). The load is applied tangent to the GMCURV 3. This is illustrated in [Figure 13-61](#). Note that an edge load on solid elements can cause singularities, but it is used here for demonstration purposes.

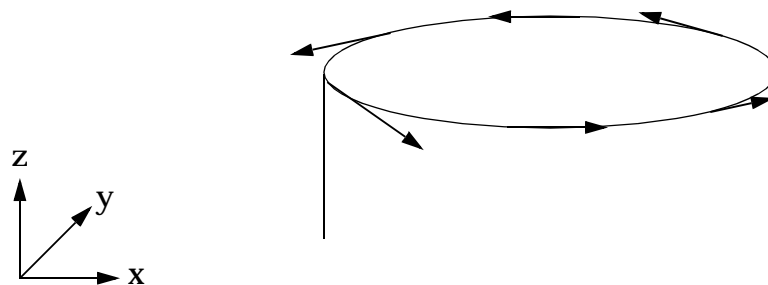


Figure 13-61

The input file of this model is listed in [Listing 13-16](#).

Listing 13-16 Input File for the Model

```

ID TORSION, STATIC
TIME 500
SOL 101 $ STATIC ANALYSIS
CEND
TITLE = TORSION, ELLIPSE CSA
SUBTITLE = METHOD 2
$
$ DEFINE ANALYSIS ENTRY
$
ADAPT = 100
$
LOAD = 1
SPC = 1
$
$ DEFINE OUTPUT REQUEST OPTION
$
DATAREC = 100
OUTRCV = 200
VUGRID (PUNCH) = ALL
$
$ DEFINE THE SETS OF ELEMENTS REQUIRED FOR THE ADAPTIVITY ANALYSIS
$ THIS IS REQUIRED WHEN SETTYP=SET IN PVAL ENTRY
$ (REFER TO PVAL ENTRY ALSO)
$
SETS DEFINITION
SET 120 = ALL
$
BEGIN BULK
PARAM  VUPENTA CPENTA
$
$ BOTH THE ADAPT AND PVAL ENTRY ARE USED TO
$ DEFINE ADAPTIVITY OPTION ENTRY FOR ALL ELEMENTS
$
$ FOR THE ADAPT ENTRY:
$ ID      = ADAPT ENTRY ID SELECTED IN CASE CONTROL
$ ADGEN   = ID OF THE FIRST PVAL ENTRY GENERATED IN THE ADAPTIVE PROCESS;
$         MUST BE LARGER THAN PSTRID, PMINID, AND PMAXID
$ MAXITER = NUMBER OF ITERATIONS REQUESTED BEFORE THE ADAPTIVE PROCESS
STOPPED
$ PSTRTID = ID OF THE STARTING PVAL ENTRY
$ PMINID  = ID OF THE MINIMUM PVAL ENTRY
$ PMAXID  = ID OF THE MAXIMUM PVAL ENTRY
$ PART    = ANY NAME CHOSEN FOR THE ELEMENTS;
$         USED FOR ERROR ESTIMATOR REFERENCE
$ ELSETID = ID OF SETS OF ELEMENTS REQUESTED FOR ADAPTIVE PROCESS
$ TYPE    = EBEP : THE P-ORDER IS INCREASED ELEMENTS-BY-ELEMENT
$ ERREST  = ERROR ESTIMATOR METHOD USED FOR ADAPTIVE PROCESS
$ ERRTOL  = ERROR TOLERANCE (IN FRACTION), REQUIRED IF MAXITER IS NOT
DEFINED
$ SIGTOL  = STRESS TOLERANCE
$
$          SID      ADGEN  MAXITER PSTRID  PMINID  PMAXID

```



```

ADAPT 100      1001    10      10      20      30                                +000001
++000001PART=ALL,ELSET=120,TYPE=EBEP,ERREST=1,ERRTOL=0.02,SIGTOL=5.0
+000002
$
$ DEFINE THE POLYNOMIAL ORDER DISTRIBUTION:
$ PVAL ID = ID OF PVAL DEFINED IN THE ADAPT ENTRY ABOVE
$ PX      = THE POLYNOMIAL ORDER DISTRIBUTION IN X-DIRECTION
$ PY      = THE POLYNOMIAL ORDER DISTRIBUTION IN Y-DIRECTION
$ PZ      = THE POLYNOMIAL ORDER DISTRIBUTION IN Z-DIRECTION
$ SETTYP  = TYPE OF SET PROVIDED. IT CAN BE EITHER SET OR ELID
$ ID      = SET ID OR ELEMENT ID, WHICHEVER IS APPROPRIATE
$
$
$      ID      Px      Py      Pz      SETTYP  ID
PVAL  10      2      2      2      SET      120
PVAL  20      2      2      2      SET      120
PVAL  30      4      4      4      SET      120
$
$ DEFINE THE OUTPUT REQUEST OPTION
$ REQUEST DISPLACEMENT AND STRESS OUTPUT FOR SET 120 FOR THE LAST
$ ADAPTIVITY ANALYSIS
$
OUTPUT 100                                +000003
++000003ELSET=120,DISP=PRINT,STRESS=PRINT,STRAIN=NONE
+000004
++000004FIRST=NO, LAST=YES, BY=2
+000005
$
$ THE OUTPUT COORDINATE SYSTEM IS THE BASIC COORDINATE SYSTEM (CID=0)
$
OUTRCV 200      120                        +000003
++000003CID=0,VIEW=2*2*2
+000004
$
$ DEFINE THE CYLINDRICAL COORDINATE SYSTEM FOR
$ GRID POINTS
$
CORD2C 1      0      0.0      0.0      0.0      0.0      0.0      1.0
+000005
++0000051.0      0.0      0.0                                +000006
$
$ DEFINE THE CONVECTIVE COORDINATE SYSTEM FOR THE APPLIED LOAD
$ IF GMCURV IS USED AS THE ENTITY TO DEFINE THE COORDINATE SYSTEM,
$ TWO ID'S ARE REQUIRED. THE FIRST ID IS THE GMCURV ID AND THE SECOND
$ ONE IS THE CORRESPONDING GMSURF ID
$
$      CID      ENTITY  ID 1      ID 2
GMCORD 10      GMCURV  3      2
$
$ DEFINE PENTA ELEMENTS
$
CPENTA 1      1      12      1      2      11      4      5
CPENTA 2      1      12      2      3      11      5      6
CPENTA 3      1      11      4      5      10      7      8
CPENTA 4      1      11      5      6      10      8      9

```

```

$
$ DEFINE GRID POINTS
$
GRID   1      1      4.      0.0     0.0
GRID   2      1      2.52982245.  0.
GRID   3      1      2.0      90.0     0.0
GRID   4      1      4.0      0.0     5.0
GRID   5      1      2.52982245.0  5.0
GRID   6      1      2.0      90.0     5.0
GRID   7      1      4.0      0.0    10.0
GRID   8      1      2.52982245.0  10.0
GRID   9      1      2.0      90.0    10.0
GRID  10      0.0      0.0    10.0
GRID  11      0.0      0.0     5.0
GRID  12      0.0      0.0     0.0
$
$ DEFINE FEEDGES
$ FEEDGES BY DEFAULT ARE STRAIGHT LINES
$ GMCURVS ARE REQUIRED FOR FEEDGES THAT DEFINE CURVED EDGES
$
$ FEEDGE 18 AND FEEDGE 19 MAKE GMCURV 1
$ FEEDGE 20 AND FEEDGE 21 MAKE GMCURV 2
$ FEEDGE 22 AND FEEDGE 23 MAKE GMCURV 3
$
$
$      ID      GRID 1  GRID 2  CID      GEOMIN  ID
FEEDGE 1      12      1      0
FEEDGE 2      12      2      0
FEEDGE 3      12      3      0
FEEDGE 4      11      4      0
FEEDGE 5      11      5      0
FEEDGE 6      11      6      0
FEEDGE 7      10      7      0
FEEDGE 8      10      8      0
FEEDGE 9      10      9      0
FEEDGE 10     12      11     0
FEEDGE 11     1      4      0
FEEDGE 12     2      5      0
FEEDGE 13     3      6      0
FEEDGE 14     11     10     0
FEEDGE 15     4      7      0
FEEDGE 16     5      8      0
FEEDGE 17     6      9      0
FEEDGE 18     1      2      0      GMCURV  1
FEEDGE 19     2      3      0      GMCURV  1
FEEDGE 20     4      5      0      GMCURV  2
FEEDGE 21     5      6      0      GMCURV  2
FEEDGE 22     7      8      0      GMCURV  3
FEEDGE 23     8      9      0      GMCURV  3
$
$ DEFINE CURVE GEOMETRY USING EQUATION METHOD
$
DEQATN 1      X(U) = 4.0*COS(U)/(SQRT(1.0+3.0*SIN(U)*SIN(U)))
DEQATN 2      Y(U) = 4.0*SIN(U)/(SQRT(1.0+3.0*SIN(U)*SIN(U)))
DEQATN 3      Z(U) = 0.0

```

```

DEQATN 4      Z1(U) = 5.0
DEQATN 5      Z2(U) = 10.0
DEQATN 6      XP(U) = -16.0*SIN(U)/((1.0+3.0*SIN(U)*SIN(U))**1.5)
DEQATN 7      YP(U) = 4.0*COS(U)/((1.0+3.0*SIN(U)*SIN(U))**1.5)
DEQATN 8      ZP(U) = 0.0
DEQATN 9      XPP(U) = (-16.0*COS(U)+96.0*SIN(U)*SIN(U)*COS(U))/
+000009
++000009      (1.0+3.0*SIN(U)*SIN(U))**2.5
+000010
DEQATN 10     YPP(U) = (-4.0*SIN(U)*(10.0-6.0*SIN(U)*SIN(U)))/
+000011
++000011      ((1.0+3.0*SIN(U)*SIN(U))**2.5)
+000012
DEQATN 11     ZPP(U) =0.0
$
$
GMCURV 1      MSCGRP1 0      0      +000013
++000013EQUATION,0.0,1.5708,1,2,3,6,7,
+000014
++0000148      9      10      11      +000015
GMCURV 2      MSCGRP1 0      0      +000016
++000016EQUATION,0.0,1.5708,1,2,4,6,7,
+000017
++0000178      9      10      11      +000018
GMCURV 3      MSCGRP1 0      0      +000019
++000019EQUATION,0.0,1.5708,1,2,5,6,7,
+000020
++0000208      9      10      11      +000021
$
$ DEFINE FEACES
$
$      ID      GRID 1  GRID 2  GRID 3  GRID 4  CID      SURFID
FEFACE 1      12      1      2      0      0      1
FEFACE 2      12      2      3      0      0      1
FEFACE 3      11      4      5      0      0
FEFACE 4      11      5      6      0      0
FEFACE 5      10      7      8      0      0      2
FEFACE 6      10      8      9      0      0      2
FEFACE 7      12      1      4      11      0
FEFACE 8      12      3      6      11      0
FEFACE 9      11      4      7      10      0
FEFACE 10     11      6      9      10      0
$
$ DEFINE SURFACE GEOMETRY USING EQUATION METHOD
$
DEQATN 13     X(U,V) = 4.0*V*COS(U)
DEQATN 14     Y(U,V) = 2.0*V*SIN(U)
DEQATN 15     Z1(U,V) = 0.0
DEQATN 16     Z2(U,V) = 10.0
DEQATN 17     XU(U,V) = -4.0*V*SIN(U)
DEQATN 18     YU(U,V) = 2.0*V*COS(U)
DEQATN 19     ZU(U,V) = 0.0
DEQATN 20     XV(U,V) = 4.0*COS(U)
DEQATN 21     YV(U,V) = 2.0*SIN(U)

```

```

DEQATN 22      ZV(U,V) = 0.0
DEQATN 23      XU(U,V)= -4.0*V*COS(U)
DEQATN 24      YU(U,V)= -2.0*V*SIN(U)
DEQATN 25      ZU(U,V)= 0.0
DEQATN 26      XV(U,V)= 0.0
DEQATN 27      YV(U,V)= 0.0
DEQATN 28      ZV(U,V)= 0.0
DEQATN 29      XU(U,V)= -4.0*SIN(U)
DEQATN 30      YU(U,V)= 2.0*COS(U)
DEQATN 31      ZU(U,V)= 0.0
$
$
GMSURF 1      MSCGRP1 0      0      +000022
++000022EQUATION,0.0,1.5708,0.0,1.0,13,14,15,
+000023
++00002317,18,19,20,21,22,23,24,
+000024
++00002425,26,27,28,29,30,31
+000025
GMSURF 2      MSCGRP1 0      0      +000026
++000026EQUATION,0.0,1.5708,0.0,1.0,13,14,16,
+000027
++00002717,18,19,20,21,22,23,24,
+000028
++00002825,26,27,28,29,30,31
+000029
$
$ DEFINE CONSTRAINT IN THE X-DIRECTION
$
$      SID      C      ENTITY  ID
GMSPC  1      1      FEFACE  1
GMSPC  1      1      FEFACE  2
GMSPC  1      1      FEFACE  7
GMSPC  1      1      FEFACE  9
$
$ DEFINE CONSTRAINT IN THE Y-DIRECTION
$
$      SID      C      ENTITY  ID
GMSPC  1      2      FEFACE  1
GMSPC  1      2      FEFACE  2
GMSPC  1      2      FEFACE  8
GMSPC  1      2      FEFACE 10
$
$ DEFINE CONSTRAINT IN THE Z-DIRECTION
$
$      SID      C      ENTITY  ID
GMSPC  1      3      FEFACE  7
GMSPC  1      3      FEFACE  8
GMSPC  1      3      FEFACE  9
GMSPC  1      3      FEFACE 10
$
$ DEFINE THE MATERIAL PROPERTY (ALUMINUM)
$
MAT1    1      1.+7      0.3

```

```

PSOLID 1      1                      GRID      SMECH
$
$ CONSTANT LOAD OF 100 LB/INCH IS APPLIED TANGENTIALLY TO
$ GMCURV 3
$
$ LID      = LOAD IDENTIFICATION NUMBER DEFINED IN CASE CONTROL SECTION
$ CID      = COORDINATE SYSTEM USED TO DEFINE THE APPLIED LOAD
$ N1,N2,N3 = DIRECTION OF THE LOAD VECTOR
$ ENTITY   = ENTITY AT WHICH LOAD IS APPLIED TO
$ ID       = ID OF THE ENTITY
$ METHOD    = METHOD USED TO DEFINE THE LOAD
$
$          LID      CID      N1      N2      N3      ENTITY ID      METHOD
GMLOAD 1      10      1.0      0.0      0.0      GMCURV 3      CONSTANT
+000007
$
$          VALUE OF LOAD DENSITY
++000007100.0                      +000008
$
ENDDATA

```

Results

The OUTPUT entry requests the displacements and stress output solely for the last adaptivity iteration. An abridged output for this model is shown in [Figure 13-62](#).

```

1  TORSION, ELLIPSE CSA                                DECEMBER 22, 1993 MSC/NASTRAN 12/20/93 PAGE 73
METHOD 2
0*** USER INFORMATION MESSAGE 5293 FOR DATA BLOCK KLL
LOAD SEQ. NO.      EPSILON      EXTERNAL WORK      EPSILONS LARGER THAN .001 ARE FLAGGED WITH ASTERISKS
1                  5.2309875E-15      1.6239751E-02
*** USER INFORMATION MESSAGE 6502 (ADAPT)
THE CURRENT SUPERELEMENT REQUIRES NO FURTHER P-ADAPTIVE ANALYSIS BECAUSE THERE ARE NO CHANGES IN P-VALUES.
0*** USER WARNING MESSAGE 4594. THE POSTCDB CONTAINS NO LEGITIMATE SURFACE OR VOLUME DEFINITIONS.
1  TORSION, ELLIPSE CSA                                DECEMBER 22, 1993 MSC/NASTRAN 12/20/93 PAGE 74
METHOD 2                                                ADAPTIVITY INDEX= 3
0

```

DISPLACEMENT VECTOR							
POINT ID.	TYPE	T1	T2	T3	R1	R2	R3
1	G	.0	.0	.0	.0	.0	.0
2	G	.0	.0	-8.692895E-07	.0	.0	.0
3	G	.0	.0	.0	.0	.0	.0
4	G	.0	1.357096E-05	.0	.0	.0	.0
5	G	-3.062882E-06	-2.173771E-06	-1.839771E-06	.0	.0	.0
6	G	1.593498E-05	.0	.0	.0	.0	.0
7	G	.0	1.513064E-04	.0	.0	.0	.0
8	G	1.887783E-06	-4.328616E-05	1.990558E-05	.0	.0	.0
9	G	1.121671E-04	.0	.0	.0	.0	.0
10	G	.0	.0	.0	.0	.0	.0
11	G	.0	.0	.0	.0	.0	.0
12	G	.0	.0	.0	.0	.0	.0
201001001	G	.0	.0	-6.008400E-23	.0	.0	.0
201001002	G	.0	.0	-3.919389E-23	.0	.0	.0
201001003	G	.0	.0	-1.764301E-22	.0	.0	.0
201001004	G	.0	.0	-4.865105E-07	.0	.0	.0
201001005	G	.0	.0	-8.088551E-07	.0	.0	.0
201001006	G	.0	.0	-8.692895E-07	.0	.0	.0
201001007	G	-3.961429E-23	1.151926E-22	6.248783E-24	.0	.0	.0
201001008	G	-1.317319E-22	2.320618E-06	-1.101103E-22	.0	.0	.0
201001009	G	-1.197587E-22	3.720583E-06	-2.934319E-24	.0	.0	.0
201001010	G	-7.882488E-07	1.256443E-06	-2.743236E-07	.0	.0	.0
201001011	G	-1.074418E-06	2.716110E-06	-1.095060E-06	.0	.0	.0
201001012	G	-1.967381E-06	1.468333E-06	-7.467912E-07	.0	.0	.0
201001013	G	-9.475001E-23	4.980112E-22	-3.150599E-23	.0	.0	.0
201001014	G	-3.235471E-22	3.547284E-06	-1.786648E-22	.0	.0	.0
201001015	G	-4.842875E-22	1.357096E-05	6.322074E-23	.0	.0	.0
201001016	G	-8.565316E-07	4.229675E-06	-7.164187E-07	.0	.0	.0
201001017	G	-3.000400E-06	4.619390E-06	-5.842737E-07	.0	.0	.0
201001018	G	-3.062882E-06	-2.173771E-06	-1.839771E-06	.0	.0	.0
201002001	G	.0	.0	3.004200E-23	.0	.0	.0
201002002	G	.0	.0	-4.865105E-07	.0	.0	.0
201002003	G	.0	.0	-8.692895E-07	.0	.0	.0
201002004	G	.0	.0	2.921973E-23	.0	.0	.0
201002005	G	.0	.0	5.445964E-07	.0	.0	.0
201002006	G	.0	.0	4.219457E-22	.0	.0	.0
201002007	G	-1.446019E-22	-5.887475E-23	-3.124391E-24	.0	.0	.0
201002008	G	-7.882488E-07	1.256443E-06	-2.743236E-07	.0	.0	.0
201002009	G	-1.967381E-06	1.468333E-06	-7.467912E-07	.0	.0	.0
201002010	G	-8.927043E-07	3.961755E-23	-6.052314E-24	.0	.0	.0
201002011	G	-2.660616E-06	7.911209E-07	5.446515E-08	.0	.0	.0
201002012	G	-7.240385E-08	1.330381E-22	-1.194728E-22	.0	.0	.0
201002013	G	-8.723581E-22	-3.184499E-22	1.575299E-23	.0	.0	.0
201002014	G	-8.565316E-07	4.229675E-06	-7.164187E-07	.0	.0	.0
201002015	G	-3.062882E-06	-2.173771E-06	-1.839771E-06	.0	.0	.0
201002016	G	-1.335821E-06	1.364713E-22	2.932544E-23	.0	.0	.0
201002017	G	-7.250181E-06	1.251031E-06	1.185184E-06	.0	.0	.0
201002018	G	1.593498E-05	2.834062E-22	4.533070E-22	.0	.0	.0
201003001	G	-9.475001E-23	4.980112E-22	-3.150599E-23	.0	.0	.0
201003002	G	-3.235471E-22	3.547284E-06	-1.786648E-22	.0	.0	.0

Figure 13-62 Abridged Output for the Model

```

1  TORSION, ELLIPSE CSA                                DECEMBER 22, 1993 MSC/NASTRAN 12/20/93 PAGE 77
METHOD 2                                               ADAPTIVITY INDEX= 3
0  PVAL ID= 1002 (P1,P2,P3= 3 2 3), SUBCASE 1
STRESSES IN P - VERSION PENTAHEDRON SOLID ELEMENTS (PENTA)
VIEW-ELEMENT ID=100001001, P-ELEMENT ID = 1, OUTPUT COORD. ID= 0, P OF EDGES = 3 3 4 4 3 4 4 4 4
VUGRID SIGMA-X/1 SIGMA-Y/2 SIGMA-Z/3 TAU-XY TAU-YZ TAU-XZ MEAN-PRESSURE VON-MISES
201001001 2.507561E-15 2.507561E-15 5.850975E-15 -2.450845E-31 -3.490772E+00 3.045360E-16
3.490772E+00 -3.490772E+00 2.881587E-15
201001002 -9.512720E-17 -6.704689E-16 3.533783E-16 7.719806E-32 7.080933E-01 -7.649862E-16
7.080933E-01 -7.080933E-01 -2.876032E-16 2.068900E-16 1.226454E+00
201001004 5.034403E+00 5.034403E+00 1.174694E+01 4.981801E-15 -1.109684E+01 -4.317494E+00
2.076182E+01 -3.980472E+00 5.034403E+00 -7.271916E+00 2.168871E+01
201001007 1.006225E+01 2.347857E+01 1.006225E+01 -2.168649E+00 3.630430E-01 -1.325106E-16
2.382975E+01 9.711069E+00 1.006225E+01 -1.453436E+01 1.394641E+01
201001008 -3.267787E+00 -7.624836E+00 -3.267787E+00 -1.073696E+00 -6.464741E-01 -1.551473E-15
-2.933002E+00 -7.959620E+00 -3.267787E+00 4.720137E+00 4.867867E+00
201001010 -1.527811E+00 -2.566048E+00 -9.243130E-01 -5.258331E-02 -3.754325E-01 -2.206367E+00
1.018136E+00 -3.538538E+00 -2.497769E+00 1.672724E+00 4.135702E+00
STRESSES IN P - VERSION PENTAHEDRON SOLID ELEMENTS (PENTA)
VIEW-ELEMENT ID=100001002, P-ELEMENT ID = 1, OUTPUT COORD. ID= 0, P OF EDGES = 3 3 4 4 3 4 4 4 4
VUGRID SIGMA-X/1 SIGMA-Y/2 SIGMA-Z/3 TAU-XY TAU-YZ TAU-XZ MEAN-PRESSURE VON-MISES
201001002 -9.512720E-17 -6.704689E-16 3.533783E-16 7.719806E-32 7.080933E-01 -7.649862E-16
7.080933E-01 -7.080933E-01 -2.876032E-16 2.068900E-16 1.226454E+00
201001003 2.474018E-14 5.246847E-14 2.999879E-14 -1.088513E-15 2.914146E+00 -2.396744E-15
2.914146E+00 -2.914146E+00 2.316292E-14 -3.525450E-14 5.047449E+00
201001005 -7.025239E-01 -7.025239E-01 -1.639222E+00 3.874331E-15 1.205412E+01 -6.993091E+00
1.277274E+01 -1.511449E+01 -7.025239E-01 1.014757E+00 2.415559E+01
201001008 -3.267787E+00 -7.624836E+00 -3.267787E+00 -1.073696E+00 -6.464741E-01 -1.551473E-15
-2.933002E+00 -7.959620E+00 -3.267787E+00 4.720137E+00 4.867867E+00
201001009 6.025518E+00 1.405954E+01 6.025518E+00 7.571502E-01 6.719486E+00 7.331966E-16
1.790772E+01 2.177344E+00 6.025518E+00 -8.703526E+00 1.420281E+01
201001011 2.021004E+00 -5.901449E+00 -1.849734E+00 2.802238E+00 2.882026E+00 -1.129690E+00
2.922884E+00 -8.277588E+00 -3.754748E-01 1.910060E+00 9.969281E+00
1  TORSION, ELLIPSE CSA                                DECEMBER 22, 1993 MSC/NASTRAN 12/20/93 PAGE 78
METHOD 2                                               ADAPTIVITY INDEX= 3
0  PVAL ID= 1002 (P1,P2,P3= 3 2 3), SUBCASE 1
STRESSES IN P - VERSION PENTAHEDRON SOLID ELEMENTS (PENTA)
VIEW-ELEMENT ID=100001003, P-ELEMENT ID = 1, OUTPUT COORD. ID= 0, P OF EDGES = 3 3 4 4 3 4 4 4 4
VUGRID SIGMA-X/1 SIGMA-Y/2 SIGMA-Z/3 TAU-XY TAU-YZ TAU-XZ MEAN-PRESSURE VON-MISES
201001005 -7.025239E-01 -7.025239E-01 -1.639222E+00 3.874331E-15 1.205412E+01 -6.993091E+00
1.277274E+01 -1.511449E+01 -7.025239E-01 1.014757E+00 2.415559E+01
201001004 5.034403E+00 5.034403E+00 1.174694E+01 4.981801E-15 -1.109684E+01 -4.317494E+00
2.076182E+01 -3.980472E+00 5.034403E+00 -7.271916E+00 2.168871E+01
201001002 -9.512720E-17 -6.704689E-16 3.533783E-16 7.719806E-32 7.080933E-01 -7.649862E-16
7.080933E-01 -7.080933E-01 -2.876032E-16 2.068900E-16 1.226454E+00
201001011 2.021004E+00 -5.901449E+00 -1.849734E+00 2.802238E+00 2.882026E+00 -1.129690E+00
2.922884E+00 -8.277588E+00 -3.754748E-01 1.910060E+00 9.969281E+00
201001010 -1.527811E+00 -2.566048E+00 -9.243130E-01 -5.258331E-02 -3.754325E-01 -2.206367E+00
1.018136E+00 -3.538538E+00 -2.497769E+00 1.672724E+00 4.135702E+00
201001008 -3.267787E+00 -7.624836E+00 -3.267787E+00 -1.073696E+00 -6.464741E-01 -1.551473E-15
-2.933002E+00 -7.959620E+00 -3.267787E+00 4.720137E+00 4.867867E+00
STRESSES IN P - VERSION PENTAHEDRON SOLID ELEMENTS (PENTA)
VIEW-ELEMENT ID=100001004, P-ELEMENT ID = 1, OUTPUT COORD. ID= 0, P OF EDGES = 3 3 4 4 3 4 4 4 4
VUGRID SIGMA-X/1 SIGMA-Y/2 SIGMA-Z/3 TAU-XY TAU-YZ TAU-XZ MEAN-PRESSURE VON-MISES
201001004 5.034403E+00 5.034403E+00 1.174694E+01 4.981801E-15 -1.109684E+01 -4.317494E+00
2.076182E+01 -3.980472E+00 5.034403E+00 -7.271916E+00 2.168871E+01
201001005 -7.025239E-01 -7.025239E-01 -1.639222E+00 3.874331E-15 1.205412E+01 -6.993091E+00
1.277274E+01 -1.511449E+01 -7.025239E-01 1.014757E+00 2.415559E+01
201001006 -3.200166E-02 -3.200166E-02 -7.467054E-02 -8.733983E-14 1.493832E+01 -4.544772E+00
1.556104E+01 -1.566771E+01 -3.200166E-02 4.622462E-02 2.704490E+01
201001010 -1.527811E+00 -2.566048E+00 -9.243130E-01 -5.258331E-02 -3.754325E-01 -2.206367E+00
1.018136E+00 -3.538538E+00 -2.497769E+00 1.672724E+00 4.135702E+00
201001011 2.021004E+00 -5.901449E+00 -1.849734E+00 2.802238E+00 2.882026E+00 -1.129690E+00
2.922884E+00 -8.277588E+00 -3.754748E-01 1.910060E+00 9.969281E+00
201001012 2.319424E+00 -3.111391E+00 -1.438776E+00 -7.330344E+00 8.622318E+00 -9.914967E+00
1.681246E+01 -1.124202E+01 -7.801181E+00 7.435812E-01 2.650212E+01

```

Figure 13-62 Abridged Output for the Model (continued)

13.8.8 Circular Solid Model

Objectives and Problem Description

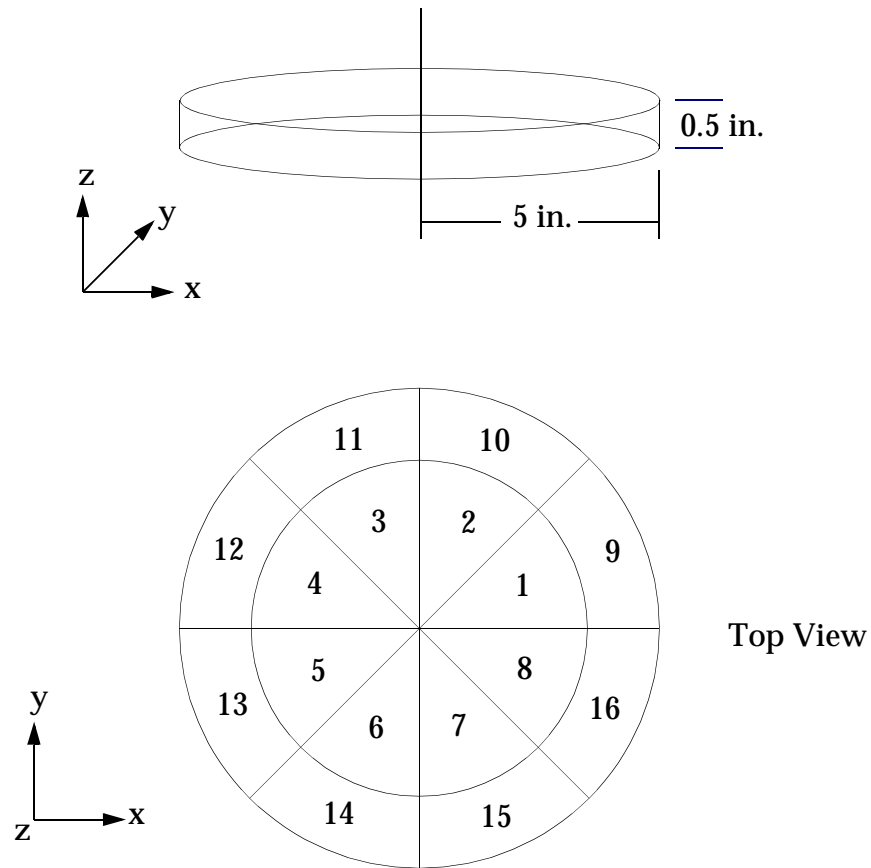
This example is a circular solid with a simply supported edge. It is made of aluminum and has the following properties:

$$E = 1.0E7 \text{ lb/in}^2$$

$$\rho_w = 0.098 \text{ lbf/in}^3$$

$$\nu = 0.3$$

It is illustrated in **Figure 13-63**.



element 1 thru 8: CPENTA elements
 element 9 thru 16: CHEXA elements (boundary elements)

Figure 13-63 The Circular Solid Model

Theoretical Results

The behavior of this circular solid model can be compared to the theoretical circular plate model described in Reference 6. From Reference 6.,

$$f = \frac{K_n}{2\pi} \sqrt{\frac{Dg}{wr^4}}$$

where:

$$D = \frac{Et^3}{12(1-\nu^2)}$$

w = uniform load per unit area including own weight

$$= \frac{(\rho_w)(Volume)}{Area}$$

$$= \frac{(\rho_w)(Area)(t)}{Area}$$

$$= (\rho_m)(g)(t)$$

$$f = \frac{K_n}{2\pi} \sqrt{\frac{Et^2}{12(1-\nu^2)(\rho_m)(r^4)}} \text{ cycles/sec}$$

where:

E = modulus of elasticity

K_n = constant, n refers to the mode of vibration

t = thickness of the plate

r = radius of the plate

ν = Poisson's ratio

For aluminum,

$$E = 1.0E7 \text{ lb/in}^2$$

$$\nu = 0.3$$

$$\rho_w = 0.098 \text{ lbf/in}^3$$

$$\rho_m = 2.538E-4 \text{ lbfm/in}^3$$

For the circular plate, radius (r) = 5 inches and thickness (t) = 0.5 inches. The theoretical natural frequencies for all modes:

$$f = \frac{K_n}{2\pi} \sqrt{\frac{(10^7)(0.5^2)}{12(1-0.3^2)(2.538 \times 10^{-4})(5^4)}} = 191.2 K_n \text{ cycles/sec}$$

The theoretical natural frequencies for the first several modes are calculated and tabulated in **Table 13-24**.

Table 13-24 Theoretical Natural Frequency Values for Circular Plate

Mode	K_n	Natural Frequency (cycles/sec)
Fundamental	4.99	954.1
One Nodal Diameter	13.9	2657.7
Two Nodal Diameters	25.7	4913.9
One Nodal Circle	29.8	5697.8

Setup

The model is constructed from 8 CPENTA elements and 8 CHEXA elements. The CHEXA elements are the boundary elements. The circular edge of the model is described using the circle equation (GMCURV entry, method=equation). Since the GMCURV can not be a closed curve, the full circle must be described with multiple GMCURV entries. In this model, one full circle is defined by two GMCURV entries. The first GMCURV entry has an input range from 0.0 to 3.1416 radians, and the second GMCURV entry has an input range from 3.1415 to 6.2832 radians. Note that there is some overlap, so that both segments include π . In addition, since the circle equation is a straightforward equation, the first and the second derivatives of the equation are not necessarily required. In this case, the derivatives are then computed numerically.

For the boundary conditions, GMSPC entries are used. Due to the hierarchy order which places grids as the highest priority, GMSPC entries that are imposed on grid 4 and 8 in the x-direction will override GMSPC entries on the adjacent FEFACES in the z-direction. Thus, it is necessary to add two more GMSPC entries for grids 4 and 8 in the z-direction to restore the constraints in the z-direction.

For the material property, the MAT1 entry is used. In the MAT1 entry, the weight density term ρ_w must be converted to the mass density ρ_m for consistency of units. Therefore,

$$\begin{aligned}\rho_m &= (0.098 \text{ lbf/in}^3)(1 / 32.174 \text{ ft/s}^2)(1 / 12 \text{ in/ft}) \\ &= 2.538\text{E-}4 \text{ lbf/in}^3\end{aligned}$$

The input file for this model is shown in [Listing 13-17](#).

Listing 13-17 Input File for the Circular Solid Model

```

ID NORMAL, MODES
TIME 200
SOL 103 $ NORMAL MODE ANALYSIS
CEND
TITLE = CIRCULAR PLATE
SUBTITLE = EDGE SIMPLY SUPPORTED
METHOD = 100
$
$ DEFINE ADAPTIVITY ANALYSIS ENTRY
$
ADAPT = 1
$
$ OUTPUT REQUEST
$
DISPLACEMENT = ALL
ECHO = BOTH
$
SPC =1
$
$ REQUEST FOR VIEW COORDINATES OF THE P-ELEMENTS TO BE
$ PRINTED IN THE PUNCH FILE. THE VIEW COORDINATES
$ ARE THE POINTS AT WHICH THE DISPLACEMENTS ARE
$ CALCULATED.
$
VUGRID(PUNCH) = ALL
$
$ DEFINE THE SETS OF ELEMENTS REQUIRED FOR THE ADAPTIVITY
$ ANALYSIS. THIS IS REQUIRED WHEN SETTYP=SET IN PVAL ENTRY
$ ( REFER TO PVAL ENTRY ALSO )
$
SETS DEFINITION
SET 10 = ALL
$
BEGIN BULK
PARAM   POST      0
$
PARAM   VUHEXA   CHEXA
PARAM   VUPENTA  CPENTA
$
$ BOTH THE ADAPT AND PVAL ENTRY ARE USED TO
$ DEFINE ADAPTIVITY OPTION ENTRY FOR ALL ELEMENTS,
$
$ FOR THE ADAPT ENTRY:
$ ID      = ADAPT ENTRY ID SELECTED IN CASE CONTROL
$ ADGEN   = ID OF THE FIRST PVAL ENTRY GENERATED IN THE ADAPTIVE PROCESS;
$          MUST BE LARGER THAN PSTRID, PMINID, AND PMAXID
$ MAXITER = NUMBER OF ITERATIONS REQUESTED BEFORE THE ADAPTIVE PROCESS STOPPED
$ PSTRTID = ID OF THE STARTING PVAL ENTRY
$ PMINID  = ID OF THE MINIMUM PVAL ENTRY
$ PMAXID  = ID OF THE MAXIMUM PVAL ENTRY
$ PART    = ANY NAME CHOSEN FOR THE ELEMENTS;
$          USED FOR ERROR ESTIMATOR REFERENCE
$ ELSETID = ID OF SETS OF ELEMENTS REQUESTED FOR ADAPTIVE PROCESS
$ TYPE    = EBEP : THE P-ORDER WILL INCREASE ONLY IN THE ELEMENTS WHOSE ERROR
$          IS GREATER THAN ERROR TOLERANCE.
$ ERREST  = ERROR ESTIMATOR METHOD USED FOR ADAPTIVE PROCESS

```

```

$ ERRTOL = ERROR TOLERANCE (IN FRACTION), REQUIRED IF MAXITER IS NOT DEFINED
$ SIGTOL = STRESS TOLERANCE
$
$          ID          ADGEN    MAXITER PSTRITD PMINID  PMAXID
ADAPT    1            1001      5        10      20      30                +000001
$
++000001PART=ALL,ELSET=10,TYPE=EBEP,ERREST=1,ERRTOL=0.02,SIGTOL=0.0      +000002
$

```

Listing 13-17 Input File for the Circular Solid Model (continued)

```

$ DEFINE THE POLYNOMIAL ORDER DISTRIBUTION:
$ PVAL ID = ID OF PVAL DEFINED IN THE ADAPT ENTRY ABOVE
$ Px      = THE POLYNOMIAL ORDER DISTRIBUTION IN X-DIRECTION
$ Py      = THE POLYNOMIAL ORDER DISTRIBUTION IN Y-DIRECTION
$ Pz      = THE POLYNOMIAL ORDER DISTRIBUTION IN Z-DIRECTION
$ SETTYP  = TYPE OF SET PROVIDED. IT CAN BE EITHER SET or ELID
$ ID      = SET ID or ELEMENT ID WHICHEVER IS APPROPRIATE
$
$          ID          Px          Py          Pz          SETTYP  ID
PVAL    10            2            2            2            SET      10
PVAL    20            4            4            2            SET      10
PVAL    30            6            6            2            SET      10
$
$ DEFINE A CYLINDRICAL COORDINATE SYSTEM
$ USED IN DEFINING THE GRIDS
$
CORD2C  1            0            0.0        0.0        0.0        0.0        0.0        1.0        +000003
++0000031.0        0.0        0.0
$
$ DEFINE SOLID ELEMENTS
$ HEXA ELEMENTS ARE THE BOUNDARY ELEMENTS
$
CPENTA  1            1            1            21            22            10            29            30
CPENTA  2            1            1            22            23            10            30            31
CPENTA  3            1            1            23            24            10            31            32
CPENTA  4            1            1            24            25            10            32            33
CPENTA  5            1            1            25            26            10            33            34
CPENTA  6            1            1            26            27            10            34            35
CPENTA  7            1            1            27            28            10            35            36
CPENTA  8            1            1            28            21            10            36            29
CHEXA   9            1            2            3            22            21            11            12            +000005
++00000530        29                +000006
CHEXA  10            1            3            4            23            22            12            13            +000007
++00000731        30                +000008
CHEXA  11            1            4            5            24            23            13            14            +000009
++00000932        31                +000010
CHEXA  12            1            5            6            25            24            14            15            +000011
++00001133        32                +000012
CHEXA  13            1            6            7            26            25            15            16            +000013
++00001334        33                +000014
CHEXA  14            1            7            8            27            26            16            17            +000015
++00001535        34                +000016
CHEXA  15            1            8            9            28            27            17            18            +000017
++00001736        35                +000018
CHEXA  16            1            9            2            21            28            18            11            +000019
++00001929        36
$
$DEFINE EIGRL CARD (LANCZOS METHOD)

```

```

$
EIGRL 100 10
$
$DEFINE THE CURVE GEOMETRY, METHOD = EQUATION, U IS THE CURVE PARAMETER
$
$DEFINE THE OUTER BOTTOM CIRCLE OF RADIUS=5.0 AT Z=0.0 USING DEQATN 1,2,AND 3.
$THE OUTER BOTTOM CIRCLE IS DEFINED BY TWO GMCURV ENTRIES:
$GMCURV 1 WITH THE RANGE OF CURVE PARAMETER,U = 0.0 TO 3.1416 RADIANS
$GMCURV 2 WITH THE RANGE OF CURVE PARAMETER,U = 3.1415 TO 6.2832 RADIANS
$
DEQATN 1 X(U) = 5.0*COS(U)
DEQATN 2 Y(U) = 5.0*SIN(U)
DEQATN 3 Z(U) = 0.0
$
GMCURV 1 MSCGRP1 0 0 +000021
++000021EQUATION,0.0,3.1416,1,2,3 +000022

GMCURV 2 MSCGRP1 0 0 +000025
++000025EQUATION,3.1415,6.2832,1,2,3 +000026
$
$ DEFINE THE TOP OUTER CIRCLE USING THE SAME EQUATION, DEQATN 1 AND DEQATN 2,
$ BUT WITH Z=0.5 (DEQATN 4). THIS TOP OUTER CIRCLE IS DEFINED BY TWO GMCURV
$ ENTRIES:
$ GMCURV 3 WITH THE RANGE OF CURVE PARAMETER,U = 0.0 TO 3.1416 RADIANS
$ GMCURV 4 WITH THE RANGE OF CURVE PARAMETER,U = 3.1415 TO 6.2832 RADIANS
$
DEQATN 4 Z1(U) = 0.5
$
GMCURV 3 MSCGRP1 0 0 +000023
++000023EQUATION,0.0,3.1416,1,2,4 +000024
GMCURV 4 MSCGRP1 0 0 +000027
++000027EQUATION,3.1415,6.2832,1,2,4 +000028
$
$ DEFINE THE BOTTOM INNER CIRCLE OF RADIUS = 4.0 AT Z=0.0 USING DEQATN 5,6,
$ AND 7. THE BOTTOM INNER CIRCLE IS DEFINED BY TWO GMCURV ENTRIES:
$ GMCURV 5 WITH THE RANGE OF CURVE PARAMETER,U = 0.0 TO 3.1416 RADIANS
$ GMCURV 6 WITH THE RANGE OF CURVE PARAMETER U = 3.1415 TO 6.2832 RADIANS
$
DEQATN 5 X(U) = 4.0*COS(U)
DEQATN 6 Y(U) = 4.0*SIN(U)
DEQATN 7 Z(U) = 0.0
$
GMCURV 5 MSCGRP1 0 0 +000029
++000029EQUATION,0.0,3.1416,5,6,7 +000030
GMCURV 6 MSCGRP1 0 0 +000033
++000033EQUATION,3.1415,6.2832,5,6,7 +000034
$
$ DEFINE THE TOP INNER CIRCLE USING THE SAME EQUATION, DEQATN 5 AND DEQATN 6,
$ BUT WITH Z=0.5 (DEQATN 8). THIS TOP INNER CIRCLE IS DEFINED BY TWO GMCURV
$ ENTRIES:
$ GMCURV 7 WITH THE RANGE OF CURVE PARAMETER,U = 0.0 TO 3.1416 RADIANS
$ GMCURV 8 WITH THE RANGE OF CURVE PARAMETER U = 3.1415 TO 6.2832 RADIANS
$
DEQATN 8 Z1(U) = 0.5
$
GMCURV 7 MSCGRP1 0 0 +000031
++000031EQUATION,0.0,3.1416,5,6,8 +000032

```

```

GMCURV 8          MSCGRP1 0          0          +000035
++000035EQUATION,3.1415,6.2832,5,6,8      +000036
$
$ DEFINE FEEDGES
$ FEEDGE BY DEFAULT IS A STRAIGHT LINE.
$ GMCURVS ARE REQUIRED FOR FEEDGES THAT DEFINE CURVED EDGES
$
$ FEEDGES 1, 2, 3 AND 4 MAKE GMCURV 1
$ FEEDGES 5, 6, 7 AND 8 MAKE GMCURV 2
$ FEEDGES 9, 10, 11 AND 12 MAKE GMCURV 3
$ FEEDGES 13, 14, 15 AND 16 MAKE GMCURV 4
$ FEEDGES 17, 18, 19 AND 20 MAKE GMCURV 5
$ FEEDGES 21, 22, 23 AND 24 MAKE GMCURV 6
$ FEEDGES 25, 26, 27 AND 28 MAKE GMCURV 7
$ FEEDGES 29, 30, 31 AND 32 MAKE GMCURV 8
$
$          ID          GRID 1  GRID 2  CID          GEOMIN ID
FEEDGE 1          2          3          0          GMCURV 1
FEEDGE 2          3          4          0          GMCURV 1
FEEDGE 3          4          5          0          GMCURV 1
FEEDGE 4          5          6          0          GMCURV 1
FEEDGE 5          6          7          0          GMCURV 2
FEEDGE 6          7          8          0          GMCURV 2
FEEDGE 7          8          9          0          GMCURV 2
FEEDGE 8          9          2          0          GMCURV 2
FEEDGE 9          11         12         0          GMCURV 3
FEEDGE 10         12         13         0          GMCURV 3
FEEDGE 11         13         14         0          GMCURV 3
FEEDGE 12         14         15         0          GMCURV 3
FEEDGE 13         15         16         0          GMCURV 4
FEEDGE 14         16         17         0          GMCURV 4
FEEDGE 15         17         18         0          GMCURV 4
FEEDGE 16         18         11         0          GMCURV 4
FEEDGE 17         21         22         0          GMCURV 5
FEEDGE 18         22         23         0          GMCURV 5
FEEDGE 19         23         24         0          GMCURV 5
FEEDGE 20         24         25         0          GMCURV 5
FEEDGE 21         25         26         0          GMCURV 6
FEEDGE 22         26         27         0          GMCURV 6
FEEDGE 23         27         28         0          GMCURV 6
FEEDGE 24         28         21         0          GMCURV 6
FEEDGE 25         29         30         0          GMCURV 7
FEEDGE 26         30         31         0          GMCURV 7
FEEDGE 27         31         32         0          GMCURV 7
FEEDGE 28         32         33         0          GMCURV 7
FEEDGE 29         33         34         0          GMCURV 8
FEEDGE 30         34         35         0          GMCURV 8
FEEDGE 31         35         36         0          GMCURV 8
FEEDGE 32         36         29         0          GMCURV 8
$
$ DEFINE FEFACES FOR BOUNDARY CONDITION
$
$          ID          GRID 1  GRID 2  GRID 3  GRID 4  CID
FEFACE 1          2          3          12         11         0
FEFACE .2         3          4          13         12         0
FEFACE 3          4          5          14         13         0
FEFACE 4          5          6          15         14         0
FEFACE 5          6          7          16         15         0
FEFACE 6          7          8          17         16         0

```

```

FEFACE  7      8      9      18      17      0
FEFACE  8      9      2      11      18      0
$
$ DEFINE CONSTRAINTS IN THE Z-DIRECTION
$
$ SID      = SPC ID THAT IS DEFINED IN CASE CONTROL
$ C        = CONSTRAINT COMPONENT NUMBER IN GLOBAL COORDINATE SYSTEM.
$          IT HAS TO BE A SINGLE INTEGER (1 OR 2 OR 3).
$ ENTITY   = ENTITY AT WHICH THE CONSTRAINT IS APPLIED
$ ID       = ID OF THE ENTITY SELECTED ABOVE
$
$          SID      C          ENTITY  ID
GMSPC    1         3          FEFACE   1
GMSPC    1         3          FEFACE   2
GMSPC    1         3          FEFACE   3
GMSPC    1         3          FEFACE   4
GMSPC    1         3          FEFACE   5
GMSPC    1         3          FEFACE   6
GMSPC    1         3          FEFACE   7
GMSPC    1         3          FEFACE   8
$
$ DEFINE CONSTRAINTS IN THE Y-DIRECTION
$
GMSPC    1         2          GRID     1
$
$ DEFINE CONSTRAINTS IN THE X-DIRECTION AND Z-DIRECTION
$ FOR BOTH GRID 4 AND GRID 8. MULTIPLE GMSPC ENTRIES ARE
$ REQUIRED FOR CONSTRAINING MULTIPLE COMPONENTS.
$
GMSPC    1         1          GRID     4
GMSPC    1         3          GRID     4
GMSPC    1         1          GRID     8
GMSPC    1         3          GRID     8
$
$ DEFINE CONSTRAINTS FOR ROTATION FOR ALL GRIDS
$
GRDSET           0                               0          456
$
$ DEFINE GRIDS
$
GRID    1          0.0      0.0      0.0
GRID    2          5.0      0.0      0.0
GRID    3          5.0      45.0     0.0
GRID    4          5.0      90.0     0.0
GRID    5          5.0     135.0     0.0
GRID    6          5.0     180.0     0.0
GRID    7          5.0     225.0     0.0
GRID    8          5.0     270.0     0.0
GRID    9          5.0     315.0     0.0
GRID   10          0.0      0.0      0.5
GRID   11          5.0      0.0      0.5
GRID   12          5.0     45.0      0.5
GRID   13          5.0     90.0      0.5
GRID   14          5.0    135.0      0.5
GRID   15          5.0    180.0      0.5
GRID   16          5.0    225.0      0.5
GRID   17          5.0    270.0      0.5
GRID   18          5.0    315.0      0.5
GRID   21          4.0      0.0      0.0

```

```

GRID    22      1      4.0    45.0    0.0
GRID    23      1      4.0    90.0    0.0
GRID    24      1      4.0   135.0    0.0
GRID    25      1      4.0   180.0    0.0
GRID    26      1      4.0   225.0    0.0
GRID    27      1      4.0   270.0    0.0
GRID    28      1      4.0   315.0    0.0
GRID    29      1      4.0     0.0    0.5
GRID    30      1      4.0    45.0    0.5
GRID    31      1      4.0    90.0    0.5
GRID    32      1      4.0   135.0    0.5
GRID    33      1      4.0   180.0    0.5
GRID    34      1      4.0   225.0    0.5
GRID    35      1      4.0   270.0    0.5
GRID    36      1      4.0   315.0    0.5
$
$ DEFINE THE MATERIAL PROPERTIES (ALUMINUM)
$
MAT1     1      1.+7      0.3    2.538-4
PSOLID  1      1      GRID      SMECH
$
ENDDATA

```

This example illustrates a normal modes analysis, SOL 103, using the Lanczos method (EIGRL entry). The output requests are eigenvectors for all modes (DISPLACEMENT = ALL). Adaptivity analysis is required for all the elements. For this purpose, both the ADAPT and PVAL entries are used in conjunction with the ADAPT command in Case Control Section.

Results

```

1  CIRCULAR PLATE                               DECEMBER 13, 1993 MSC/NASTRAN 12/10/93 PAGE 929
0  EDGE SIMPLY SUPPORTED
0
    E I G E N V A L U E  A N A L Y S I S  S U M M A R Y  (LANCZOS ITERATION)
    BLOCK SIZE USED ..... 7
    NUMBER OF DECOMPOSITIONS ..... 2
    NUMBER OF ROOTS FOUND ..... 10
    NUMBER OF SOLVES REQUIRED ..... 10
    TERMINATION MESSAGE : REQUIRED NUMBER OF EIGENVALUES FOUND.
1  CIRCULAR PLATE                               DECEMBER 13, 1993 MSC/NASTRAN 12/10/93 PAGE 930
0  EDGE SIMPLY SUPPORTED
0
    R E A L  E I G E N V A L U E S
    MODE   EXTRACTION   EIGENVALUE   RADIANS   CYCLES   GENERALIZED   GENERALIZED
    NO.    ORDER         EIGENVALUE   RADIANS   CYCLES   MASS         STIFFNESS
1         1           3.465334E+07  5.886709E+03  9.368988E+02  1.000000E+00  3.465334E+07
2         2           1.182291E+08  1.087332E+04  1.730543E+03  1.000000E+00  1.182291E+08
3         3           1.483814E+08  1.218119E+04  1.938696E+03  1.000000E+00  1.483814E+08
4         4           2.502755E+08  1.582010E+04  2.517847E+03  1.000000E+00  2.502755E+08
5         5           2.654028E+08  1.629119E+04  2.592823E+03  1.000000E+00  2.654028E+08
6         6           3.027782E+08  1.740052E+04  2.769379E+03  1.000000E+00  3.027782E+08
7         7           8.642510E+08  2.939815E+04  4.678860E+03  1.000000E+00  8.642510E+08
8         8           8.645740E+08  2.940364E+04  4.679734E+03  1.000000E+00  8.645740E+08
9         9           1.166431E+09  3.415306E+04  5.435628E+03  1.000000E+00  1.166431E+09
10        10          2.009605E+09  4.482862E+04  7.134696E+03  1.000000E+00  2.009605E+09
1  CIRCULAR PLATE                               DECEMBER 13, 1993 MSC/NASTRAN 12/10/93 PAGE 931
0  EDGE SIMPLY SUPPORTED
0
*** USER INFORMATION MESSAGE 6502 (ADAPT)
    THE CURRENT SUPERELEMENT REQUIRES NO FURTHER P-ADAPTIVE ANALYSIS BECAUSE ALL P-ELEMENTS REACHED MAXIMUM P-VALUES.
0*** USER WARNING MESSAGE 4594. THE POSTCDB CONTAINS NO LEGITIMATE SURFACE OR VOLUME DEFINITIONS.
1  CIRCULAR PLATE                               DECEMBER 13, 1993 MSC/NASTRAN 12/10/93 PAGE 932
0  EDGE SIMPLY SUPPORTED                               ADAPTIVITY INDEX= 5
0
    EIGENVALUE = 3.465334E+07
    CYCLES = 9.368987E+02
    R E A L  E I G E N V E C T O R  N O . 1
    POINT ID.  TYPE   T1           T2           T3           R1           R2           R3
1         G   6.661354E-06  .0           -1.868621E+01  .0           .0           .0
2         G   1.333745E+00  -2.678128E-05  .0           .0           .0           .0
3         G   9.431428E-01  9.430584E-01  .0           .0           .0           .0
4         G   .0           1.333743E+00  .0           .0           .0           .0
5         G   -9.430543E-01  9.431386E-01  .0           .0           .0           .0
6         G   -1.333741E+00  2.663938E-05  .0           .0           .0           .0
7         G   -9.431665E-01  -9.430221E-01  .0           .0           .0           .0
8         G   .0           -1.333743E+00  .0           .0           .0           .0
9         G   9.430305E-01  -9.431752E-01  .0           .0           .0           .0
10        G   6.681371E-06  -6.129073E-08  -1.868621E+01  .0           .0           .0
11        G   -1.333732E+00  -3.440074E-04  .0           .0           .0           .0
12        G   -9.428723E-01  -9.433200E-01  .0           .0           .0           .0
13        G   3.471338E-04  -1.333743E+00  .0           .0           .0           .0
14        G   9.433249E-01  -9.428771E-01  .0           .0           .0           .0
15        G   1.333754E+00  3.440111E-04  .0           .0           .0           .0
16        G   9.429217E-01  9.432840E-01  .0           .0           .0           .0
17        G   -3.258019E-04  1.333743E+00  .0           .0           .0           .0
18        G   -9.432753E-01  9.429132E-01  .0           .0           .0           .0
21        G   1.343029E+00  -8.329611E-05  -5.443803E+00  .0           .0           .0
22        G   9.497321E-01  9.495923E-01  -5.443810E+00  .0           .0           .0
23        G   8.442988E-05  1.343021E+00  -5.443810E+00  .0           .0           .0
24        G   -9.495862E-01  9.497259E-01  -5.443810E+00  .0           .0           .0
25        G   -1.343013E+00  8.315749E-05  -5.443817E+00  .0           .0           .0
26        G   -9.497337E-01  -9.495776E-01  -5.443811E+00  .0           .0           .0
27        G   -7.570248E-05  -1.343021E+00  -5.443810E+00  .0           .0           .0
28        G   9.495844E-01  -9.497408E-01  -5.443809E+00  .0           .0           .0
29        G   -1.343016E+00  -2.135180E-04  -5.443803E+00  .0           .0           .0
30        G   -9.495108E-01  -9.498014E-01  -5.443810E+00  .0           .0           .0
31        G   2.196022E-04  -1.343021E+00  -5.443810E+00  .0           .0           .0
32        G   9.498075E-01  -9.495170E-01  -5.443810E+00  .0           .0           .0
33        G   1.343026E+00  2.135190E-04  -5.443817E+00  .0           .0           .0
34        G   9.495386E-01  9.497868E-01  -5.443811E+00  .0           .0           .0
35        G   -2.027499E-04  1.343021E+00  -5.443810E+00  .0           .0           .0
36        G   -9.497795E-01  9.495314E-01  -5.443809E+00  .0           .0           .0

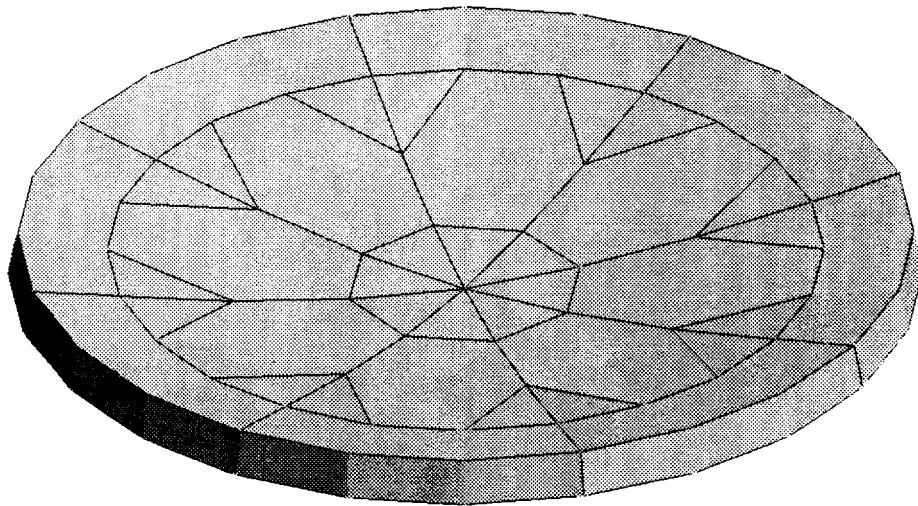
```

Figure 13-64 Abridged Output from the Circular Solid Model

201001001	G	1.333745E+00	-2.678128E-05	.0	.0	.0	.0
201001002	G	1.288204E+00	3.459260E-01	.0	.0	.0	.0
201001003	G	1.155554E+00	6.657130E-01	.0	.0	.0	.0
201001004	G	9.431428E-01	9.430584E-01	.0	.0	.0	.0
201001005	G	1.354012E+00	-1.118739E-07	-1.801449E+00	.0	.0	.0
201001006	G	1.307760E+00	3.511871E-01	-1.801487E+00	.0	.0	.0
201001007	G	1.173140E+00	6.758680E-01	-1.801314E+00	.0	.0	.0
201001008	G	9.574580E-01	9.574031E-01	-1.801455E+00	.0	.0	.0
201001009	G	1.357469E+00	-5.671428E-05	-3.626504E+00	.0	.0	.0
201001010	G	1.311071E+00	3.521154E-01	-3.626570E+00	.0	.0	.0
201001011	G	1.176151E+00	6.776471E-01	-3.626329E+00	.0	.0	.0
201001012	G	9.599283E-01	9.598196E-01	-3.626513E+00	.0	.0	.0
201001013	G	1.343029E+00	-8.329611E-05	-5.443803E+00	.0	.0	.0
201001014	G	1.297096E+00	3.483891E-01	-5.443859E+00	.0	.0	.0
201001015	G	1.163644E+00	6.704805E-01	-5.443617E+00	.0	.0	.0
201001016	G	9.497321E-01	9.495923E-01	-5.443810E+00	.0	.0	.0

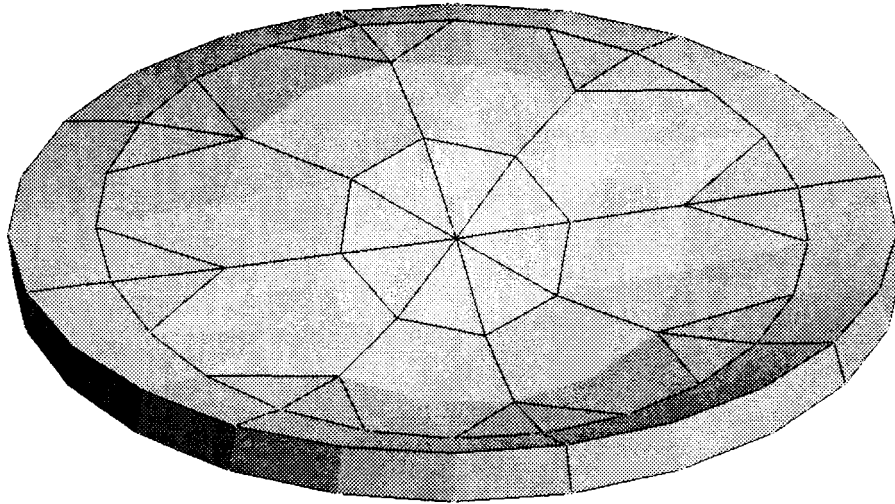
Figure 13-64 Abridged Output from the Circular Solid Model (continued)

The mode shapes for modes 1, 5, 8, and 9 are shown in **Figure 13-65**. Mode 1 is the fundamental mode, mode 5 is the one nodal diameter mode, mode 8 is the two nodal diameters mode, and mode 9 is one nodal circle mode.

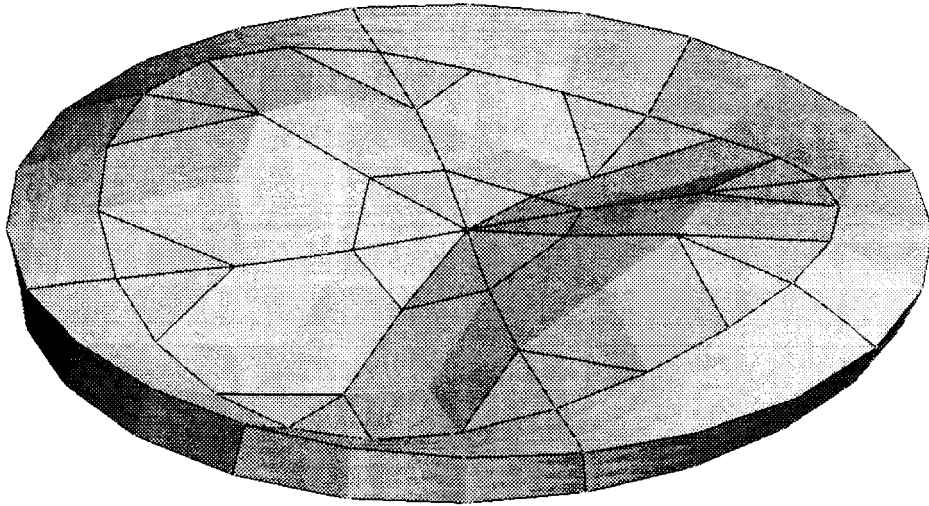


Mode 1: Fundamental Mode

Figure 13-65 The Mode Shapes for Modes 1, 5, 8, and 9

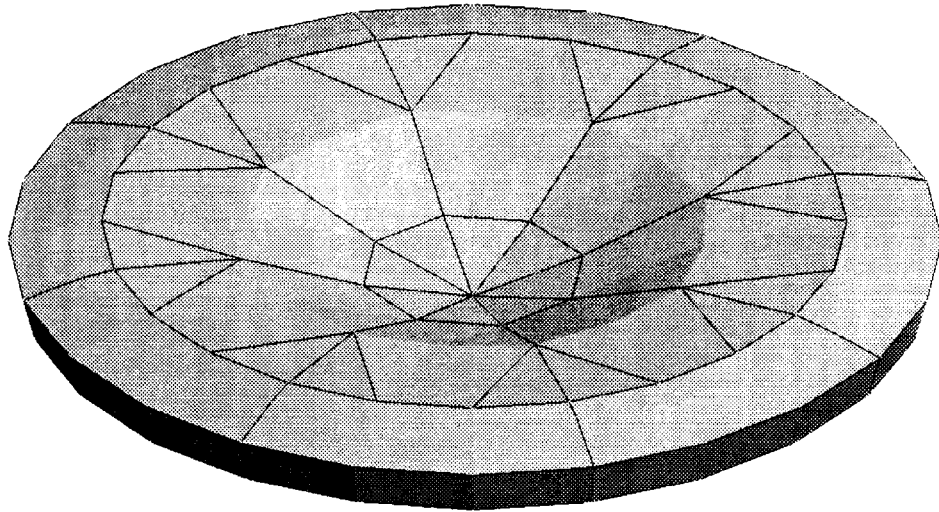


Mode 5: One Nodal Diameter Mode



Mode 8: Two Nodal Diameter Mode

Figure 13-65 The Mode Shapes for Modes 1, 5, 8, and 9 (continued)



Mode 9: One Nodal Circle Mode

Figure 13-65 The Mode Shapes for Modes 1, 5, 8, and 9 (continued)

Comparison of the natural frequencies obtained from the adaptivity analysis for the solid circular model to the theoretical natural frequencies for the circular plate is shown in **Table 13-25**. The discrepancies between the two values decrease as the p-order increases.

Table 13-25 Comparison of the Natural Frequencies

Mode	Theoretical (Hertz)	Numerical (Hertz)
Fundamental	954.1	936.9
One Nodal Diameter	2657.7	2592.8
Two Nodal Diameters	4913.9	4679.7
One Nodal Circle	5697.8	5435.6

Further studies were performed in which the p-order was kept constant and the thickness of the solid model was varied. Results showed that as the thickness of the circular solid model decreases, the solid model behaves like a plate model, which is to be expected.

Reference:

[1] Roark, R. J. and Young, W. C. *Formulas for Stress and Strain*. 5th edition. McGraw-Hill, Inc., 1975, p. 578.

13.8.9 Clamped Rhombic Plate

The first eigenfrequency of a rhombic plate is calculated using SOL 103. **Figure 13-66** shows a plate clamped at the left end and the first eigenmode. Compare the convergence behavior of three different models:

- One CHEXA p-element with p-orders varying from $p = 1$ to $p = 8$
- One CQUAD p-element with p-orders varying from $p = 1$ to $p = 8$
- Multiple CQUAD h-elements with varying meshes (1×1 , 2×2 , 4×4 , 8×8 , 16×16 , 32×32)

The convergence behavior of the three models is shown in **Figure 13-67**. The first eigenfrequency is normalized with 2.932 rad/sec, which is the value from a series solution using Kirchhoff plate theory. All three models exhibit excellent convergence behavior.

The solid p-element and the shell p-element models converge when the p-level is increased. The CQUAD4 h-element model converges when the mesh is refined. If the error rate in the first eigenfrequency must be under 0.2%, then either: (a) a single shell or solid p-element with $p = 8$, or (b) 16×16 shell h-elements should be used. The shell p-element needs the smallest number of degrees-of-freedom to achieve the required accuracy. For a comparison of the degrees-of-freedom, see **Table 13-26**.

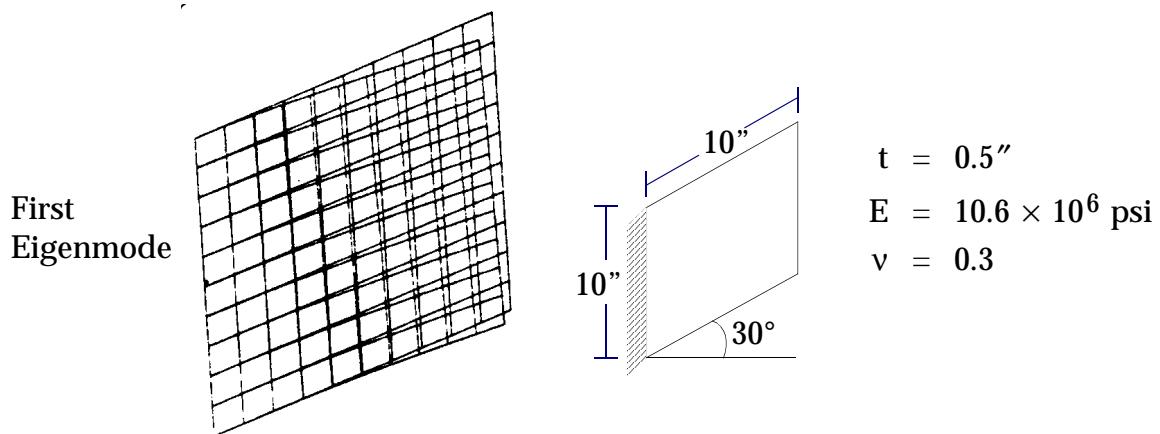


Figure 13-66 Rhombic Plate

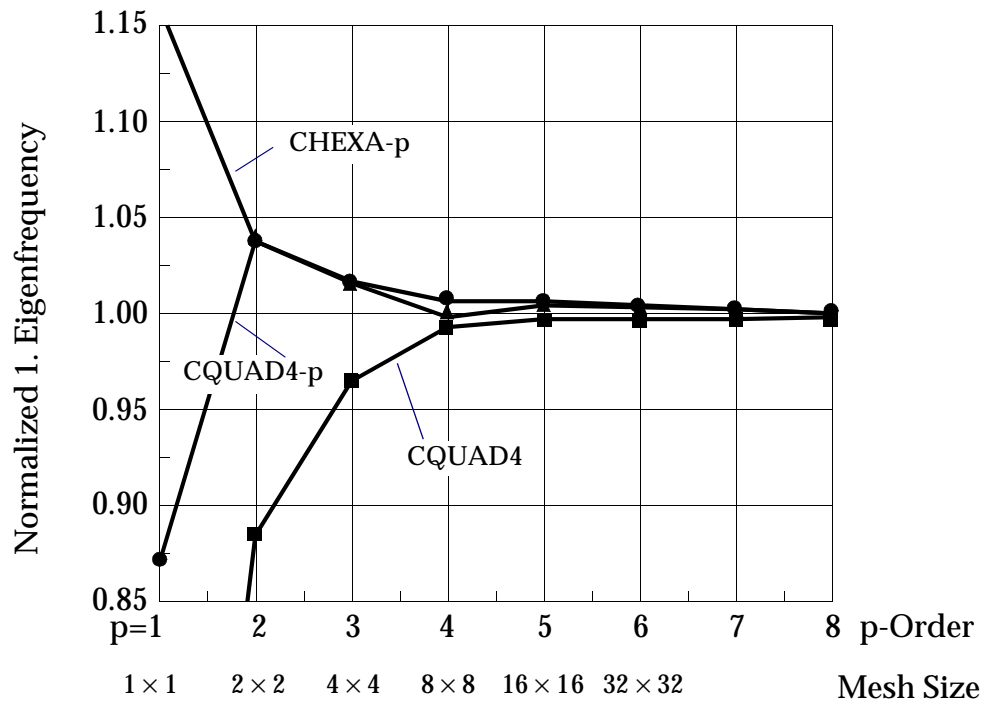


Figure 13-67 Normalized First Eigenfrequency for Different p-Levels and Mesh Sizes

Table 13-26 Comparison of the Number of Degrees-of-Freedom for 0.2% Accuracy in the First Eigenfrequency

Element	Mesh Size and p-Level	Normalized Eigenfrequency	Degrees of Freedom
CQUAD4	16 × 16	0.9977	1445
CQUAD4-p	p = 8	1.0012	264
CHEXA-p	p = 8	1.0017	510

13.8.10 Helix Modeled with CBEAM p-Elements

A 360° helix is modeled with nonadaptive beam h-elements and with adaptive beam p-elements. The helix is clamped at the bottom and free at the top, with a single force applied at the top. The undeformed and deformed helix is shown in [Figure 13-68](#). The normalized deflection at the tip is shown in [Figure 13-69](#) for different mesh sizes.

Both beam h- and p-elements converge well. If the error in the deflection must be under 0.2%, then 8 beam p-elements with $p = 3$ (45° angle for each element) or 64 beam h-elements should be used. A comparison of the degrees-of-freedom (see

Table 13-27) shows that the beam p-elements require a smaller number of degrees-of-freedom than the number required by the beam h-elements in order to achieve the required accuracy.

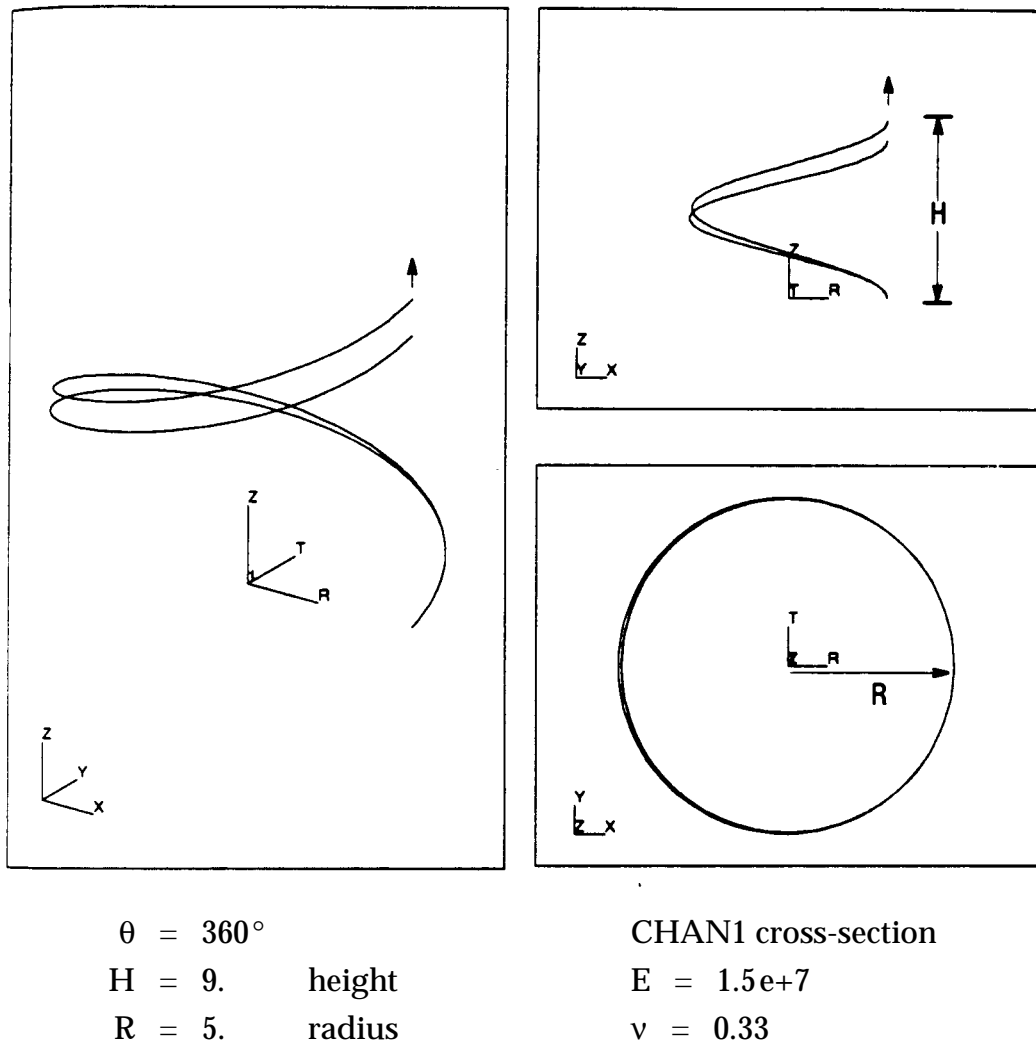


Figure 13-68 Helix

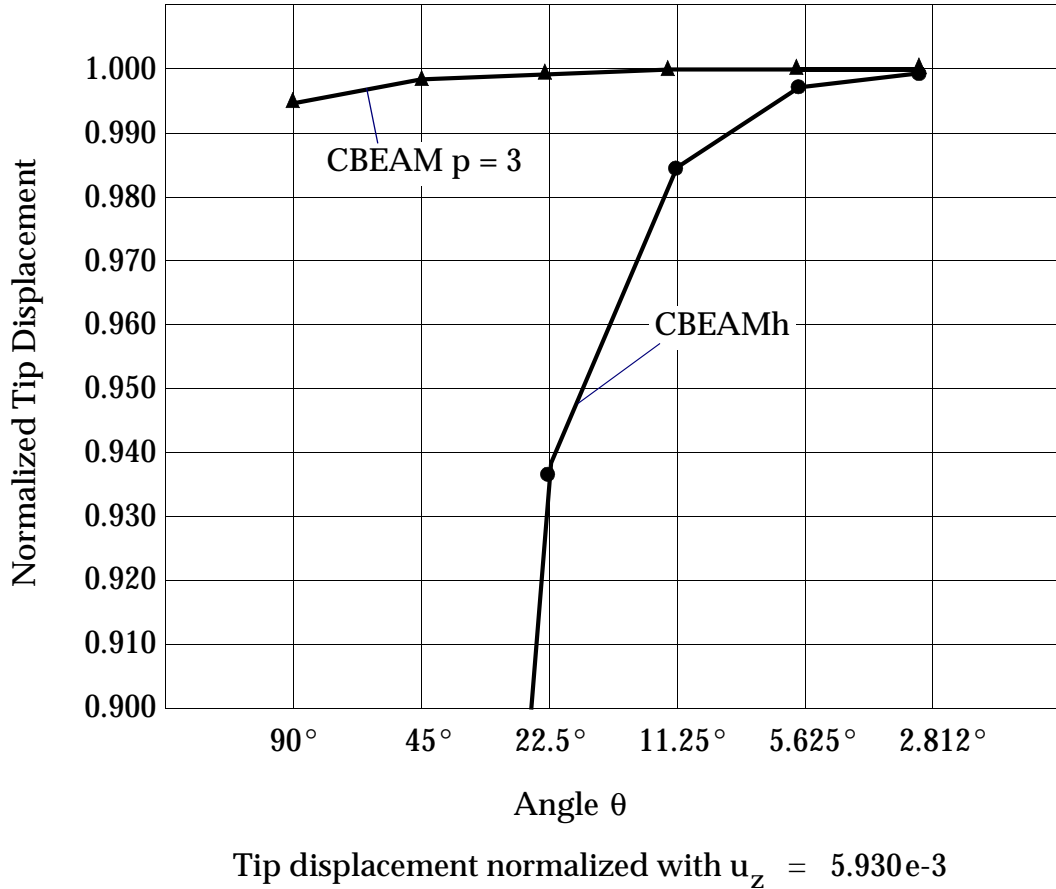


Figure 13-69 Normalized Tip Displacement for Different Mesh Sizes

Table 13-27 Comparison of the Number of Degrees-of-Freedom for 0.2% Accuracy in the Tip Displacement

Element and Mesh	Normalized Tip Displacement	Degrees of Freedom
CBEAM h, $\theta = 5.6^\circ$; 64 Elements	0.9970	390
CBEAM p, $\theta = 45^\circ$; p = 3,8 Elements	0.9978	150

13.8.11 Plate with Stiffener

In stiffened structures, an efficient practice is to model the stiffeners using beam elements with offsets. Typically, nonadaptive h-elements are used. The example below demonstrates the benefits of p-elements for this class of problems.

A plate with stiffener is loaded in the middle and simply supported at the ends (see **Figure 13-70**). The plate is modeled with 10×2 shell elements and the stiffener is modeled with 10 beam elements with offsets. Compare the stress results of two models, an h-element model and a p-element model, with a uniform p-order of $p = 2$ for shells and beams. The solution is compared with the solution derived from beam theory.

The nonadaptive h-elements produce very inaccurate stresses with errors around 30%, as shown in **Figure 13-71**. The adaptive p-elements with a uniform p-order of $p = 2$ yield very accurate results--the error rate is only around 4%. In the h-element model, the normal forces in the shell and beam elements are constant within each element, which causes the high stress error. In the p-element model, the normal forces in each element are linear ($p = 2$), which improves the stress distribution in the cross-section of the beam.

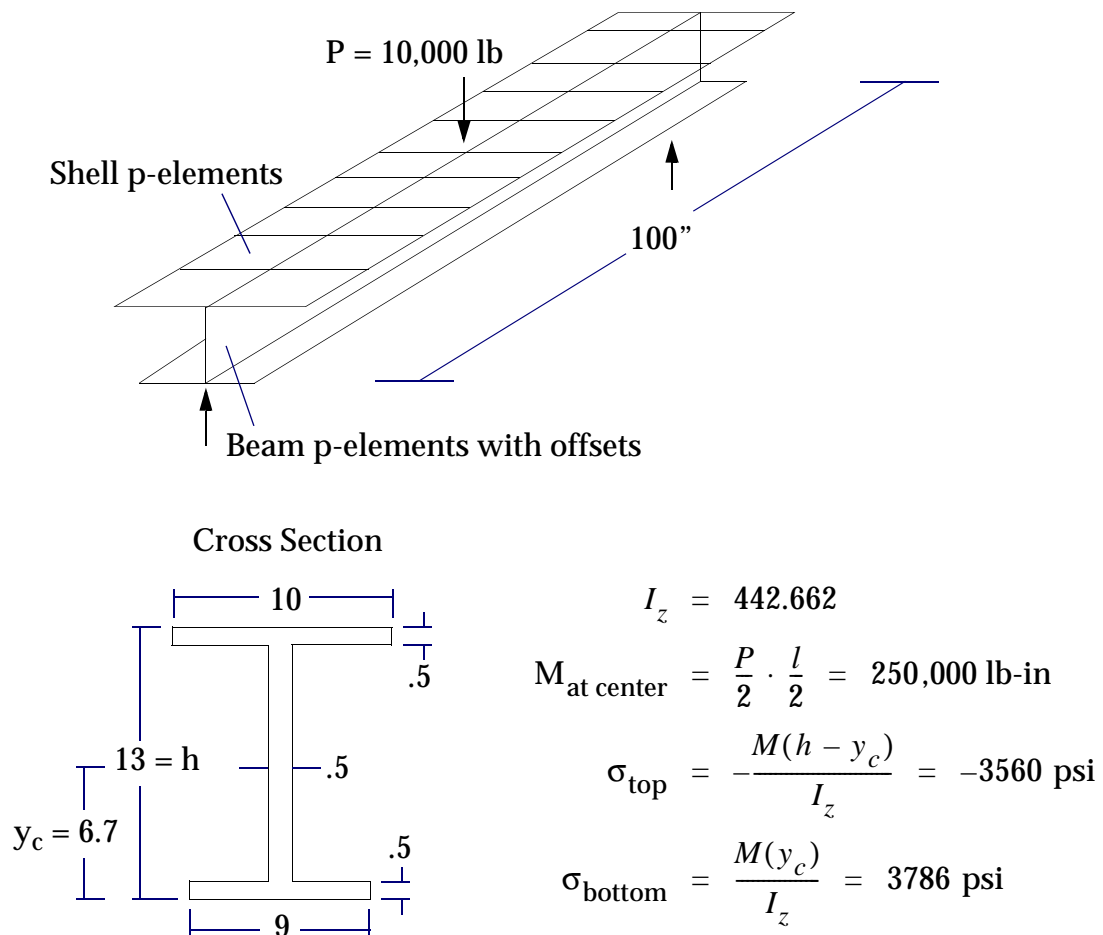
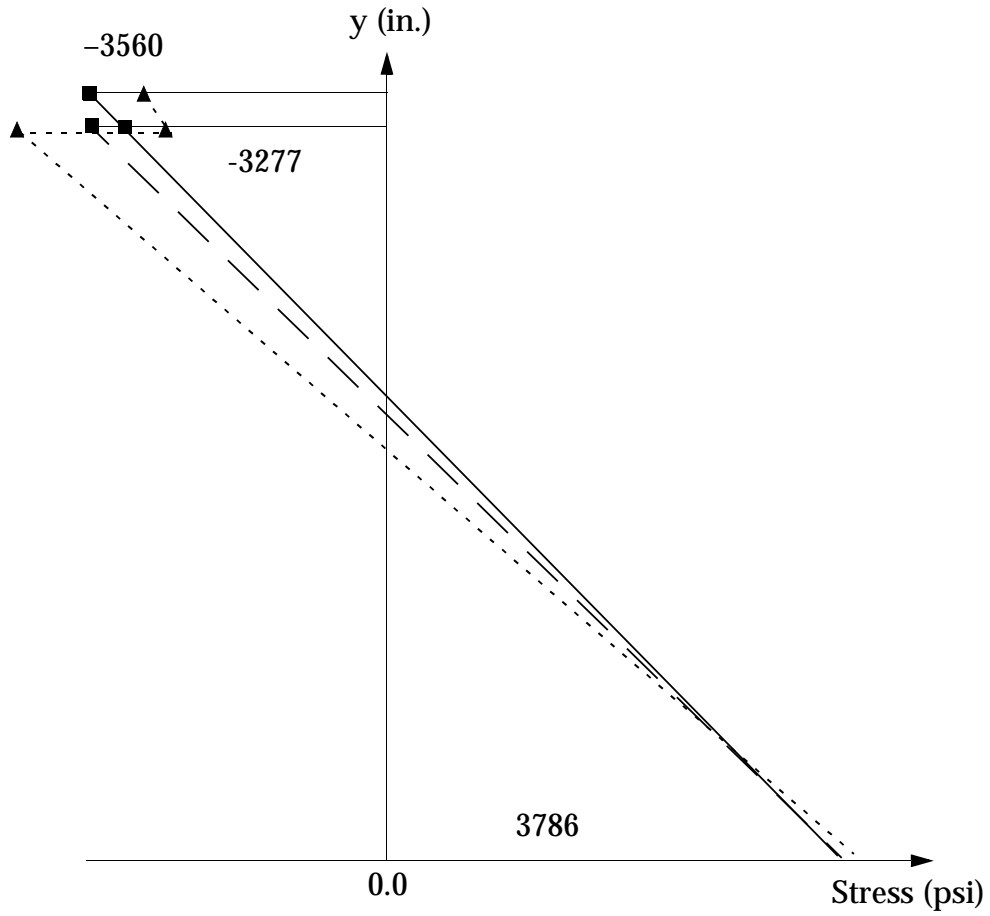


Figure 13-70 Plate with Stiffener



Stress [psi]	y [in]
3786.	0.
-3277.	12.5
-3560.	13.0

beam theory

———

Stress [psi]	y [in]
3978.	0.
-4311.	12.5
-2804.	12.5
-3082.	13.0

h-elements

▲ - - - - ▲

Stress [psi]	y [in]
3834.	0.
-3545.	12.5
-3250.	12.5
-3590.	13.0

p=elements
p = 2

■ - - - ■

Figure 13-71 Stress Distribution in the Middle of the Beam

13.8.12 Solid p-Elements in Linear Dynamic Solution Sequences

Solid p-elements are available in the dynamic solution sequences SOL 107, to SOL 112. The p-elements do not adapt; the analysis is done for a fixed p-level. Displacements, velocities, accelerations, stresses and strains are output in SORT1 or SORT2 format. The complex output quantities are in rectangular format (real/imaginary) or in polar format (magnitude/phase).

User Interface in MSC.Nastran

Output for p-elements is requested with the Case Control command DATAREC, where SORT1 or SORT2 formats can be chosen. For complex eigenvalues (SOL 107, SOL 110) and frequency response (SOL 108, SOL 111), the default is SORT1. For transient response (SOL 109, SOL 112), the default is SORT2.

The output quantities are requested with DISP, VELO, ACCE, STRESS, or STRAIN in the Case Control, and/or in the OUTPUT Bulk Data entry. In the complex solution sequences, the formats are rectangular (REAL, IMAG) or polar (MAGNITUDE, PHASE); the rectangular format is the default. REAL, IMAG or PHASE is requested in the Case Control command or in the OUTPUT Bulk Data entry.

Current Limitations and Recommendations

The following limitations and recommendations apply to solid p-element usage in linear dynamic solution sequences:

- The output request must be all in SORT1 or all in SORT2 format for a single run.
- PARAM,CURVPLOT,1 must be specified to get SORT1 output in transient response (SOL 109 and SOL 112).
- If an ADAPT entry is specified with multiple p-levels, the analysis is done with the first p-level.

Example

The transient response of a cantilever subject to a constant step load is calculated in SOL 109 (see [Figure 13-72](#)). The following is a comparison of the results between the two different models:

- HEXA8 h-elements, $4 \times 4 \times 40$ element mesh.
- HEXA p-elements, $1 \times 1 \times 4$ element mesh with $p = 3$.

The plot displacement versus time graph shown in [Figure 13-73](#) illustrates that the results are in good agreement.

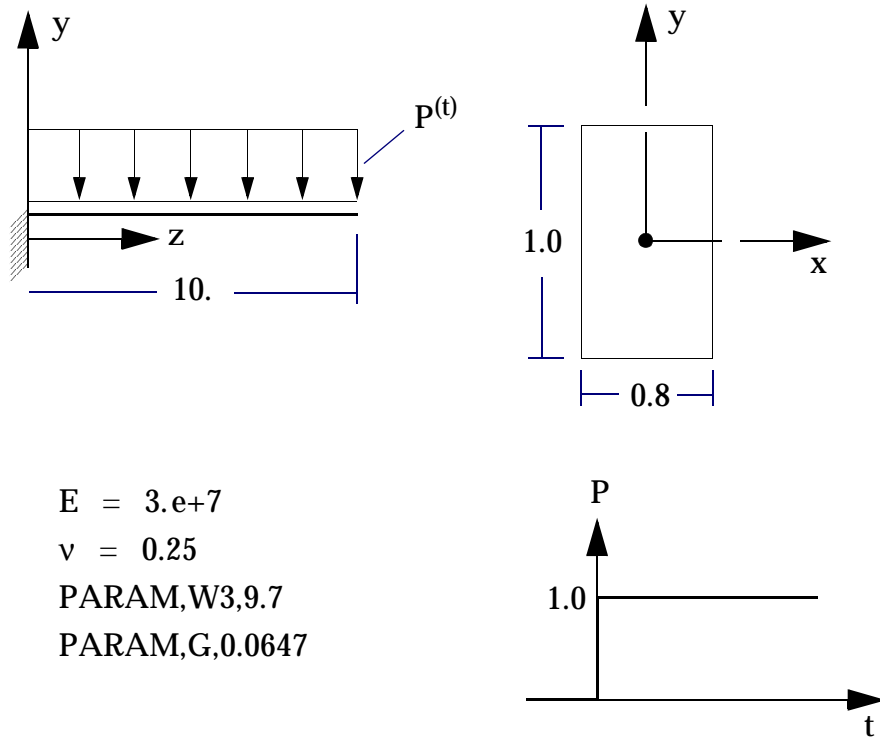


Figure 13-72 Cantilever with Step Load

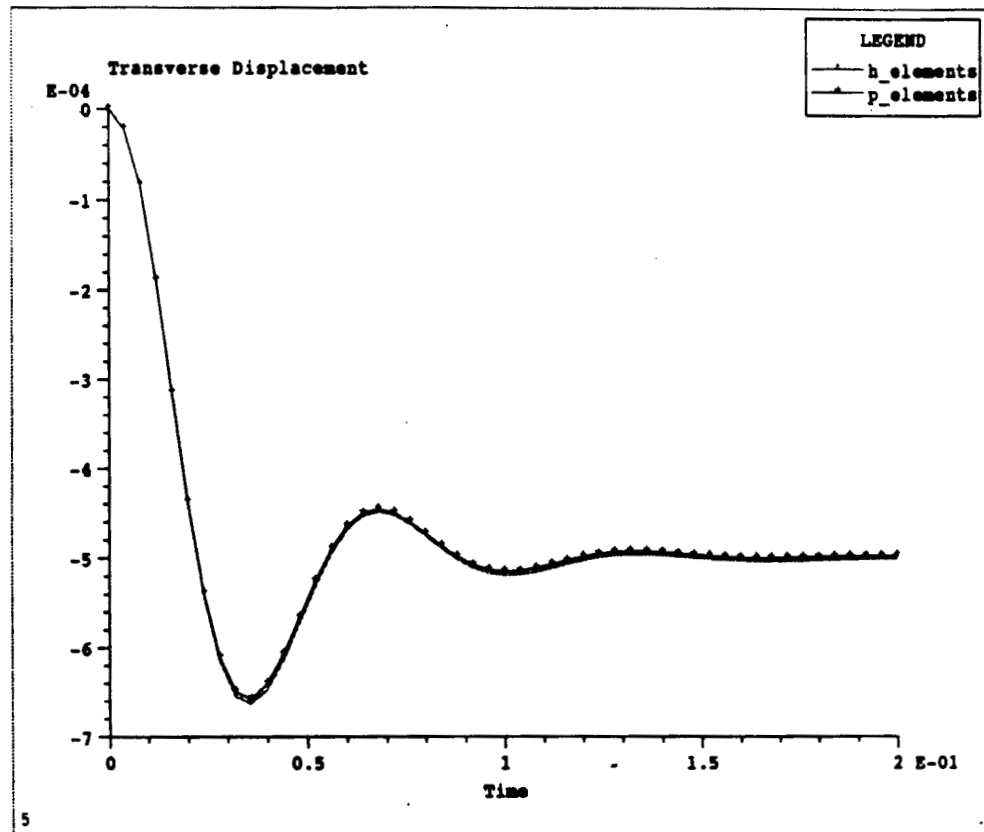


Figure 13-73 Tip Displacement of Cantilever

13.8.13 p-Element Thermal Analysis

Introduction

MSC.Nastran supports p-version elements for linear steady state thermal analysis (SOL 101). The definitions of model geometry, elements, p-values, and error estimation methods follow the general terminologies of the p-element structural analysis. A major difference exists in the specification of thermal loads and boundary conditions, which requires new or modified Bulk Data entries for thermal analysis.

The following sections provide usage guidelines and examples for the p-version thermal analysis.

Geometry

The geometry definitions are the same as those defined for the p-element structural analysis.

Elements

Six kinds of elements--CHEXA, CPENTA, CTETRA, CQUAD4, CTRIA3, and CBEAM--may be defined as p-elements for thermal analysis.

Materials

The standard isotropic, orthotropic, and anisotropic materials are available for p-version thermal elements. The only material property required is thermal conductivity which must be constant for linear steady state analysis.

Loads and Boundary Conditions

The following thermal loads and boundary conditions are supported in p-version thermal analysis.

- Prescribed temperatures applied on point, curve, 2D edge, surface, or 3D surface.
- Normal or directional heat fluxes applied on point, curve, 2D edge, surface, 3D surface, or solid.
- Volumetric heat generation within conduction elements.
- Free convection applied on point, curve, 2D edge, surface, or 3D surface.

Table 13-28 summarizes the Bulk Data entries for p-version thermal loads and boundary conditions. A full description of these entries may be found in the *MSC.Nastran Quick Reference Guide*.

Table 13-28 Bulk Data Entries for p-Version Thermal Loads and Boundary Conditions

Load or Boundary Condition Type	Bulk Data
Prescribed Temperature	GMBC
Normal or Directional Heat Flux	GMLOAD
Volumetric Heat Generation	GMQVOL
Free Convection	GMCONV

p-Version Adaptivity

p-values and error estimation methods are the same as those for p-element structural analysis.

Data Recovery

Similar to structural output, the DATAREC and OUTRCV Case Control commands as well as the OUTPUT and OUTRCV Bulk Data entries define the output of p-version thermal analysis. Temperatures, temperature gradients, heat fluxes, polynomial values, and element errors may be printed, plotted, or punched. The TEMP output command of the OUTPUT Bulk Data entries requests for calculating temperatures while the FLUX output command of the OUTPUT Bulk Data entries requests for recovering temperature gradients and heat fluxes for conduction elements.

Thermal Stress Analysis

The interface to input temperature results as temperature loads for thermal stress analysis is described in the Case Control command “**TSTRU**” on page 463 of the *MSC.Nastran Quick Reference Guide*.

Limitations and Recommendations

- Only Solution Sequence 101 (linear steady state) performs p-element thermal analysis.
- Temperature boundary conditions can be spatially varying, but must be independent of time.
- Heat fluxes and volumetric heat generation can be spatially varying, but must be independent of temperature or time.
- The convection heat transfer coefficients and ambient temperatures of free convection boundary conditions can be spatially varying, but must be independent of temperature or time.
- It is important to select the appropriate entity type while applying loads and boundary conditions. Concentrated loads or boundary conditions will cause singularities or inaccurate results and should be avoided. For example, the entity type FEFACE or GMSURF should be used to prescribe temperatures on a surface, rather than the entity type FEEDGE, GMCURVE, or GRID.

Examples

There are several example problems posted on the web site. The URL address is:

<http://www.si.macsch.com/nsp/SampleProb/heat/pelem/pelem.html>

The following three examples are recommended for the reference guide.

- Example 1 – Chip set model with solid elements (Problem 10)

- Example 2 -- Cone shape geometry with shell elements (Problem 7)
- Example 3 -- Thermal stress analysis (Problem 2 or Problem 3)

13.9 Response Spectrum Analysis

See the *MSC.Nastran Advanced Dynamic Analysis User's Guide*.

13.10 Viscoelastic Materials in Frequency Response Analysis

See the *MSC.Nastran Advanced Dynamic Analysis User's Guide*.

13.11 Introduction to Parallel Numeric Enhancements

Shared Memory Parallel (SMP) processing was first introduced in MSC.Nastran in 1987. Shared memory architectures feature multiple processors, each processor sharing a common memory and disk I/O. SMP is invoked by adding “PARALLEL=n” to the nastran submittal command, where “n” is the number of desired processors.

With the rise in popularity of distributed network computing, a new parallel paradigm was needed to accommodate the distributed computing environment. In this environment, each processor has its own memory and disk I/O, and communication is done through a network. The MSC.Nastran Distributed Memory Parallel (DMP) method addresses this environment.

MSC.Nastran DMP achieves parallelism by dividing the problem to be solved at a high level. Division of the problem into sub-problems may be performed in different ways. For example, problems posed in the frequency domain may be divided based naturally on frequency. A suitable and effective domain decomposition serves as the cornerstone of MSC.Nastran DMP parallel performance.

Previous versions of MSC.Nastran provided domain decomposition techniques in the frequency domain and the geometric domain. Frequency domain decomposition allows a problem to be divided into frequency sub-ranges and solved independently in each sub-range. Geometric domain decomposition seeks to divide the model geometrically, based on grid point connectivity, into sub-models. Each sub-model is then solved independently, followed by a solution at the geometric boundary defined by the domain decomposition.

MSC.Nastran 2004 provides a third domain decomposition option. Degree-of-Freedom (DOF) domain decomposition provides for the division of the problem at the degree of freedom level, after single- and multi-point constraints have been eliminated. Decomposition of the problem at the DOF level minimizes some of the potential complications which may arise with domain decomposition at the grid point level, and is a more appropriate decomposition choice in some cases. See “[Domain Decomposition](#)” on page 890 for further discussion on the various domain decomposition techniques, and “[New Distributed Memory Parallel Methods](#)” on page 891 for an introduction to modal analysis techniques based on DOF domain decomposition.

MSC.Nastran 2004 also allows, for the first time, a combination of frequency and geometric domain decomposition techniques in modal analysis. This combination, called the Hierarchic Method, first divides the problem geometrically at the grid point level. Then during the eigenvalue analysis, each sub-model is further divided in the frequency domain, yielding an additional level of parallelism.

Also for the first time, DMP is supported in SOL 200 design optimization. DMP methods are available during the analysis phase of the design cycle. DMP methods for normal modes (ANALYSIS=MODES), modal dynamic analyses (ANALYSIS=MFREQ, MTRAN, MCEIG, and FLUTTER), and direct frequency response (ANALYSIS=DFREQ) are supported. Please see “[New Distributed Memory Parallel Methods](#)” on page 891 for more details.

The DOMAINSOLVER command was revised and extended in order to handle new DMP developments discussed in this chapter. Please see “[Inputs](#)” on page 893 for a complete description.

“[Method Selection Guidelines](#)” on page 897, helps the user select the appropriate strategy for solving large problems with DMP. Limitations that preclude the use of DMP are presented in “[Limitations](#)” on page 898. Finally, “[Examples](#)” on page 900 show how to execute DMP strategies with the appropriate commands and submit procedures.

[Table 13-29](#) contains a list of computers and corresponding operating systems that support DMP. The system must also have Message Passing Interface (MPI) software installed. MPI software provides the interface for communication between the distributed processors. DMP analyses must be performed on the same type of machines. [Table 13-29](#) summarizes the DMP system requirement.

Table 13-29 DMP System Prerequisites

Platform	System Requirements	
AIX	Processor OS MPI	Any AIX 4.3.2 POE 2.3
Compaq Alpha UNIX	Processor OS MPI	Any UNIX 4.0f, 5.0 Compaq MPI 1.8.1
HP	Processor OS MPI	PA-RISC 2.0 HP-UX 11.x, 64-bit <i>No additional software is required.</i>

Table 13-29 DMP System Prerequisites (continued)

Platform	System Requirements	
SGI	Processor OS MPI	R8K, R10K, R12K, R14K, R16K IRIX 6.5 and maintenance releases MPT 1.4 and later
Linux	Processor OS MPI	IA32 2.2.17 or later LAM 6.5.6
Linux	Processor OS MPI	IA64 2.4.3 or later LAM 6.5.6
SUN	Processor OS MPI	SPARC64 Solaris 8.0 mpi2
NEC	Processor OS MPI	SX-6 SUPER-UX 11.1 MPI/SX 5.0
Cray	Processor OS MPI	SV1 Unicos 10.0 MPT 1.4.0.0
Fujitsu	Processor OS MPI	Fujitsu vector processor UXP/V, UNIX V5.4 with vector capability enhancement mpi2

13.11.1 Domain Decomposition

Introduction

Decomposition of a problem in the frequency domain is a natural and intuitive way to divide a problem into smaller pieces. Frequency domain decomposition is available in MSC.Nastran for the following analysis types: normal modes analysis, modal dynamic analysis, and direct frequency response (SOLs 103, 111 and 112, and 108, respectively). In addition, it is available in the analysis portion of SOL 200 (design optimization). For 103, 111, and 112, the Lanczos eigenvalue extraction method will allow for frequency domain decomposition based on the number of available processors. In 108 and 111, the frequency response calculations are also subject to

domain decomposition. The frequency domain offers straightforward parallelism since results at every frequency are mathematically independent. Generally, the number of frequency domains will equal the number of processors devoted to the problem. This method is known as Frequency Domain Decomposition.

Another intuitive way to subdivide a problem is to physically divide the problem into discrete parts, or components. In MSC.Nastran, the geometric domain is considered to be the collection of grid and scalar points, and their connectivity via elastic and rigid elements. Geometric domain decomposition is the product of an automatic partitioning algorithm based on the “nested dissection” of the model into smaller and smaller parts. Nested dissection is in turn a product of general graph theory. The product of a geometric domain decomposition is a collection of N discrete components, plus a common geometric boundary. The N components are independent and may be processed in parallel. In MSC.Nastran, this is accomplished utilizing superelement technology. For most DMP methods, N equals the number of available processors. Geometric domain decomposition is available in SOLs 101, 103, 111, and 112.

In MSC.Nastran 2004, a third domain decomposition method (DOF based Domain Decomposition) has been added.

13.11.2 New Distributed Memory Parallel Methods

Introduction

MSC.Nastran 2004 contains two new DMP features: a Hierarchic distributed parallel Lanczos method, and DOF based domain decomposition. Starting with MSC.Nastran 70.7, various Distributed Memory Parallel (DMP) solution techniques have been implemented. Direct frequency response (SOL 108) and real eigenvalue analysis (SOL 103) in the frequency domain, and static analysis (SOL 101) in the geometric domain, were implemented in Version 70.7. This was followed in MSC.Nastran 2001 by normal modes analysis (SOL 103) and Automated Component Mode Synthesis (ACMS) in the geometric domain.

Hierarchic Distributed Parallel Lanczos Method

This new method combines two existing principles of domain decomposition in the distributed parallel normal modes analysis: frequency and geometry domains. The hierarchic approach still solves the global eigenvalue problem exactly, as opposed to the component modal reduction method, which is an approximate approach.

There are two layers of the new hierarchic normal modes analysis method. First, in the outer layer of the hierarchy, the global finite element matrices are automatically partitioned and each processor sees only a local partition. In the inner layer of the hierarchy, the frequency range is again automatically subdivided and processors that are working on the same geometry partition work on different frequency segments.

More specifically, in the new method a subset of processors solves the same global problem in a given frequency interval. Another subset solves the problem in another interval. Within one subset, however, the processors work on their local geometry and communicate with the others on the boundary. Each processor of a given subset computes a (geometrically) local section of the global eigenvectors of the subset's frequency interval and communicates it with its partners in the subset. The process concludes with the final collection of the global mode shapes of these intervals on a master processor to facilitate follow-on modal analysis techniques or appropriate postprocessing.

Inputs

The user interface is identical to that of the geometric domain parallel method, but to invoke the hierarchic method the number of domains specified must be the square root of the number of processors requested. For example, to execute a hierarchic Lanczos process with 2 geometric partitions and 2 frequency segments simultaneously, a submittal command similar to the one shown below must be used:

```
nastran jobname dmp=4 ...
```

Furthermore, the following executive control statement must also be used.

```
DOMAINSOLVER MODES (NUMDOM=2, PARTOPT=GRID)
```

The PARTOPT keyword specifies the method of partitioning; “GRID” specifies the grid based geometric domain, which is equivalent to the MSC.Nastran 2001 “GDMODES” method. For complete information on the updated DOMAINSOLVER command, please see “[Inputs](#)” on page 893.

DOF Based Domain Decomposition

The Degree of Freedom (DOF) domain is defined as the degree of freedom set left over after all constraints have been eliminated from the finite element problem. DMP methods based on DOF domain decomposition postpone the partitioning of the global problem until after all constraints have been removed. For example, in a conventional real eigenvalue analysis, DOF domain decomposition is performed on the F-set (the KFF and MFF matrices).

A domain decomposition is performed on the degrees of freedom in this set, and the associated connectivity of these degrees of freedom. The partitioning algorithm is based on a multilevel nested dissection, which yields a multilevel elimination tree. The domains created are all naturally independent, and so may be processed by multiple processors in parallel. A common boundary is also generated, which is solved with the aid of message passing interface (MPI) communication.

The purposes for introducing a new parallel domain are twofold. Generally, it is a sound strategy to offer more than one way to accomplish any significant task, such as domain decomposition. Secondly, it is a convenient means of avoiding potential limitations encountered during the grid point based geometric domain decomposition. The MSC.Nastran geometric domain decomposition relies on superelement technology to drive the solution process. In some cases, this technology limits the ways problems can be posed. For example, models which contain duplicate element IDs cannot be processed with MSC.Nastran superelements. This may occur when a rigid element is added to a model whose ID is the same as an existing QUAD4 element, for example. The DOF domain thus offers more flexibility to provide a robust DMP solution.

In MSC.Nastran 2004, DOF domain decomposition is available in SOL 103 and in the normal modes analysis portion of SOLs 111, 112, and 200.

User Interface

To run a SOL 103 real eigenvalue analysis with the distributed Lanczos method, using DOF domain decomposition, a submittal command similar to the following can be used:

```
nastran jobname dmp=N ...
```

Furthermore, the following DOMAINSOLVER Executive Command Statement must be added.

```
DOMAINSOLVER MODES (PARTOPT=DOF)
```

where “N” is the number of available processors.

13.11.3 Inputs

The user interface for Distributed Memory Parallel (DMP) MSC.Nastran is comprised of a combination of the “dmp” keyword on the submittal line and the DOMAINSOLVER Executive statement. The DOMAINSOLVER command has been changed from MSC.Nastran 2001 such that many of the old keywords are no longer supported.

The format of the DOMAINSOLVER Executive Statement is as follows:

$$\text{DOMAINSOLVER} \begin{bmatrix} \text{STAT} \\ \text{MODES} \\ \text{FREQ} \\ \text{ACMS} \end{bmatrix} \left[\left(\text{PARTOPT} = \begin{bmatrix} \text{DOF} \\ \text{GRID} \\ \text{FREQ} \end{bmatrix}, \text{NUMDOM} = \text{int}, \text{UPFACT} = \text{real}, \right. \right. \\ \left. \left. \text{TREE} = \begin{bmatrix} \text{SINGLE} \\ \text{MULTI} \end{bmatrix}, \text{ALLOC} = \begin{bmatrix} \text{STATIC} \\ \text{DYNAMIC} \end{bmatrix}, \text{PRINT} = \begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix} \right) \right]$$

Examples:

DOMAINSOLVER STAT (PARTOPT=DOF)

DOMAINSOLVER ACMS (UPFACT=15,NUMDOM=128)

For DOMAINSOLVER ACMS, there are additional parameters UPFACT, ALLOC, and PRINT. The keywords function as follows:

STAT	Linear Statics
MODES	Normal Modes
FREQ	Frequency Response
ACMS	Automated Component Modal Synthesis

The description of the parameters is as follows:

PARTOPT Partitioning option. Selects which domain is to be decomposed.

DOF Degree of Freedom domain

GRID Grid point (Geometric) domain

FREQ Frequency domain

The default is dependent upon solution sequence. See [Table 13-31](#) for further descriptions.

NUMDOM Selects the number of domains as follows:

ACMS If NUMDOM = 0 or 1, then the model will not be split. Default depends on the model size (number of grid points) as follows:

NGRID <= 2,000; NUMDOM = 4

2,000 <= NGRIDS < 10,000; NUMDOM = 16

10,000 <= NGRIDS < 50,000; NUMDOM = 32

50,000 <= NGRIDS < 100,000; NUMDOM = 64

100,000 <= NGRIDS < 300,000; NUMDOM = 128

300,000 >= NGRIDS; NUMDOM = 256

The model will be split into NUMDOM domains.

STAT	Default = <i>dmp</i> ; if NUMDOM has any other value, it will automatically be set to <i>dmp</i> (= no. of processors used for the run). The model will be divided into NUMDOM domains in either the geometric (grid based) or DOF domains, depending on the value of PARTOPT.
MODES	Default = <i>dmp</i> ; if NUMDOM has any other value, the Hierarchical method will be used. For the hierarchical method, NUMDOM must be the square root of <i>dmp</i> . Otherwise it NUMDOM will be reset to <i>dmp</i> (= no. of processors used for the run). The model will be divided into NUMDOM domains in either the geometric (grid based) or DOF domains, depending on the value of PARTOPT.
FREQ	Default = <i>dmp</i> ; if NUMDOM has any other value, it will automatically be set to <i>dmp</i> (= no. of processors used for the run). The frequency range will be divided into NUMDOM regions which are then solved independently.
UPFACT (ACMS)	By default, the frequency range used for upstream component modes is two times larger than the desired range on the EIGR/L entry. To modify this factor, specify the UPFACT parameter (real; default=2.0).
TREE	Specifies the type of elimination tree to use:
(ACMS)	TREE = MULTI: binary multilevel tree (default) TREE = SINGLE: single level tree
	A multilevel tree requires that the number of domains be a power of two and be evenly divisible by <i>dmp</i> , the number of processors specified for the run; otherwise a single level tree will be used automatically.
ALLOC	Determines how superelements are assigned to processors:
(ACMS)	ALLOC = STATIC: components are divided evenly at job start (default)

ALLOC = DYNAMIC: components are assigned to processors on a first come, first served basis. This option is effective when one or more processor is heavily loaded and/or has a slower computational rate.

PRINT
(ACMS)

Controls intermediate print of upstream and data recovery processing in F06 and F04 files. Default='NO'. If PRINT=NO and an error occurs upstream, the intermediate output is placed in a separate output file named "jid.acms_out" for examination.

Table 13-30 shows the availability of partitioning methods with each analysis type. An asterisk ("*") indicates a supported implementation.

Table 13-30 Analysis Types and Partitioning Methods

Solution Sequence	DMP Method	Partitioning Methods Available				
		GEOM	DOF	FREQ	G+F	D+F
101	STAT	*	*			
103	MODES	*	*	*	*	
	ACMS	*				
108	FREQ			*		
111	MODES		*	*		*
	ACMS	*		*	*	
	FREQ			*		
112	MODES		*	*		
	ACMS	*				
200	MODES		*	*		*
	FREQ			*		

The DOMAINSOLVER command is optional. If "dmp=" is specified on the command without a DOMAINSOLVER command in the Executive Section, the following actions will result based on solution sequence.

Table 13-31 DOMAINSOLVER Defaults

Solution Number	Default DOMAINSOLVER Options	
	DMP Method	Partitioning Option
101	STAT	GRID
103	MODES	GRID
108	FREQ	FREQ
111	MODES	FREQ
	FREQ	FREQ
112	MODES	FREQ
200	MODES	DOF
	FREQ	FREQ

13.11.4 Method Selection Guidelines

The modeling and analysis of real world phenomena is a complicated process. No single numerical approach satisfies every situation. Multiple solutions, each more or less tailored for a particular modeling scenario, computer architecture, or analysis type, are available in MSC.Nastran. Only the largest and most challenging problems are suitable for Distributed Memory Parallel (DMP) methods. This section provides guidelines for the most productive use of DMP. It is organized by finite element analysis type.

Static Analysis

- Models larger than 20,000 grid points may show parallel speedup using DMP. Larger problems may benefit more.
- Nontrivial models with dozens or hundreds of load cases would exhibit speedup.
- Large solid models would also benefit from the MSC.Nastran iterative solution. This is especially true if there are only a few load cases.

Normal Modes Analysis

- For large problems in SOL 103 with only a few modes desired, the geometric domain parallel Lanczos may be preferred.
- For problems in SOL 103 with a large number of modes desired:

- Frequency domain decomposition will be effective if the modes are fairly evenly distributed throughout the frequency range. This is especially true if the computer architecture is truly distributed, i.e. each processor has its own memory and scratch disk space.
- For very large problems ACMS is effective.
- If disk space is limited, any of the geometric domain decomposition methods (parallel Lanczos, Hierarchical, ACMS) will help minimize scratch disk space requirement.

Modal Dynamic Analysis (SOLs 111 and 112)

In modal dynamic analysis, two levels of parallelism are possible. One is in the eigensolution, and the other is in the dynamic analysis. Parallelization of the frequency response module is automatically activated in SOL 111. Parallelization of the transient response module in SOL 112 is not supported at this time.

- For large cases over wide frequency ranges, with multiple dynamic loads, ACMS is the preferred DMP method.
- The DOF domain decomposition method is generally effective.
- Frequency domain decomposition in the real eigensolution will be effective if the modes are fairly evenly distributed throughout the frequency range. This is especially true if the computer architecture is truly distributed, i.e. each processor has its own memory and scratch disk space.

Direct Frequency Response (SOL 108)

- If sufficient resources are available to solve the physical problem on each processor, DMP is a very effective way to provide parallel speedup in direct frequency response (SOL 108).

13.11.5 Limitations

This section lists limitations and unsupported features in DMP.

The following limitations apply to all DMP methods:

1. ASSIGN statements are supported only in a truly distributed environment, and if separate scratch disks are assigned to each machine for the DMP run. The limitation does not apply to USER FORTRAN files (e.g., output2, output4, xdb, etc.)
2. P-elements are not supported.

Following is a list of additional limitations that are present only for the ACMS method:

1. In acoustic analysis, MPCs or rigid elements cannot be connected between the fluid and the structure.

Following is a list of additional limitations that apply only to the MODES method:

1. Acoustic analysis (fluid-structure interaction) is not supported for the geometric domain. It is supported in the DOF domain (PARTOPT=DOF).
2. Disjoint models are not allowed.
3. Scalar Points are not supported.
4. Virtual mass (MFLUID) is supported. However, if the partitioner assigns the virtual mass to the boundary, System Fatal Message 7795 will be issued. In this case, switch to the ACMS method, or use frequency domain decomposition.
5. Residual vectors are not available with geometric domain decomposition (PARTOPT=GRID).

Following is a list of additional limitations that apply only to static analysis (SOL 101):

1. Disjoint models are not allowed.
2. Scalar points are not supported.
3. Virtual mass (MFLUID) is not supported.

The following limitations apply to the Geometric Domain option:

1. Duplicate element IDs are not supported.
2. Restart is only supported for ACMS with one processor.
3. Dummy elements (CDUM8, CDUM9) are not supported.
4. User-defined superelements are not supported.
5. Large numbers of dependent degrees of freedom on MPCs or rigid elements may result in uneven partitions which may increase runtime.
6. Acoustic analysis in SOL 111 (fluid-structure interaction) is supported only in the ACMS method.
7. Results from the grid point weight generator are not correct.

There are no known limitations for the Frequency Domain option.

13.11.6 Examples

MSC.Nastran DMP requires the use of the `dmp` keyword on the `nastran` submittal line. In addition, the `DOMAINSOLVER` executive statement is required in most cases, unless the default options are desired. Below is a series of examples with the `nastran` submittal line in italics.

Example 1 -- SOL 101

The following SOL 101 example uses geometric domain decomposition with four processors. The domain decomposition algorithm divides the model into four domains which will be solved independently, followed by the distributed boundary solution.

```
nastran jobname dmp=4 ...

SOL 101
DOMAINSOLVER STAT (PARTOPT=GRID)
CEND
...
ENDDATA
```

This is equivalent to:

```
SOL 101
DOMAINSOLVER STAT
CEND
...
ENDDATA
```

and is equivalent to:

```
SOL 101
CEND
...
ENDDATA
```

This problem will be divided into four components, plus a common boundary, based on grid point (geometric) domain decomposition. Each processor will operate on its local domain independently in parallel. Solution at the boundary proceeds via inter-process communication. Since `GRID` is the default `PARTOPT` for the `STAT` method, it is not necessary to specify it explicitly. The `DOMAINSOLVER` statement is not needed for SOL 101, since it is the default.

Example 2 -- SOL 103 with Frequency Domain Decomposition

Run a normal modes analysis using frequency domain decomposition on two processors.

```
nastran jobname dmp=2 ...

SOL 103
DOMAINSOLVER MODES (PARTOPT=FREQ)
CEND
...
ENDDATA
```

In this example, the frequency range specified on the EIGR/L Bulk Data entry will be divided into two sub-regions. Each processor will solve the full eigenvalue problem for its own local frequency domain. Results are merged on the master processor and the analysis continues.

Example 3 -- SOL 103 with Geometric Domain Decomposition

Run a normal modes analysis using geometric domain decomposition on four processors.

```
nastran jobname dmp=4 ...

SOL 103
DOMAINSOLVER MODES (PARTOPT=GRID)
CEND
...
ENDDATA
```

This is equivalent to:

```
SOL 103
DOMAINSOLVER MODES
CEND
...
ENDDATA
```

and is equivalent to:

```
SOL 103
CEND
...
ENDDATA
```

This problem will be divided into four components, plus a common boundary, based on grid point (geometric) domain decomposition. Each processor will perform a complete eigenvalue analysis on the full frequency range for its local domain. Solution at the boundary follows. Then, the results are merged on the master

processor and the analysis continues. Since GRID is the default PARTOPT for the MODES method, it is not necessary to specify it explicitly. The DOMAINSOLVER statement is not needed for SOL 103 since MODES is the default describer.

Example 4 -- SOL 103 with the Hierarchical Method

This example executes a normal modes analysis using the Hierarchical combination DMP method of geometric and frequency domain decomposition with 16 processors.

```
nastran jobname dmp=16 ...

SOL 103
DOMAINSOLVER MODES (PARTOPT=GRID, NUMDOM=4)
CEND
...
ENDDATA
```

This is equivalent to:

```
SOL 103
DOMAINSOLVER MODES (NUMDOM=4)
CEND
...
ENDDATA
```

In this example, the model is divided into four components using geometric domain decomposition (the “outer layer” of the domain hierarchy). During the eigenvalue analysis (in the READ module), each of the four components is further divided in the frequency domain into four frequency segments. Each frequency segment is solved independently within its geometric domain.

Example 5 -- SOL 111 with ACMS

Run a modal frequency analysis (SOL 111) using ACMS with four processors.

```
nastran jobname dmp=4 ...

SOL 111
DOMAINSOLVER ACMS (PARTOPT=GRID)
CEND
...
ENDDATA
```

This is equivalent to:

```
SOL 111
DOMAINSOLVER ACMS
CEND
...
ENDDATA
```

This is equivalent to:

```
SOL 111
DOMAINSOLVER ACMS FREQ(NUMDOM=4)
CEND
...
ENDDATA
```

In this example, the model will be divided in the geometric (grid point) domain into a default number of components, based on the size of the model (see the `DOMAINSOLVER` command description for details). A binary multilevel superelement tree will be constructed and a component modal synthesis will be executed in parallel, with the tree logically divided among the four processors. For the frequency response portion of the run, the frequency range will be divided into four segments, and each segment will be computed independently, in parallel. (This is the default action.) Since geometric (grid point) domain decomposition is the default partitioning option, it need not be specified on the `DOMAINSOLVER` statement.

Example 6 -- SOL 108

Run a direct frequency response analysis (SOL 108) with eight processors.

```
nastran jobname dmp=8 ...

SOL 108
DOMAINSOLVER FREQ (PARTOPT=FREQ)
CEND
...
ENDDATA
```

This is equivalent to:

```
SOL 108
DOMAINSOLVER FREQ
CEND
...
ENDDATA
```

and is equivalent to:

```
SOL 108
CEND
...
ENDDATA
```

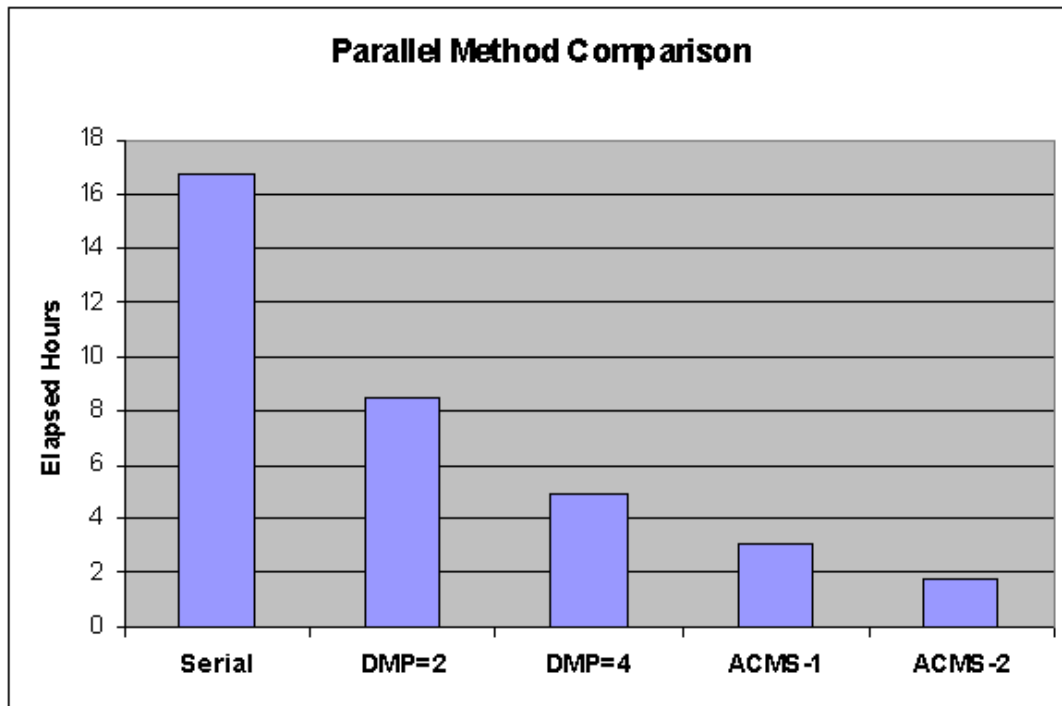
In this example, the frequency list is logically divided among eight processors, and each processor proceeds independently in parallel. Results from each processor are merged onto the master processor, and the analysis continues. Since `FREQ` is the

default PARTOPT for the FREQ method, it is not necessary to specify it explicitly. The DOMAINSOLVER statement is not needed for SOL 108 since FREQ is the default describer.

13.11.7 Parallel Performance Data

Parallel performance data for an automotive model is covered in this section. This is a SOL 103 run with a model consisting of 1.8 million DOFs and calculating 996 modes. The system configuration is shown below.

- Linux cluster
 - Four nodes each with two 2.0 GHz P4 CPUs.
 - 250 Gb striped scratch disk per node (85 mb/sec I/O).
 - 3 Gb main memory per node, 100 M/s interconnect.



13.11.8 Other Numeric Enhancements

Two other numeric enhancements have been added to MSC.Nastran 2004. They are Sparse Cholesky Solution and MPYAD Method 1 Storage 3 Option.

Sparse Cholesky Solution

The DCMF and FBS modules are enhanced to take advantage of the sparse Cholesky factor.

- DCMP modifications were introduced to create a special sparse Cholesky factor. The factor trailer identifier is form=14. The creation of the Cholesky factor may be requested by CHOLSKY input parameter to the DCMP module. This new method is selected by setting system cell 385 = 1 on the command line (sys385=1), in the rcfile (nastran system (385) = 1), or as a nastran statement in the input file. This is advisable for increased performance during large normal modes analyses using the Householder method.
- FBS module was modified to take advantage of the special sparse Cholesky factor. No user intervention is required as the form=14 of the special factor trailer triggers the Cholesky solve.

New MPYAD Method 1 Storage 3 Option

A new storage option has been added to the matrix multiply and add module MPYAD. Method 1 Storage 3 is selected under the following conditions:

- The non null portion of the A matrix is at least 50% dense
- The non null columns of A fit in available memory
- At least two columns of B and D fit in memory
- All matrices are real

The non null portion of A is multiplied with the non null portion of B using level-3 BLAS (Basic Linear Algebra Subroutine) kernels.

The new option will be automatically selected based on time estimates. The selection or deselection of the new option may be accomplished by nastran system cell system(66). The identifiers associated with the new option are:

Method	System Cell Value
Method 1, Storage 3, NT	System(66) = 67108864
Method 1, Storage 3, T	System(66) = 134217728

Note: The new storage 3 option only supports the real matrix case.

13.12 MSC.Nastran/MSC.ADAMS Interface

Introduction

MSC.Nastran allows the use of residual structure only or any super element or any part super element to be used as a component for an ADAMS/Flex flexible body. Flexible body analysis, including large rigid body motions, are available by combining the ADAMS/Flex formulation with reduced flexible component dynamics provided by MSC.Nastran. This allows designers take a component-level Finite Element Analysis and move its characteristic shapes to a system-level motion study for improved design accuracy.

Typical applications are: Elasto-kinematic analysis of vehicular suspensions, engine crankshaft-drivetrain interaction, aircraft landing conditions, satellite panel deployment.

Benefits

The MSC.Nastran/ADAMS integration provides an easy method to move from a FE analysis to a system analysis study by providing the direct generation within MSC.Nastran of the MSC.ADAMS Modal Neutral File (MNF) required for the ADAMS/Flex solver.

The interface is initiated by the simple MSC.Nastran Case Control command ADAMSMNF FLEXBODY=YES and the addition of a Bulk Data entry, DTI,UNITS. The interface is applicable in SOL 103, SOL 111, SOL 112, or SOL 103 from a nonlinear SOL 106 restart using standard MSC.Nastran PARAM,NMLOOP.

The user can easily define the flexible body attachment points in MSC.Nastran by either defining the component as a super element or part super element, in which case, the physical external (a-set) grids become the attachment points; or for a residual only type model, using standard MSC.Nastran ASET Bulk Data entries to define the attachment points.

If the component requires prestiffening which does not entail large deformation standard MSC.Nastran Case Control STATSUB can be used in SOL 103, SOL 111, or SOL 112 to provide the required loading.

For a component that will experience nonlinear deformation a standard SOL 106 run can be made and then using standard PARAM,NMLOOP the component can be restarted into SOL 103 with advantage that ASET Bulk Data entries can be used at restart time to define the attachment points.

The component modes determined by MSC.Nastran automatically represent a set of constraint modes for boundary coordinates plus a truncated set of fixed-interface normal modes; this combination is referred to as the Craig-Bampton modes. These modes, by default, are augmented by residual flexibility modes to give a correct representation of spatial loads. The spatial loads may include physical loads and inertia relief loads.

Theory

Basic Component Mode Equations

This section briefly describes the equations used in MSC.Nastran/MSC.ADAMS interface.

In MSC.Nastran modal solutions, the physical nodal displacements $\{u\}$ are related to modal amplitudes $\{\zeta\}$ by the relationship:

$$\{u\} = [\Upsilon]\{\zeta\} \quad \text{Eq. 13-105}$$

where $[\Upsilon]$ is the matrix of mode shapes containing the Craig-Bampton modes and any residual flexibility modes

The above transformation is then applied to the physical mass, stiffness, and load matrices. The resulting matrices are then reduced to analysis size (a-set) matrices. For modal analysis, the a-set consists of the modal amplitudes and the physical boundary grids external to a superelement or the modal amplitudes and the physical grids defined using an ASET Bulk Data entry. The resulting form, symbolically using the stiffness matrix as an example is:

$$[K_{aa}] = [\Upsilon_{ga}]^T [K_{gg}] [\Upsilon_{ga}] \quad \text{Eq. 13-106}$$

with appropriate similar result for the mass and load matrices.

To isolate rigid body modes as well as any other mechanisms or poorly-conditioned modes and to insure orthonormalization, an a-set eigenvalue problem is solved. This results in a new matrix of shapes $[\Phi_{a\lambda}]$ where the λ may be equal or fewer than the initial a-set modes. The modal stiffness, modal mass, and modal load a-set matrices are then transformed as:

for stiffness

$$[K_{\lambda\lambda}] = [\Phi_{a\lambda}]^T [K_{aa}] [\Phi_{a\lambda}] \quad \text{Eq. 13-107}$$

for modal mass

$$[M_{\lambda\lambda}] = [\Phi_{a\lambda}]^T [M_{aa}] [\Phi_{a\lambda}] \quad \text{Eq. 13-108}$$

and for modal loads

$$[P_{\lambda}] = [\Phi_{a\lambda}]^T [P_a] \quad \text{Eq. 13-109}$$

Additionally, a diagonal matrix of a-set unit boundary displacements is formed and taken through standard MSC.Nastran transformations up to g-set size. This matrix, say $[\Xi_{ga}]$, is transformed as:

$$[\Xi_{g\lambda}] = [\Xi_{ga}] [\Phi_{a\lambda}] \quad \text{Eq. 13-110}$$

The above four orthonormal matrices and the physical mass matrix are then exported to MSC.ADAMS in MNF format. The matrix $[\Xi_{g\lambda}]$ is also used to obtain stress and strain shapes.

Mass Invariant Calculations

When executing the MSC.Nastran/ADAMS interface, MSC.Nastran will export to MSC.ADAMS up to nine mass invariants. Briefly, these invariants are obtained as follows.

The instantaneous location $\{r_p\}$ of a grid point p can be expressed as:

$$\{r_p\} = \{x\} + [A]\{s_p\} + [A]\{u_p\} \quad \text{Eq. 13-111}$$

where

$\{x\}$ is the position vector from the origin of a ground reference frame to the origin of a reference frame attached to the flexible body.

$[A]$ is the direction cosine matrix between the ground reference and the reference frame attached to the flexible body.

$\{s_p\}$ is the position vector from the origin of the reference frame attached to the flexible body to the undeformed position of a grid point p. $[A]\{s_p\}$ are the components of $\{s_p\}$ in the ground reference frame.

$\{u_p\}$ is the translational deformation of the grid point p in the reference frame attached to the flexible body. $[A]\{u_p\}$ are the components in the ground reference frame.

The velocity $\{v_p\}$ is defined as:

$$\{v_p\} = \{x\} + [A](s_p + u_p) + [A]\{\dot{u}_p\} \quad \text{Eq. 13-112}$$

Since the columns of $[A]$ are unit vectors along the coordinate axes of the reference frame attached to the flexible body, we can use the angular velocity $\{\Omega\}$ of the reference frame attached to the flexible body relative to the ground reference to determine $[A]$. We get, expressing the results in a body fixed system:

$$[\dot{A}] = [A][\tilde{\Omega}] \quad \text{Eq. 13-113}$$

where $[\tilde{\Omega}]$ is a skew symmetric matrix containing the components of the angular velocity.

The generalized coordinates of the flexible body are defined as:

$$\{\xi\} = \begin{bmatrix} \{x\} \\ \{\psi\} \\ \{q\} \end{bmatrix} \quad \text{Eq. 13-114}$$

where $\{\psi\}$ is a body fixed 3-1-3 set of Euler angles, $\{q\}$ is the vector of modal coordinates used to define the generalized coordinates of the flexible body, and $\{x\}$ is defined above.

Then $\{\Omega\}$ can be defined as:

$$[\Omega] = [B]\{\psi\} \quad \text{Eq. 13-115}$$

where $[B]$ is the matrix of sines and cosines representing the Euler angles.

We can define the following transformations:

$$[u_p] = [\phi_p]\{q\} \quad \text{Eq. 13-116}$$

$$[\dot{u}_p] = [\phi_p]\{\dot{q}\} \quad \text{Eq. 13-117}$$

$$[\omega_p] = [B]\{\psi\} + [\phi_p^*]\{\dot{q}\} \quad \text{Eq. 13-118}$$

where $[\phi_p]$ is the partitioned orthogonal modal matrix that corresponds to the translational degrees of freedom of grid p and $[\phi_p^*]$ is the partitioned orthogonal modal matrix that corresponds to the rotational degrees of freedom of grid p.

Then the velocity $\{v_p\}$ can be expressed as:

$$\{v_p\} = \begin{bmatrix} I & -A\tilde{X}B & A\phi_p \end{bmatrix} \{\xi\} \quad \text{Eq. 13-119}$$

where

$$[\tilde{X}] = [\tilde{s}_p] + [\tilde{u}_p] \quad \text{Eq. 13-120}$$

is a skew symmetric matrix of coordinate locations and translations.

The kinetic energy of the flexible body is then approximated by the relationship:

$$T = \frac{1}{2} \sum_p m_p [v_p] \{v_p\} + \frac{1}{2} \sum_p [w_p] I_p \{w_p\} \quad \text{Eq. 13-121}$$

where $[\]$ represent a row matrix and I_p is the inertia tensor of grid p.

It is this approximation to the kinetic energy and the need to keep the $[A]$ matrix out of the mass invariant calculations that currently restricts the MSC.Nastran/ADAMS interface analysis to a lumped mass formulation.

Expanding the relationship for kinetic energy, we obtain the following nine mass invariants:

$$\begin{matrix} 1 \\ 1 \times 1 \end{matrix} I = \sum_{p=1}^N m_p \quad \text{Eq. 13-122}$$

$$\begin{matrix} 2 \\ 3 \times 1 \end{matrix} I = \sum_{p=1}^N m_p s_p \quad \text{Eq. 13-123}$$

$$\begin{matrix} 3 \\ 3 \times M \end{matrix} I_j = \sum_{p=1}^N m_p \Phi_p \quad j = 1, \dots, M \quad \text{Eq. 13-124}$$

$$\begin{matrix} 4 \\ 3 \times M \end{matrix} I = \sum_{p=1}^N m_p \tilde{s}_p \Phi_p + I_p \Phi_p^* \quad \text{Eq. 13-125}$$

$$\begin{matrix} 5 \\ 3 \times M \end{matrix} I_j = \sum_{p=1}^N m_p \tilde{\phi}_{pj} \Phi_p \quad j = 1, \dots, M \quad \text{Eq. 13-126}$$

$$\begin{matrix} 6 \\ M \times M \end{matrix} I = \sum_{p=1}^N m_p \Phi_p^T \Phi_p + \Phi_p^{*T} \mathbf{I}_p \Phi_p^* \quad \text{Eq. 13-127}$$

$$\begin{matrix} 7 \\ 3 \times 3 \end{matrix} I = \sum_{p=1}^N m_p \tilde{s}_p^T \tilde{s}_p + \mathbf{I}_p \quad \text{Eq. 13-128}$$

$$\begin{matrix} 8 \\ 3 \times 3 \end{matrix} I_j = \sum_{p=1}^N m_p \tilde{s}_p \tilde{\phi}_{pj} \quad j = 1, \dots, M \quad \text{Eq. 13-129}$$

$$\begin{matrix} 9 \\ 3 \times 3 \end{matrix} I_{jk} = \sum_{p=1}^N m_p \tilde{\phi}_{pj} \tilde{\phi}_{pk} \quad j, k = 1, \dots, M \quad \text{Eq. 13-130}$$

where $s_p = [xyz]^T$ are the coordinates of grid point p in basic and

$$\tilde{s}_p = \begin{bmatrix} \mathbf{0} & -z & y \\ z & \mathbf{0} & -x \\ -y & x & \mathbf{0} \end{bmatrix}$$

is the skew-symmetric vector cross product operator. Φ_p is the partitioned orthogonal modal matrix that corresponds to the translational degrees of freedom of grid p , \mathbf{I}_p is the inertia tensor of grid p , Φ_p^* is the partitioned orthogonal modal matrix that corresponds to the rotational degrees of freedom of grid p . $\tilde{\phi}_{pf}$ is the skew-symmetric matrix formed for each grid translational degree of freedom for each mode. M is the number of modes and N is the number of grids.

Inputs

If MSC.Nastran/ADAMS interface is desired, only **two** unique **entries** are **required**. They are:

Case Control Unique to ADAMS/Flex Interface

ADAMSMNF FLEXBODY=YES which causes the ADAMS/Flex MNF to be generated.

Bulk Data Entries Unique to ADAMS/Flex Interface

DTI,UNITS used to specify the unit system to be used by ADAMS/Flex.

Typical File Management for ADAMS/Flex Interface

- ASSIGN MASTER='location_of_106_database' - used in a SOL 103, SOL 111, or SOL 112 run to locate the SOL 106 data base for restart.
 - RESTART - used in a modal solution to indicate that run is a restart for a SOL 106 run. This is a standard MSC.Nastran restart.
 - DBLOC - used in a modal solution run to extract files from a SOL 106 data base. This is used for a nonstandard MSC.Nastran restart.

Typical Case Control for ADAMS/Flex Interface

- ADAMSMNF FLEXBODY = YES -- **required for interface run.** See the ADAMSMNF case description for a complete list of options and interface rules.
- STRESS(PLOT) -- necessary for stress shapes.
- GPSTRESS(PLOT) -- necessary for grid stress shapes to be included in MNF.
- STRAIN(PLOT) - necessary for strain shapes.
- GPSTRAIN(PLOT) -- necessary for grid strain shapes to be included in MNF.
- METHOD = n -- required above or in first subcase for modal solutions.
- RESVEC = character_value -- controls calculation of residual flexibility (including inertia relief) modes. SOL 111 and SOL 112, residual flexibility is defaulted to on for both component (o-set) and system modes (a-set). In SOL 103, residual flexibility is defaulted to on for only component modes (o-set).
- LOAD = n -- used if modal loads are desired. A different subcase will define different load conditions.
- STATSUB(PRELOAD) = subcase_ID -- necessary if a linear preload condition is to be included in the MNF.
- SUPORT1=seid -- necessary to select a static support set for a residual only linear preload run.

- SPCF(PLOT) = ALL -- necessary with preload to insure complete preload reaction condition is to be included in the MNF.
- MPCF(PLOT) = ALL -- necessary with preload to insure complete preload reaction condition is to be included in the MNF.
- OUTPUT(POST) - -necessary to define volume and surface for grid stress or strain shapes.
 - SET nnn = list -- set of elements for surface definition for grid stress or strain shapes.
 - SURFACE nnn SET nnn NORMAL z3 -- surface definition - Only one FIBER selection is allowed for all SURFACES.
 - VOLUME nnn SET nnn -- volume definition.
 - The default SYSTEM BASIC is required with SURFACE or VOLUME.
- OUTPUT(PLOT) -- necessary to define elements used to select grids to display the component in ADAMS.
 - SET nj = list of elements used to select grids to display the component.
- NLPARM=n -- used to select nonlinear parameters on a bulk entry NLPARM.
- PARAM,NMLOOP,m -- necessary for a SOL 103 restart from SOL 106. Describes the nonlinear iteration loop m to use for modal analysis.
- SUPER=n,SEALL=n -- useful with multiple superelement model to select an individual superelement as a flexible body. Cannot be used with a linear STATSUB(PRELOAD) run.

Typical Bulk Data for ADAMS/Flex Interface

- DTI,UNITS,1,MASS,FORCE,LENGTH,TIME -- **required for interface run.** See the ADAMSMNF case description for a complete description of ADAMS unit requirements and interface rules. For parts super elements, must reside in the main Bulk Data Section.
- SPOINT,id_list -- used to define and display modal amplitude.
- SESET,SEID,grid_list -- used to define a superelements (see GRID and BEGIN BULK SUPER=). The exterior grids will represent the attachment points along with the q-set.
- SEELT,SEID,element_list -- used to reassign superelement boundary elements to an upstream superelement.

- RELEASE,SEID,C,Gi -- optional entry that removes DOFs from an attachment grid for which no constraint mode is desired. For example, this allows the removal of rotational DOFs from an analysis where only translational DOFs are required.
- SEQSET, SEID,spoint_list -- use to define modal amplitudes of a superelement (see SEQSET1).
- SENQSET,SEID,N -- used to define modal amplitudes of a part super element. Must reside in the main Bulk Data Section.
- ASET,IDI,Ci -- used to define attachment points for a residual only run (see ASET1). This entry cannot exist in the SOL 106 run of a restart into SOL 103. It must be added in the SOL 103 restart.
- QSET1,C,IDI -- used to define modal amplitudes for the residual structure or modal amplitudes for a part super element (see QSET). This entry cannot exist in the SOL 106 run of a restart into SOL 103. It must be added in the SOL 103 restart.
- SUPORT1,SID,IDI,Ci -- used to define the static support for a preload condition with a residual only run. This entry is case control selectable. **Do Not Use SUPORT.**
- PLOTEL,EID,Gi -- can be used, along with existing model elements, to define elements used to select grids to display the component in ADAMS.
- EIGR,SID,METHOD,... -- used to obtain real eigenvalue extraction (see EIGRL).
- PARAM,GRDPNT,value -- mass invariants 1I , 2I , and 7I will be computed using results of MSC.Nastran grid point weight generator execution in the basic coordinate system.
- PARAM,LGDISP -- used to turn on large displacement theory in SOL 106.
- NLPARM,N,INC,... -- used to select nonlinear parameters.

Outputs

Outputs unique to the MSC.Nastran/ADAMS interface run is the MNF for export to the ADAMS/Flex run and if ADMOUT=YES on the ADAMSMNF Case Control, the op2 file described on the ADAMSMNF Case Control command description.

Guidelines and Limitations

The following guide lines are associated with the capability:

1. Superelements and parts super elements require a superelement license.

2. Lumped mass formulation (default) is required. Either leave PARAM,COUPMASS out of the input file or supply PARAM,COUPMASS,-1 (default) to insure lumped mass.
3. p-elements are not allowed as they are always coupled mass formulation.
4. If for preload conditions a simple static support system and a self equilibrating load can be used, the SPCF(PLOT) and MPCF(PLOT) statements are not required.
5. For nonlinear SOL 106 runs, if the MATS1 entry is included, an incremental approach between MSC.Nastran and ADAMS/Flex should be adopted because of the path dependent load history.
6. For a standard MSC.Nastran restart (using RESTARTF File Management statement) from a SOL 106 (nonlinear) in to SOL 103 (modal with MSC.Nastran/ADAMS interface) the SPOINTS must be in the SOL 106 run and the associated QSET1s must be in the SOL 103 run.
7. If a user has legacy SOL 106 runs or forgets to put the SPOINTS in the SOL 106 run, the SPOINTS and associated QSET1s can be placed in the modal run which then must use a nonstandard restart using the DBLOCATE file management command. The command **must contain** the following six data blocks: bulk, estnl, ugni, qgni, qmgni, pgni. The command must also have a dbname that corresponds to the dbname on the ASSIGN command. The command takes the form:
 - dblocate datablk=(bulk,estnl,ugni,qgni,qmgni,pgni) where (version=1), logi=dbname
8. When **restarting** with a RESTART or DBLOCATE commands from a SOL 106 run into a modal run (SOL 103, SOL 111, SOL 112) **the first SUBCASE in the modal run must contain the LOAD= entry** that reflects the load condition of the chosen converged nonlinear solution. Other SUBCASES with their own load conditions may follow the first SUBCASE to be used for generation of modal load shapes. This is required for both labeling reasons and proper column alignment.
9. When supplying SPOINTS/QSETS combinations there should be enough supplied to capture accurately the modal shapes. If n is the number of modes specified on the EIGR or EIGRL Bulk Data entries and p is the number of load cases specified, then the number of SPOINTS(ns) should at least be $ns = n + (6 + p)$ assuming that residual flexibility is on. In general, there cannot be too many SPOINTS as excess ones will be simply truncated with no performance penalty.

10. For FLEXBODY=YES runs, residual vectors for the component should always be calculated as they result in very accurate representation of the component shapes at little additional cost.
11. The current ADAMS/Flex does not support OMIT or OMIT1 Bulk Data entries.
12. PLOTELS or other elements may be used to define FE mesh detail for ADAMS/Flex using the PSETID=set_entry on the ADAMSMNF Case Control command. With superelement analysis, for any of these elements that lie entirely on the superelement boundary (all of the elements grids attached only to a-set or exterior grids), a SEELT Bulk Data entry MUST be specified to keep that display element with the superelement component. The SEELT entry would not be required with parts superelements as boundary elements stay with their component.
13. PARAM,WTMASS,value with a value other than 1.0 may be used with an MSC.Nastran/ADAM/Flex run when generating a MNF. It must have consistent units with regard to the DTI,UNITS Bulk Data entry as discussed on the ADAMSMNF Case Control command. Before generating the MNF, MSC.Nastran will appropriately scale the WTMASS from the physical mass matrix and the mode shapes.
14. There are two types of loads in MSC.Nastran; those that are element specified (such as a PLOAD4 entry) and those that are grid specific (such as a FORCE entry). When doing a standard MSC.Nastran run, the distinctions not important. When doing an ADAMSMNF FLEXBODY=YES run there is a subtle distinction when superelements are used. Element specified type loads always stay with the superelement. The superelement sees the total element specified applied load. For grid specified loads, any grid specified load attached to an external grid will move down stream with the grid. That is to say, it is part of the boundary and not part of the superelement (or ADAMS/Flex component). This distinction applies only to a superelement run and not to a residual only or a parts super element run.
15. The loads specified in MSC.Nastran in general fall into two categories: non follower or fixed direction loads (non circulatory) and follower loads (circulatory). The follower loads are nonconservative in nature. Examples of a fixed direction load is the FORCE entry or a PLOAD4 entry when its direction is specified via direction cosines. Examples of a follower load is the FORCE1 entry or the PLOAD4 entry when used to apply a normal pressure. By default in MSC.Nastran, the follower loads are always active in SOL 103 and SOL 106 and will result in follower stiffness being added to the differential stiffness and elastic stiffness of the structure. In an ADAMSMNF

FLEXBODY=YES with superelements, if the follower force is associated with a grid description (such as a FORCE1) and the grid is external to the superelement (or ADAMS/Flex component), the follower load will move down stream with the grid. Thus, for an ADAMSMNF FLEXBODY=YES run, the down stream follower contribution to the components stiffness will be lost. This could yield poor results. This caution applies only to a superelement run and not to a residual only or a parts superelement run.

16. In the following Examples 4-8, an in plane tensioning preload is applied to the plate structure. If a non follower type load such as a FORCE entry is used as in Example 4, two rigid body modes will be lost (a rotation about the y-axis and a rotation about the z-axis). This is due to the fact that when the preload is applied, the FORCE entry, which is fixed in direction, does not rotate with the structure. Since a preload condition requires at least a set of static supports, the resulting internal stress system used to form the differential stiffness reflects this support state and when added the elastic stiffness will add stiffness to two rotational rigid body modes. The FORCE1 entry used in Examples 5-8 are follower loads and are applied in such a manner as to rotate with the structure. The differential stiffness still has the constraints discussed for Example 4, however, for these examples a corrective follower stiffness is added which cancels out the differential stiffness and the component sees six rigid body modes.
17. For the MSC.Nastran 2004 release, the ADAMS/Flex tool kit does not support the combination of both preload and stress or strain shapes. Therefore, when using preload do not include GPSTRESS or GPSTRAIN Case Control commands.
18. For MSC.Nastran 2004, to insure compatibility with the MSC.Adams msc2mnf tool kit, if the ADAMSMNF Case Control command has the keyword ADMOUT=YES, the MSC.Nastran SYSTEM word OP2NEW is automatically set to OP2NEW=0. This means that any output2 files generated will have a pre-MSC.Nastran 2004 format.

Examples

Eight small examples are discussed below that are intended to show salient features of the MSC.Nastran/ADAMS interface with the following options:

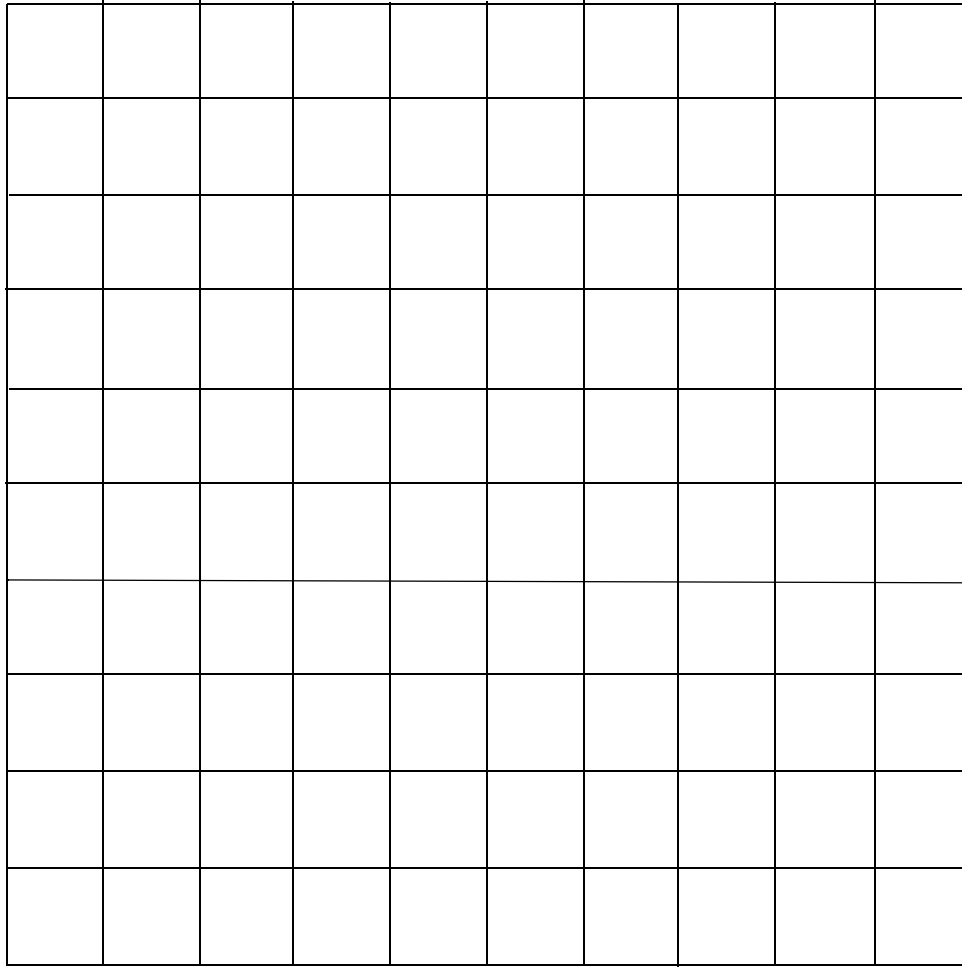
1. Flexible body component as a superelement.
2. Flexible body component as a residual only run.
3. Flexible body component as a part superelement.
4. Flexible body component with preload as a superelement -- no follower load.

5. Flexible body component with preload as a residual only run -- with follower load.
6. Flexible body component with preload as a part superelement -- with follower load.
7. A structural component with nonlinear effects run in SOL 106 and restarted into SOL 103 as a flexible body component using standard restart -- with follower load.
8. A structural component with nonlinear effects run in SOL 106 and restarted into SOL 103 as a flexible body component using nonstandard dblocate restart -- with follower load.

The basic model is a square plate divided into a 10 x 10 mesh. The four corner points, grid point 1 (.0,.0,.0), grid point 11 (0.,1.,0.), grid point 111 (1.,0.,0.), and grid point 121 (1.,1.,0.) are considered the attachment points. The basic geometry, material, and loads are given by the include file shown at the end of the examples. The Case Control commands and Bulk Data entries shown in the examples try to highlight features important to the interface run.

11

121



1

111

Ex:1 Component Modeled as a Superelement

This example represents a MSC.Nastran/ADAMS interface run with the component modeled as superelement 200 as defined by the SESET Bulk Data entries. Grids listed on this entry represent the component, grids not listed (in this case grid point 1, grid point 11, grid point 111, and grid point 121) are exterior to the superelement and are used as the attachment points for ADAMS.

The ADAMSMNF Case Control command initiates the interface run with the **required** FLEXBODY=YES entry. Also include on this entry is the optional PSETID=7772 which provides a set of PLOTTEL elements whose grids are retained in the MNF and whose connectivity defines face geometry for ADAMS display. The OUTPUT(PLOT) section

is used to define the set of elements for the PSETID. For this case, PLOTTELs were used and these are defined at the end of the bulk section. The SEELT is used so that the plotels defined on the boundary stay with the component.

Grid point stress and strain shapes are desired, thus the sequence of STRESS, STRAIN, GPSTRESS, GPSTRAIN commands. The (PLOT) item suppresses output of these items to the f06 file. The OUTPUT(POST) section is used to define the surface for the grid point stress and strain requests.

The DTI,UNITS in the Bulk Data Section is **required** for the interface run. This defines the units to be used by ADAMS/Flex.

Standard SUBCASE is used to define two pressure load conditions. No constraints are applied to the component, hence the modal analysis will represent a free structure.

The PARAM,WTMASS is not recommended with interface runs and this is noted by explicitly setting it to its default value of 1.

The PARAM,GRDPNT will cause the MSC.Nastran grid point weight generator (GPWG) to run and display a summary of the mass properties of the model. Any physical grid location may be chosen for the display purpose, however with the ADAMS interface if PARAM,GRDPNT is present, GPWG will be run a second time **always** at the origin of the basic system and its results will be used for mass invariants 1I , 2I , and 7I computations for MNF export.

The SPOINT and SEQSET1 Bulk Data entries are used to define the component modes. These entries must specify enough degrees of freedom or modal amplitudes to capture the shape of the component and residual flexibility for any loading conditions.

```
SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
```

```

RESVEC = COMPONENT
$
$ Generate stress and strain grid shapes
$
STRESS(PLOT) = ALL
STRAIN(PLOT) = ALL
GPSTRESS(PLOT) = ALL
GPSTRAIN(PLOT) = ALL
$
SUBCASE 5
LABEL= one atu overpressure - uniform
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
load =50
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
$ Define surface for stress and strain grid shapes
$
OUTPUT(POST)
SET 9998 = ALL
SURFACE 9998 SET 9998 NORMAL X3
$
BEGIN BULK
$
$ ADAMS REQUIRES following DTI
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Turn on gridpoint weight generator
$
PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the above DTI,UNITS
$
PARAM,WTMASS,1.0
$
$ Select number of modes:
$
$ =====
EIGR      300      LAN                      10
$ =====
$
$ Define interior grids as superelement 200
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
SESET,200,2,THRU,10
SESET,200,12,THRU,110
SESET,200,112,THRU,120

```

```

$
$ SCALAR Point and SEQSET1 to define DOFs to use for component modes
$
SPOINT,80001,THRU,80018
SEQSET1,200,0,80001,THRU,80018
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
$ SEELT to put element 1003 and 1004 into SE 200
$
SEELT 200 10003 10004
$
$ Get model data and load data
$
include 'modell1.dat'
$
ENDDATA

```

Ex:2 Component Modeled as a Residual Only Run

This example represents a MSC.Nastran/ADAMS interface run with the component modeled as residual only structure. Grid point 1, grid point 11, grid point 111, and grid point 121 are to be used as the attachment points for ADAMS. This is accomplished by use of the ASET1 Bulk Data entry.

The ADAMSMNF Case Control entry initiates the interface run with the **required** FLEXBODY=YES entry. Also include on this entry is the optional PSETID=7772 which provides a set of PLOTTEL elements whose grids are retained in the MNF and whose connectivity defines face geometry for ADAMS display. The OUTPUT(PLOT) section is used to define the set of elements for the PSETID. For this case, PLOTTELS were used and these are defined at the end of the Bulk Data Section.

For this run, it is desired to generate in addition to the MNF for export to ADAMS, op2 files for export into MSC.Fatigue. This op2 files generation is initiated by the ADMOUT=YES entry the ADAMSMNF Case Control command. The ASSIGN statement in the MSC.Nastran File Management Section will cause the op2 file to be written with the file name residual_op2.out to fortran unit 20.

The additional OUTSTRS=YES on the ADAMSMNF Case Control command will cause element stress shapes to be output to the op2. This requires the STRESS command in Case Control command.

Grid point stress and strain shapes are desired for export to the MNF, thus the sequence of STRESS, STRAIN, GPSTRESS, GPSTRAIN commands. The (PLOT) item suppresses output of these items to the f06 file. The OUTPUT(POST) section is used to define the surface for the grid point stress and strain requests.

The DTI,UNITS in the Bulk Data Section is **required** for the interface run. This defines the units to be used by ADAMS/Flex.

Standard SUBCASE is used to define two pressure load conditions. No constraints are applied to the component, hence the modal analysis will represent a free structure.

The PARAM,WTMASS is not recommended with interface runs and this is noted by explicitly setting it to its default value of 1.

The PARAM,GRDPNT will cause the MSC.Nastran grid point weight generator (GPWG) to run and display a summary of the mass properties of the model. Any physical grid location may be chosen for the display purpose, however with the ADAMS interface if PARAM,GRDPNT is present, GPWG will be run a second time **always** at the origin of the basic system and its results will be used for mass invariants 1I , 2I , and 7I computations for MNF export.

The SPOINT and QSET1 Bulk Data entries are used to define the component modes. These entries must specify enough degrees of freedom or modal amplitudes to capture the shape of the component and residual flexibility for any loading conditions.

```
$ ADAMS/Durability with MSC.Fatigue requires op2 generation
$
ASSIGN OUTPUT2='e2_residual_op2.out' STATUS=UNKNOWN UNIT=20 FORM=UNIFORM
$
SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$ admout=yes also output op2 file for ADAMS/Durability with MSC.Fatigue
$ outstrs=yes output element stress shapes to op2 file
$
ADAMSMNF flexbody=yes, psetid=7772, admout=yes,
          outstrs=yes
$
$ Select real eigen value parameters - must be above or in first subcase
$
```

```

METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$
$ Generate stress and strain grid shapes
$
STRESS(PLOT) = ALL
STRAIN(PLOT) = ALL
GPSTRESS(PLOT) = ALL
GPSTRAIN(PLOT) = ALL
$
SUBCASE 5
LABEL= one atu overpressure - uniform
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
load =50
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
$ Define surface for stress and strain grid shapes
$
OUTPUT(POST)
SET 9998 = ALL
SURFACE 9998 SET 9998 NORMAL X3
$
BEGIN BULK
$
$ ADAMS REQUIRES following DTI
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Turn on gridpoint weight generator
$
PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the above DTI,UNITS
$
PARAM,WTMASS,1.0
$
$ Select number of modes:
$
$ =====
EIGR      300      LAN                               10
$ =====
$

```

```

$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
ASET1,123456,1,11,111,121
$
$ SCALAR Point and QSET1 to define DOFs to use for component modes
$
SPOINT,80001,THRU,80018
QSET1,0,80001,THRU,80018
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
$ Get model data and load data
$
include 'modell1.dat'
$
ENDDATA

```

Ex:3 Component Modeled as a Part Superelement

This example represents a MSC.Nastran/ADAMS interface run with the component modeled as part superelement 200 as defined by the BEGIN BULK SUPER = 200 entry. Grids listed in this section represent the complete component as a substructure (including in this case grid point 1, grid point 11, grid point 111, and grid 121). The **main bulk section** identified by BEGIN BULK includes GRID entries for grid point 1, grid point 11, grid point 111, and grid point 121 which identify the boundaries and are exterior to the part super element and are used as the attachment points for ADAMS.

The ADAMSMNF Case Control command initiates the interface run with the **required** FLEXBODY=YES entry. Also include on this entry is the optional PSETID=7772 which provides a set of PLOTTEL elements whose grids are retained in the MNF and whose connectivity defines face geometry for ADAMS display. The OUTPUT(PLOT) section is used to define the set of elements for the PSETID. For this case, PLOTTELs were used and these are defined at the end of the Bulk Data Section.

Grid point stress and strain shapes are desired, thus the sequence of STRESS, STRAIN, GPSTRESS, GPSTRAIN commands. The (PLOT) item suppresses output of these items to the f06 file. The OUTPUT(POST) section is used to define the surface for the grid point stress and strain requests.

The DTI,UNITS in the Bulk Data Section is **required** for the interface run and **must be placed in the main bulk section**. This defines the units to be used by ADAMS/Flex.

Standard SUBCASE is used to define two pressure load conditions. No constraints are applied to the component, hence the modal analysis will represent a free structure.

The PARAM,WTMASS is not recommended with interface runs and this is noted by explicitly setting it to its default value of 1.

The PARAM,GRDPNT will cause the MSC.Nastran grid point weight generator (GPWG) to run and display a summary of the mass properties of the model. Any physical grid location may be chosen for the display purpose, however with the ADAMS interface if PARAM,GRDPNT is present, GPWG will be run a second time **always** at the origin of the basic system and its results will be used for mass invariants 1I , 2I , and 7I computations for MNF export.

The SENQSET Bulk Data entry is used to define the component modes and **must be placed in the main bulk section**. This entry must specify enough degrees of freedom or modal amplitudes to capture the shape of the component and residual flexibility for any loading conditions. Instead of the SENQSET in the main bulk section, SPOINT and QSET1 Bulk Data entries could have been used in the BEGIN BULK SUPER = 200 section.

Note that the EIGR entry occurs in both the BEGIN BULK and BEGIN BULK SUPER = 200 sections.

```
SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$
$ Generate stress and strain grid shapes
```

```

$
STRESS(PLOT) = ALL
STRAIN(PLOT) = ALL
GPSTRESS(PLOT) = ALL
GPSTRAIN(PLOT) = ALL
$
SUBCASE 5
LABEL= one atu overpressure - uniform
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
load =50
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
$ Define surface for stress and strain grid shapes
$
OUTPUT(POST)
SET 9998 = ALL
SURFACE 9998 SET 9998 NORMAL X3
$
BEGIN BULK
$
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
GRID      1          0.0      0.0      0.0
GRID     11          0.0      1.0      0.0
GRID    111          1.0      0.0      0.0
GRID    121          1.0      1.0      0.0
$
$ SENQSET to define DOFs to use for component modes -
$ must be in main Bulk Data Section
$
SENGSET 200      18
$
$ ADAMS REQUIRES following DTI -
$ with PARTS this must be in main Bulk Data Section
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Select number of modes:
$
$ =====
EIGR      300      LAN                      10
$ =====
$
BEGIN BULK SUPER = 200
$
$ This PART defines interior grids as superelement 200
$ The corner grids 1, 11, 111, 121 are the exterior

```

```

$ or attachment point grids
$
$
$ Turn on gridpoint weight generator
$
PARAM,GRDPNT,0
$
$ Default value -
$ ADAMS must use the DTI,UNITS defined in main Bulk Data Section
$
PARAM,WTMASS,1.0
$
$ Select number of modes - for PART (Substructure) 200
$
$ =====
EIGR      300      LAN                      10
$ =====
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
$ Get model data for PART 200 and load data
$
include 'modell1.dat'
$
ENDDATA

```

Ex:4 Component Modeled as a Superelement with Preload

This example is the same as **Ex:1**, but since preload requires a static solution, the following additions are needed:

In each of the SUBCASEs that define the pressure loads, SPC=100 entries have been added to supply static supports. Additionally, a SUBCASE 200 has been added, with SPC=100 and LOAD=100 entries, to define the preload condition. This condition is defined by the FORCE 100 entries added to the Bulk Data Section. The resultant differential stiffness matrix will add stiffness to modes with rotation about the y-axis and z-axis and only four rigid body modes will be present. In this run, the FORCE entries should not be replaced with equivalent follower load FORCE 1 entries because the follower stiffness will move down stream with grids 1, 11, 111, 121 and not contribute to the component.

Also in the Bulk Data Section, SPC1 100 entries have been included to supply a static support system at the attachment points.

A SUBCASE 300 has been included that identifies the preload condition with the STATSUB(PRELOAD)=200 entry. Note that this subcase contains no SPC entry, hence the modal analysis will represent a free - free solution.

```

SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$
SUBCASE 5
LABEL= one atu overpressure - uniform
spc=100 $
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
spc=100 $
load =50
SUBCASE 200
LABEL= static stiffening load in plane of plate use a preload
spc=100 $
load=100
$
$ Select preload and generate free-free component shapes
$
SUBCASE 300
LABEL= Select SC200 as preload - modes as free-free structure
STATSUB((PRELOAD) = 200
$
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010

```

```

$
BEGIN BULK
$
$ ADAMS REQUIRES following DTI
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Turn on gridpoint weight generator
$
PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the above DTI,UNITS
$
PARAM,WTMASS,1.0
$
$ Add in plate stiffening load
$
$ For an ADAMS/Flex run do not use follower
$ FORCE1 etc. entries with this example as the
$ loads at grids 1,11,111,121 will move
$ down stream with these boundary grids
$ and their contribution to the follower
$ stiffness will be lost to the compont
$
FORCE 100 111 500. 6.
FORCE 100 112 1000. 6.
FORCE 100 113 1000. 6.
FORCE 100 114 1000. 6.
FORCE 100 115 1000. 6.
FORCE 100 116 1000. 6.
FORCE 100 117 1000. 6.
FORCE 100 118 1000. 6.
FORCE 100 119 1000. 6.
FORCE 100 120 1000. 6.
FORCE 100 121 500. 6.
$
FORCE 100 1 500. -6.
FORCE 100 2 1000. -6.
FORCE 100 3 1000. -6.
FORCE 100 4 1000. -6.
FORCE 100 5 1000. -6.
FORCE 100 6 1000. -6.
FORCE 100 7 1000. -6.
FORCE 100 8 1000. -6.
FORCE 100 9 1000. -6.
FORCE 100 10 1000. -6.
FORCE 100 11 500. -6.
$
$ static support set for preload -
$ note because of differential stiffness formulation,
$ only 4 zero modes now exist. The zero modes
$ for y-axis and z-axis rotation will be lost
$
SPC1 100 123 1

```

```

SPC1      100      13      11
SPC1      100      3       111
$
$
$ Select number of modes:
$
$ =====
FIGR      300      LAN                               10
$ =====
$
$ Define interior grids as superelement 200
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
SESET,200,2,THRU,10
SESET,200,12,THRU,110
SESET,200,112,THRU,120
$
$ SCALAR Point and SEQSET1 to define DOFs to use for component modes
$
SPOINT,80001,THRU,80019
SEQSET1,200,0,80001,THRU,80019
$
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
$ SEELT to put element 1003 and 1004 into SE 200
$
SEELT    200      10003  10004
$
$ Get model data and load data
$
include 'modell1.dat'
$
ENDDATA

```

Ex:5 Component Modeled as a Residual only Run with Preload

This example is the same as **Ex:2**, but since preload requires a static solution, the following additions are needed:

In each of the SUBCASEs that define the pressure loads, SUPORT1=100 entries have been added to supply static supports. Additionally, a SUBCASE 200 has been added, with SUPORT1=100 and LOAD=100 entries, to define the preload condition. This condition is defined by the follower FORCE1 100 entries added to the Bulk Data Section. Also in the Bulk Data Section, SUPORT1 100 entries have been included to supply a static support system at the attachment points. This run requires the use of SUPORT1 entries rather than SPC or SPC1 entries in order to avoid conflict between s-set points defined via SPC or SPC1 and a-set points defined via the ASET1 entry.

A SUBCASE 300 has been included that identifies the preload condition with the STATSUB(PRELOAD)=200 entry. Note that this subcase contains no SUPORT1 entry, hence the modal analysis will represent a free - free solution.

```
SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$
SUBCASE 5
LABEL= one atu overpressure - uniform
suport1=100 $
load=5
$
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
suport1=100 $
load =50
$
SUBCASE 200
LABEL= static stiffening load in plane of plate use a preload
suport1=100 $
load=100
$
$ Select preload and generate free-free component shapes
```

```

$
SUBCASE 300
LABEL= Select SC200 as preload - modes as free-free structure
STATSUB((PRELOAD) = 200
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
BEGIN BULK
$
$ ADAMS REQUIRES following DTI
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Turn on gridpoint weight generator
$
PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the above DTI,UNITS
$
PARAM,WTMASS,1.0
$
$ Add in plate tensioning follower load
$
FORCE1 100 111 3000. 1 111
FORCE1 100 112 6000. 2 112
FORCE1 100 113 6000. 3 113
FORCE1 100 114 6000. 4 114
FORCE1 100 115 6000. 5 115
FORCE1 100 116 6000. 6 116
FORCE1 100 117 6000. 7 117
FORCE1 100 118 6000. 8 118
FORCE1 100 119 6000. 9 119
FORCE1 100 120 6000. 10 120
FORCE1 100 121 3000. 11 121
$
FORCE1 100 1 3000. 111 1
FORCE1 100 2 6000. 112 2
FORCE1 100 3 6000. 113 3
FORCE1 100 4 6000. 114 4
FORCE1 100 5 6000. 115 5
FORCE1 100 6 6000. 116 6
FORCE1 100 7 6000. 117 7
FORCE1 100 8 6000. 118 8
FORCE1 100 9 6000. 119 9
FORCE1 100 10 6000. 120 10
FORCE1 100 11 3000. 121 11
$
$ Select number of modes:
$
$ =====
EIGR 300 LAN 10

```



```

$ =====
$
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
ASET1,123456,1,11,111,121
$
$ SCALAR Point and QSET1 to define DOFs to use for component modes
$
SPOINT,80001,THRU,80019
QSET1,0,80001,THRU,80019
$
$ Static support for preload
$
$ Note use of SUPORT1 entry and not SPC entry! This is to avoid
$ fatal conflict of grids 1, 11, and 111 of being in s-set and a-set
$ SUPORT1 places them in r-set which is a subset of a-set
$ SUPORT1 chosen over SUPORT as it is subcase selectable
$
SUPORT1 100      1      123
SUPORT1 100      11     13
SUPORT1 100     111     3
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
$ Get model data and load data
$
include 'modell1.dat'
$
ENDDATA

```

Ex:6 Component Modeled as a Part Superelement with Preload

This example is the same as **Ex:3**, but since preload requires a static solution, the following additions are needed:

In each of the SUBCASEs that define the pressure loads, SPC=100 entries have been added to supply static supports. Additionally, a SUBCASE 200 has been added, with SPC=100 and LOAD=100 entries, to define the preload condition. This condition is defined by the follower FORCE1 100 entries added to the BEGIN BULK SUPER = 200

section. Also in the main BEGIN BULK Section, SPC1 100 entries have been included to supply a static support system at the attachment points. For this run, the SPC1 100 entries could have been placed in the BEGIN BULK SUPER = 200 section.

A SUBCASE 300 has been included that identifies the preload condition with the STATSUB(PRELOAD)=200 entry. Note that this subcase contains no SPC entry, hence the modal analysis will represent a free - free solution.

```

SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$
SUBCASE 5
LABEL= one atu overpressure - uniform
spc=100 $
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
spc=100 $
load =50
SUBCASE 200
LABEL= static stiffening load in plane of plate use a preload
spc=100 $
load=100
$
$
$ Select preload and generate free-free component shapes
$
SUBCASE 300
LABEL= Select SC200 as preload - modes as free-free structure
STATSUB((PRELOAD) = 200
$
$ Set to define component shapes
$

```

```

OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
BEGIN BULK
$
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
GRID    1           0.0    0.0    0.0
GRID   11           0.0    1.0    0.0
GRID  111           1.0    0.0    0.0
GRID  121           1.0    1.0    0.0
$
$ static support set for preload
$
SPC1   100    123    1
SPC1   100    13     11
SPC1   100    3     111
$
$ SENQSET to define DOFs to use for component modes -
$ must be in main Bulk Data Section
$
SENGSET 200    19
$
$ ADAMS REQUIRES following DTI -
$ with PARTS this must be in main Bulk Data Section
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Select number of modes:
$
$ =====
EIGR    300    LAN                                10
$ =====
$
BEGIN BULK SUPER = 200
$
$ This PART defines interior grids as superelement 200
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids
$
$
$ Turn on gridpoint weight generator
$
PARAM,GRDPNT,0
$
$ Default value -
$ ADAMS must use the DTI,UNITS defined in main Bulk Data Section
$
PARAM,WTMASS,1.0
$
$ Add in plate tensioning follower load
$
FORCE1 100    111    3000.    1    111

```

```

FORCE1 100 112 6000. 2 112
FORCE1 100 113 6000. 3 113
FORCE1 100 114 6000. 4 114
FORCE1 100 115 6000. 5 115
FORCE1 100 116 6000. 6 116
FORCE1 100 117 6000. 7 117
FORCE1 100 118 6000. 8 118
FORCE1 100 119 6000. 9 119
FORCE1 100 120 6000. 10 120
FORCE1 100 121 3000. 11 121
$
FORCE1 100 1 3000. 111 1
FORCE1 100 2 6000. 112 2
FORCE1 100 3 6000. 113 3
FORCE1 100 4 6000. 114 4
FORCE1 100 5 6000. 115 5
FORCE1 100 6 6000. 116 6
FORCE1 100 7 6000. 117 7
FORCE1 100 8 6000. 118 8
FORCE1 100 9 6000. 119 9
FORCE1 100 10 6000. 120 10
FORCE1 100 11 3000. 121 11
$
$ Select number of modes - for PART (Substructure) 200
$
$ =====
EIGR 300 LAN 10
$ =====
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
$ Get model data for PART 200 and load data
$
include 'modell1.dat'
$
ENDDATA

```

Ex:7a Nonlinear Static Analysis of Component for Standard Restart

This example represents a MSC.Nastran nonlinear SOL 106 run with the component a standard residual structure.

SUBCASE 200 defines with SPC=100 and LOAD=100 entries, a preload condition or stiffening load condition. This condition is defined by the follower FORCE1 100 entries added to the Bulk Data Section. Also in the Bulk Data Section, SPC1 100 entries have been included to supply a static support system at the plate corner points; these will become the attachment points in the restart SOL 103 run.

NLPARM in Case Control command selects the Bulk Data entry NLPARM to define non linear parameters.

The PARAM,LGDISP,1 is required to turn on large displacement.

The SPCF(PLOT) = ALL generates forces of constraints. These will be used to form the total reactive preload in SOL 103.

SPOINT entries are **required** to be in the nonlinear run. They have no effect on the nonlinear solution but are place holders for the restart. The associated QSET1 entries **must not** appear in the nonlinear run as they will cause SOL 106 to fail because by current definition in MSC.Nastran, the q-set and a-set are synonyms and nonlinear requires a physical a-set.

When the job is submitted, SCR=NO must be invoked to insure that the data base is saved.

```
$ The data base must be saved for this run therefore SCR=NO required
SOL 106
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Get nonlinear stress output
$
NLSTRESS = ALL
$
SUBCASE 200
LABEL= static stiffining load in plane of plate for preload
SPCF(PLOT) = ALL $ Generate forces of constraint
SPC = 100 $
LOAD=100
$
$ Select nonlinear parameters
$
NLPARM = 1
$
BEGIN BULK
$
$ Turn on large displacements
$
PARAM,LGDISP,1
$
$ Nonlinear parameters
```

```

$
NLPARM,1,4,,,,,UPW,YES
$
$ ADAMS REQUIRES following DTI
$ either in nonlinear run or Sol 103 run but not both
$
$DTI,UNITS,1,KG,N,M,SEC
$
$ If wanted, turn on gridpoint weight generator
$ either in nonlinear run or Sol 103 run but not both
$
$PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the DTI,UNITS
$
PARAM,WTMASS,1.0
$
$ Add in plate tensioning follower load
$
FORCE1 100 111 3000. 1 111
FORCE1 100 112 6000. 2 112
FORCE1 100 113 6000. 3 113
FORCE1 100 114 6000. 4 114
FORCE1 100 115 6000. 5 115
FORCE1 100 116 6000. 6 116
FORCE1 100 117 6000. 7 117
FORCE1 100 118 6000. 8 118
FORCE1 100 119 6000. 9 119
FORCE1 100 120 6000. 10 120
FORCE1 100 121 3000. 11 121
$
FORCE1 100 1 3000. 111 1
FORCE1 100 2 6000. 112 2
FORCE1 100 3 6000. 113 3
FORCE1 100 4 6000. 114 4
FORCE1 100 5 6000. 115 5
FORCE1 100 6 6000. 116 6
FORCE1 100 7 6000. 117 7
FORCE1 100 8 6000. 118 8
FORCE1 100 9 6000. 119 9
FORCE1 100 10 6000. 120 10
FORCE1 100 11 3000. 121 11
$
$
$ static support set for preload
$
SPC1 100 123 1
SPC1 100 13 11
SPC1 100 3 111
$
$
$ =====
$
$ scalar points must be added here for modal sol103 nmloop restart

```

```

$ because to add them in the modal 103 run would cause restart
$ to generate whole model over since changing grids and elements
$ flags model as new
$ The Qsets and Asets must be added to the sol103 run
$ They can not exist in the sol106 run
$
SPOINT,80001,THRU,80019
$
$ Get model data and load data
$
include 'modell.dat'
$
ENDDATA

```

Ex:7b Nonlinear Static Analysis Restart into SOL103

This example represents a restart for SOL 106 into SOL 103 to perform a MSC.Nastran/ADAMS interface run with the component modeled as residual only structure. Grid point 1, grid point 11, grid point 111, and grid point 121 are to be used as the attachment points for ADAMS. This is accomplished by use of the ASET1 Bulk Data entry which is inserted in this file. The **nonlinear run cannot contain this ASET entry**.

The restart is indicated by the presence of the ASSIGN MASTER and RESTART file management commands in the file management section of the MSC.Nastran input stream.

The PARAM,NMLOOP in Case Control selects the converged nonlinear solution to use for the restart.

The ADAMSMNF Case Control command initiates the interface run with the **required** FLEXBODY=YES entry. Also include on this entry is the optional PSETID=7772 which provides a set of PLOTTEL elements whose grids are retained in the MNF and whose connectivity defines face geometry for ADAMS display. The OUTPUT(PLOT) section is used to define the set of elements for the PSETID. For this case, PLOTTELS were used and these are defined at the end of the Bulk Data Section.

The DTI,UNITS in the Bulk Data Section is **required** for the interface run. This defines the units to be used by ADAMS/Flex. This entry could have been placed in the SOL 106 run but it cannot be placed in both because of standard MSC.Nastran restart requirements.

Standard SUBCASE is used to define two pressure load conditions. No constraints are applied to the component, hence the modal analysis will represent a free structure.

The PARAM,WTMASS is not recommended with interface runs and this is noted by explicitly setting it to its default value of 1 in the SOL 106 run. Hence, it is not reinserted in the SOL 103 restart run.

The PARAM,GRDPNT will cause the MSC.Nastran grid point weight generator (GPWG) to run and display a summary of the mass properties of the model. Any physical grid location may be chosen for the display purpose, however with the ADAMS interface if PARAM,GRDPNT is present, GPWG will be run a second time **always** at the origin of the basic system and its results will be used for mass invariants 1I , 2I , and 7I computations for MNF export.

The SPOINT and QSET1 Bulk Data entries are used to define the component modes. The SPOINT, **as required**, were placed in the SOL 106 run. The QSET1 is **required** to be placed in the restart SOL 103 run. These entries must specify enough degrees of freedom or modal amplitudes to capture the shape of the component and residual flexibility for any loading conditions.

```

$
$ Get the data base for the nonlinear run
$
ASSIGN MASTER='e7a_residual_106.MASTER'
$
$ Indicate a restart run
$
RESTART VERSION=1 KEEP
$
SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$
$ Select the desired converged nonlinear solution
$

```



```

PARAM,NMLOOP,4
$
$ include preload condition from sol 106 for labeling
$ must be first subcase
$
SUBCASE 2
LABEL=Sol 106 preload
load=100
$
SUBCASE 5
LABEL= one atu overpressure - uniform
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
load =50
$
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
BEGIN BULK
$
$ ADAMS REQUIRES following DTI
$ either in nonlinear run or Sol 103 run but not both
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Turn on gridpoint weight generator in this run
$
PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the above DTI,UNITS
$ Already done is Sol 106 run
$
$PARAM,WTMASS,1.0
$
$ Select number of modes:
$
$ =====
EIGR      300      LAN                      10
$ =====
$
$ define attachment points
$
ASET1,123456,1,11,111,121
$
$ SCALAR Points supplied in sol106 run should match following QSET
$ QSET to define DOFs to use for normal modes
$
QSET1,0,80001,THRU,80019
$
$ Define plotel's for set 7772 in output(plot) section

```

```

$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
ENDDATA

```

Ex:8a Nonlinear Static Analysis of Component for dblocate Restart

This example represents a MSC.Nastran nonlinear SOL 106 run with the component a nonlinear residual structure.

SUBCASE 200 defines with SPC=100 and LOAD=100 entries, a preload condition or stiffening load condition. This condition is defined by the follower FORCE1 100 entries added to the Bulk Data Section. Also in the Bulk Data Section, SPC1 100 entries have been included to supply a static support system at the plate corner points; these will become the attachment points in the restart SOL 103 run.

NLPARM in Case Control selects the Bulk Data entry, NLPARM, to define nonlinear parameters.

The PARAM,LGDISP,1 is required to turn on large displacement.

The SPCF(PLOT) = ALL generates forces of constraints. These will be used to form the total reactive preload in SOL 103.

No SPOINT entries (**required** for a standard restart from SOL 106 to SOL 103) are given. Thus a nonstandard restart into SOL 103, using dblocate in SOL 103 will be required. QSET1 entries **must not** appear in the nonlinear run as they will cause SOL 106 to fail because by current definition in MSC.Nastran, the q-set and a-set are synonyms and nonlinear requires a physical a-set.

```

$ The data base must be saved for this run therefore SCR=NO required
SOL 106
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Get nonlinear stress output
$
NLSTRESS = ALL
$
SUBCASE 200
LABEL= static stiffening load in plane of plate use a preload
SPCF(PLOT) = ALL $ Generate forces of constraint

```

```

SPC = 100 $
LOAD=100
$
$ Select nonlinear parameters
$
NLPARM = 1
$
BEGIN BULK
$
$ Turn on large displacements
$
PARAM,LGDISP,1
$
$ Nonlinear parameters
$
NLPARM,1,4,,,,,UPW,YES
$
$ ADAMS REQUIRES following DTI
$either in nonlinear run or Sol 103 run but not both
$
$DTI,UNITS,1,KG,N,M,SEC
$
$ If wanted, turn on gridpoint weight generator
$ either in nonlinear run or Sol 103 run but not both
$
$PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the DTI,UNITS
$
PARAM,WTMASS,1.0
$
$ Add in plate tensioning follower load
$
FORCE1 100      111      3000.   1        111
FORCE1 100      112      6000.   2        112
FORCE1 100      113      6000.   3        113
FORCE1 100      114      6000.   4        114
FORCE1 100      115      6000.   5        115
FORCE1 100      116      6000.   6        116
FORCE1 100      117      6000.   7        117
FORCE1 100      118      6000.   8        118
FORCE1 100      119      6000.   9        119
FORCE1 100      120      6000.  10       120
FORCE1 100      121      3000.  11       121
$
FORCE1 100      1        3000.  111      1
FORCE1 100      2        6000.  112      2
FORCE1 100      3        6000.  113      3
FORCE1 100      4        6000.  114      4
FORCE1 100      5        6000.  115      5
FORCE1 100      6        6000.  116      6
FORCE1 100      7        6000.  117      7
FORCE1 100      8        6000.  118      8
FORCE1 100      9        6000.  119      9

```

```

FORCE1  100      10      6000.  120      10
FORCE1  100      11      3000.  121      11
$
$ static support set for preload
$
SPC1     100      123      1
SPC1     100      13       11
SPC1     100      3       111
$
$ =====
$
include 'modell1.dat'
$
ENDDATA

```

Ex:8b Nonlinear Static Analysis dblocate Restart into SOL103

This example represents a restart for SOL 106 into SOL 103 to perform a MSC.Nastran/ADAMS interface run with the component modeled as residual only structure. Grid point 1, grid point 11, grid point 111, and grid point 121 are to be used as the attachment points for ADAMS. This is accomplished by use of the ASET1 Bulk Data entry which is inserted in this file. The **nonlinear run cannot contain this ASET** entry.

Since the SOL 106 run did not contain SPOINT Bulk Data entries, the nonstandard restart is indicated by the presence of the ASSIGN MASTER and DBLOCATE file management commands in the file management section of the MSC.Nastran input stream.

The PARAM,NMLOOP in case control selects the converged nonlinear solution to use for the restart.

The ADAMSMNF Case Control command initiates the interface run with the **required** FLEXBODY=YES entry. Also include on this entry is the optional PSETID=7772 which provides a set of PLOTTEL elements whose grids are retained in the MNF and whose connectivity defines face geometry for ADAMS display. The OUTPUT(PLOT) section is used to define the set of elements for the PSETID. For this case, PLOTTELS were used and these are defined at the end of the Bulk Data Section.

The DTI,UNITS in the Bulk Data Section is **required** for the interface run. This defines the units to be used by ADAMS/Flex. This entry could have been placed in the SOL 106 run but it cannot be placed in both because of standard MSC.Nastran restart requirements.

Standard SUBCASE is used to define two pressure load conditions. No constraints are applied to the component, hence the modal analysis will represent a free structure.

The PARAM,WTMASS is not recommended with interface runs and this is noted by explicitly setting it to its default value of 1 in the SOL 106 run. Hence, it is not reinserted in the SOL 103 restart run.

The PARAM,GRDPNT will cause the MSC.Nastran grid point weight generator (GPWG) to run and display a summary of the mass properties of the model. Any physical grid location may be chosen for the display purpose, however with the ADAMS interface if PARAM,GRDPNT is present, GPWG will be run a second time **always** at the origin of the basic system and its results will be used for mass invariants 1I , 2I , and 7I computations for MNF export.

The SPOINT and QSET1 Bulk Data entries are used to define the component modes. The SPOINT entry are placed in this restart run and is what causes the use of a DBLOCATE nonstandard restart run. The QSET1 is **required** to be placed in the restart SOL 103 run. These entries must specify enough degrees of freedom or modal amplitudes to capture the shape of the component and residual flexibility for any loading conditions.

```

$
$ Get the data base for the nonlinear run
$
ASSIGN MASTER='e8a_residual_106_db.MASTER'
$
$ A dblocate run
$
dblocate datablk=(bulk,estnl,ugni,
                  qgni,qmgni,pgni) where(version=1) logi=master
$
SOL 103
CEND
$
TITLE= SIMPLE PLATE MODEL 10 X 10 ELEMENTS
$
$ Initiate an MSC.Nastran/ADAMS interface run
$ flexbody=yes is REQUIRED
$ psetid will define face geometry for adams component display
$
ADAMSMNF flexbody=yes, psetid=7772
$
$ Select real eigen value parameters - must be above or in first subcase
$
METHOD=300
$
$ Starting with v2004, residual vectors are by default always computed.
$ They should be controlled with the following case control command:
$ It is recommended that the MSC.Nastran defaults be taken.
$ The defaults result in numerically improved shapes.
$
RESVEC = COMPONENT
$

```

```

$ Select the desired converged nonlinear solution
$
PARAM,NMLOOP,4
$
$ include preload condition from sol 106 for labeling
$ must be first subcase
$
SUBCASE 2
LABEL=Sol 106 preload
load=100
$
$ Generate two modal loads
$
SUBCASE 5
LABEL= one atu overpressure - uniform
load=5
SUBCASE 50
LABEL= spin about edge (Y-Axis) at 1 revs/sec (60 RPM)
load =50
$
$
$ Set to define component shapes
$
OUTPUT(PLOT)
SET 7772 = 10001 THRU 10010
$
$ Define surface for stress and strain grid shapes
$
BEGIN BULK
$
param,respath,old
$
$ ADAMS REQUIRES following DTI
$ either in nonlinear run or Sol 103 run but not both
$
DTI,UNITS,1,KG,N,M,SEC
$
$ Turn on gridpoint weight generator in this run
$
PARAM,GRDPNT,0
$
$ Default value - ADAMS must use the above DTI,UNITS
$ Already done is Sol 106 run
$
$PARAM,WTMASS,1.0
$
$ Select number of modes:
$
$ =====
EIGR      300      LAN                               10
$ =====
$
$ The corner grids 1, 11, 111, 121 are the exterior
$ or attachment point grids

```

```

$
ASET1,123456,1,11,111,121
$
$ SCALAR Point and QSET1 to define DOFs to use for component modes
$
SPOINT,80001,THRU,80019
QSET1,0,80001,THRU,80019
$
$ Define plotel's for set 7772 in output(plot) section
$
PLOTTEL,10001,1,12
PLOTTEL,10002,12,121
PLOTTEL,10003,121,111
PLOTTEL,10004,111,1
PLOTTEL,10006,2,10
PLOTTEL,10007,22,110
PLOTTEL,10008,120,112
PLOTTEL,10009,12,100
$
ENDDATA

```

Model Data for the Examples

The following data defines the plate model data.

```

$
$ START GRID SPECS
$
GRID, 1,, 0.0, 0.0, 0.0
=,*11,,*.10,==
=9
GRID, 2,, 0.0, 0.1, 0.0
=,*11,,*.10,==
=9
GRID, 3,, 0.0, 0.2, 0.0
=,*11,,*.10,==
=9
GRID, 4,, 0.0, 0.3, 0.0
=,*11,,*.10,==
=9
GRID, 5,, 0.0, 0.4, 0.0
=,*11,,*.10,==
=9
GRID, 6,, 0.0, 0.5, 0.0
=,*11,,*.10,==
=9
GRID, 7,, 0.0, 0.6, 0.0
=,*11,,*.10,==
=9
GRID, 8,, 0.0, 0.7, 0.0
=,*11,,*.10,==
=9
GRID, 9,, 0.0, 0.8, 0.0
=,*11,,*.10,==

```

```

=9
GRID, 10,, 0.0, 0.9, 0.0
=,*11,,*.10,==
=9
GRID, 11,, 0.0, 1.0, 0.0
=,*11,,*.10,==
=9
$
$ START ELEMENT SPECS
$
CQUAD4, 1,1,1,2,13,12
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 11,1,2,3,14,13
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 21,1,3,4,15,14
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 31,1,4,5,16,15
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 41,1,5,6,17,16
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 51,1,6,7,18,17
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 61,1,7,8,19,18
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 71,1,8,9,20,19
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 81,1,9,10,21,20
=,*1,=,*11,*11,*11,*11
=8
CQUAD4, 91,1,10,11,22,21
=,*1,=,*11,*11,*11,*11
=8
$
$ ELEMENT PROPERTIES INPUT
$
PSHELL,1,1,0.010,1
$
$ MATERIAL PROPERTIES INPUT
$
MAT1,1,.2223+12,.0855+12,0.3,7803.0
$
$ Modal load set 5: one atu overpressure - uniform
$
PLOAD45      1      101344.          THRU 100
$
$

```



```
$ Modal load set 50: spin about edge (Y-Axis) at 1 revs/sec (60 RPM)  
$  
RFORCE,50,1,,1.0,0.0,1.0,0.0  
$  
$
```

13.13 Rotordynamics

13.13.1 Introduction

The new rotordynamics capability provides users with a relatively simple method of performing the design and analysis of structures with rotating components. The procedures were developed with input from aerojet engine manufacturers to ensure that their basic requirements were met. The new rotordynamic capability can be used in frequency response (direct and modal), complex modes (direct and modal), static, nonlinear transient, and linear transient (direct only) analyses.

Frequency response analyses determine the rotor/support response to arbitrary excitation with the rotors spinning at user specified rates (ASYNC) or excitation that is synchronous with the reference rotor (SYNC). The ASYNC calculation determines the response of the system to an external input that is independent of the rotor speed. The SYNC calculation determines the system response to a rotor imbalance or other excitation that is dependent on the rotor spin rate.

Complex modes analyses determine the whirl frequencies and critical speeds. Whirl modes are modes of a rotor/support system with the rotors spinning at specified rates. Critical speeds are whirl frequencies that coincide with a rotor spin rate. Calculation of whirl frequencies is performed by the asynchronous option (ASYNC). Calculation of critical speeds is performed by the synchronous option (SYNC).

Static analysis determines the loads on the rotor due to pitch or yaw of an aircraft. Severe transverse rotations may be part of the design environment and limiting rotor motion to prevent rotor blades from rubbing against the interior of the casing may be design criteria.

Direct linear and nonlinear transient analyses allow simulation of engine blade-out and subsequent windmilling. The analysis is used to ensure structural integrity during flight and guarantee that excessive vibration levels are not reached.

All the above analyses are performed in the stationary (inertial) coordinate system with the assumption that the rotors have rotational symmetry of three or greater and are reduced (by static condensation) to a line model. The rotor is assumed to be flexible along its axis and planes perpendicular to the axis remain perpendicular to the axis as the axis flexes. Motion off-axis can be recovered from the motion of the point on the line model that is in the same perpendicular plane.

Benefits

Traditionally, companies have used proprietary in-house codes and customized DMAP solutions to perform analyses of structures with rotating components. The rotordynamics capability in MSC.Nastran provides an easy-to-use standardized procedure for analyzing structures. The rotordynamics capability is included in the MSC.Nastran dynamic solution sequences, and invoked using case control commands and bulk data entries. This standardization allows easier exchange of model data between in-house groups or external vendors.

The design of the rotordynamics capability allows multiple rotors to be included in the analyses. The rotors may spin at different rates and be aligned at arbitrary angles.

Basic Theory and Methods

All analyses are performed in the stationary coordinate system. This requires that the rotors are symmetric about the rotation axis. Analysis in the stationary system also requires that the rotors be modeled as line models or 3-D models statically reduced to a line model (component-mode models are not allowed). Data recovery is correct for the engine casing and support structure. Data recovery for the rotors is limited to the overall response along the axis of rotation, detailed displacements and stress recovery off the axis of rotation is not correct.

Basic Equations

The MSC.Nastran time-domain equation solved by the new dynamic solution sequences is:

$$M\ddot{u}(t) + \left(B_s + \left(\frac{g}{W3} \right) K_s + \left(\frac{1}{W4} \right) K4_s + B_r + \left(\frac{g_r}{WR3} \right) K_r + \left(\frac{1}{WR4} \right) K4_r + \Omega B^G \right) \dot{u}(t) + \left(K_s + K_r + \Omega \left(K^{Cv} + \left(\frac{g_r}{WR3} \right) K^{Cgr} + \left(\frac{1}{WR4} \right) K^{Cge} \right) \right) u(t) = F(t) \quad \text{Eq. 13-131}$$

M = total mass matrix (CONMi, CMASSi, MATi)

B_s = support viscous damping matrix (CVISC, DCAMPi)

$\left(\frac{g}{W3} \right) K_s$ = support viscous damping equivalent to structural damping (PARAM,G)

$\left(\frac{1}{W4} \right) K4_s$ = support viscous damping equivalent to material structural damping (GE on MATi)

B_r = rotor viscous damping matrix (CVISC, CDAMPi)

- $\left(\frac{g_r}{WR3}\right)K_r$ = rotor viscous damping equivalent to structural damping (GR on RSPINT)
 $\left(\frac{1}{WR4}\right)K4_r$ = rotor viscous damping equivalent to material structural damping (GE on MATi)
 B^G = gyroscopic force matrix (dependent on polar moment of inertia)
 K_s = support stiffness matrix
 K_r = rotor stiffness matrix
 $K4_s$ = support material damping matrix (GE on MATi)
 $K4_r$ = rotor material damping matrix (GE on MATi)
 Ω = rotor rotation rate
 K^{Cv} = “circulation” matrix due to B_r
 $g_r K_r^{Cgr}$ = “circulation” matrix due to $g_r K_r$
 K_r^{Cge} = “circulation” matrix due to $K4_r$

G , $WR3$, and $WR4$ are user parameters

The MSC.Nastran frequency-domain equation is:

$$\begin{aligned}
 &(-\omega^2 M + i\omega(B_s + B_r + \Omega B^G)) \\
 &+ \left((1 + ig)K_s + iK4_s + (1 + ig_r)K_r + iK4_r + \Omega \left(K^{Cv} + \left(\frac{g_r}{\omega}\right)K^{Cgr} + \left(\frac{1}{\omega}\right)K^{Cge} \right) \right) u(\omega) = F(\omega)
 \end{aligned}
 \tag{Eq. 13-132}$$

The rotor spin rate, Ω , is specified by the user. In the synchronous option where $\omega = \Omega$, the above equation becomes

$$\begin{aligned}
 &(-\Omega^2(M - iB^G) + i\Omega(B_s + B_r - iK^{Cv})) \\
 &+ ((1 + ig)K_s + iK4_s + (1 + ig_r)K_r + iK4_r + g_r K_r^{Cgr} + K^{Cge}) u(\Omega) = F(\Omega)
 \end{aligned}
 \tag{Eq. 13-133}$$

Eq. 13-131 is used in transient analysis; **Eq. 13-132**, and **Eq. 13-133** are used in frequency response and complex modes analyses. See “**Equations Used in Analyses**” on page 970 for more details.

Whirl Modes and Critical Speeds

A characteristic of the motion associated with whirl frequencies and critical speeds is that the rotor appears to “whirl” in a circular or elliptical motion relative to the support structure. The whirl motion may be either in the direction of rotor rotation (forward whirl) or against it (backward whirl). For whirl frequency motion, the rotor “whirl” is at a frequency that can be different than the rotor spin rate. Critical speeds can be viewed as a special case of whirl frequencies; the rotor “whirl” is at the same frequency as the rotor spin rate.

Complex eigenvalue analysis can be used to determine whirl frequencies and critical speeds, and their associated motion. For whirl frequencies, [Eq. 13-132](#) is solved for ω (whirl frequency) and u (mode shape) with $F = 0$ and the rotor spin rate Ω specified by the user. For critical speeds, [Eq. 13-133](#) is solved for Ω (critical speed) and u (mode shape) with $F = 0$.

Critical speeds can also be determined using whirl frequency calculations. This is done using Campbell diagrams. Campbell diagrams plot eigenvalues versus rotor speed or other parameters that affect the natural frequencies of the system. [Eq. 13-132](#) is solved for a series of rotor speeds and the results plotted on the Campbell diagram. Eigenvalues with similar modes shapes are connected to form a series of lines that represent the changing natural frequencies. Critical speeds are determined by assessing which eigenvalues are identical to the rotor speed.

An additional line for $\omega = \Omega$ (natural frequency = rotor spin rate) is drawn. The intersection of the $\omega = \Omega$ line with the eigenvalue lines are the critical speeds.

Whirl frequencies are important in determining rotor stability. Although the whirling motion may not be at the rotational speed, whirl modes may be unstable due to internal rotor damping or external forces due to bearing and seals. Estimating the stability of the whirl frequencies is important for ensuring a system that operates smoothly and will not become unstable.

Critical speeds are also important for ensuring a system that operates smoothly and will not become unstable. Although rotors are usually finely balanced, they are never perfectly balanced. Any imbalance in the rotor will produce oscillating forces in the rotor/support structure. If the oscillation is at or near natural frequency of the system (critical speed), the system will experience high vibratory motion. If the imbalance is large enough, the structure may fail or cause significant damage. Determining the critical speeds of the system allows the user to specify operating speeds which avoid the critical speeds or the structure can be redesign to move critical speeds away from the required operating speeds.

Multiple Rotors and Reference Rotor

For the situation of multiple rotors, [Eq. 13-131](#) through [Eq. 13-133](#), are modified to include the gyroscopic terms and spin rates of the individual rotors (see Section [13.13.4](#).) For time-domain analyses, the user specifies the rotor spin rates directly. For frequency domain and static analyses, a reference rotor is specified. The analyses are performed with the reference rotor spinning at the specified speed. The spin rates of the other rotors are determined by means of user specified relationships between the rotor spin rates.

For frequency-domain analyses (complex modes and frequency response) with synchronous gyroscopic forces, it is important to remember that the analyses are performed with respect to the reference rotor. The reference rotor is spinning at the analysis frequency, or for complex modes, at the eigenfrequency. The results should be interpreted in terms of the reference rotor frequency.

For example, using complex eigenvalue analysis, it is possible to determine the natural frequencies of a multi-rotor system consisting of rotors spinning at different rates. Consider a structure with two rotors. The second rotor spins at twice the speed of the first rotor. The first rotor is defined as the reference rotor and complex eigenvalue analysis is performed using the synchronous option $\omega = \Omega$. Upon inspection of the results, a mode appears to be a whirl mode of the second rotor. One would expect that the second rotor has a critical speed at twice the reported eigenfrequency (since with the synchronous option, the eigenfrequency corresponds to the rotation rate of the reference or first rotor, and the second rotor is spinning at twice the rate of the first). This would be an incorrect assumption. What it means is that when second rotor is spinning at twice the eigenfrequency (and the first rotor is spinning at the eigenfrequency), there is a whirl mode of the second rotor at that eigenfrequency. When inspecting the mode shapes, whirl modes of the reference rotor should be noted, their eigenfrequencies are critical speed of the reference rotor. Other modes should be ignored.

Input

Case Control

RGYRO Case Control Command invokes all rotordynamics capabilities.

- RGYRO = YES/NO, n (Default = No)
 - For complex modes, frequency response, and static analysis, the RGYRO Case Control command selects the RGYRO Bulk Data entry for use in the analysis.

- For direct linear and nonlinear transient response, the RGYRO Case Control command selects the UNBALNC Bulk Data entry that specifies the unbalance loading. If no unbalance loading is present, but gyroscopic terms are desired in the analysis, the RGYRO Case Control command should be set to YES.

Bulk Data Additions for All Rotor Dynamic Analyses

ROTORG -- Specifies grids that compose the rotor line model

Format:

1	2	3	4	5	6	7	8	9	10
ROTORG	ROTORID	GRID1	GRID2	...	GRIDn				

or

ROTORG	ROTORID	GRID1	THRU	GRID2	BY	Inc			
--------	---------	-------	------	-------	----	-----	--	--	--

Field	Contents
-------	----------

ROTORID	Identification number of rotor group. (Integer > 0; Required)
GRIDi	Grids comprising the rotor. (Integer > 0; Required, no Default)
THRU	Specifies a range of identification numbers. (Optional)
BY	Specifies an increment for a THRU specification (Optional)
Inc	Increment for THRU range. (Integer > 0; Default = 1)

Remarks:

1. Grid entries must be unique, duplicate entries will produce a fatal error.
2. Multiple ROTORG entries with the same ROTORID are supported.
3. All grids specified on ROTORG entries for a specific ROTORID must be collinear. Collinearity will be checked.
4. Grids specified but not found in the model will not cause an error. Grids which are not found will be listed in a summary.

Bulk Data Additions for Frequency Response, Complex Modes, and Static Analysis

RGYRO -- Specifies synchronous or asynchronous motion of the reference rotor, rotation speed of the reference rotor, and reference rotor number.

Format:

1	2	3	4	5	6	7	8	9	10
RGYRO	RID	SYNCLFG	REFROTR	SPDUNIT	SPDLOW	SPDHIGH	SPEED		

Field	Contents
RID	Identification number of RGYRO entry. Selected by Case Control command, RGYRO. (Required, no Default)
SYNCLFG	Specifies whether the analysis is synchronous or asynchronous analysis. Required input for frequency response and complex modes analyses. Not required for static analyses. (Character: 'SYNC', 'ASYNC', or blank)
REFROTR	Specifies the reference rotor ID for the analysis. (Integer > 0; Required, no Default)
SPDUNIT	Specified whether the entries SPDLOW, SPDHIGH, and SPEED given in terms of RPM (revolutions/minute) or frequency (revolutions (cycles)/unit time). (Character: 'RPM' or 'FREQ'; no Default)
SPDLOW	Specifies the low speed for complex modes synchronous analysis, see Remark 3. (Real, Default = 0.0)
SPDHIGH	Specifies the high speed for complex modes synchronous analysis, see Remark 3. (Real, Default = 99999.0)
SPEED	Specifies reference rotor rotation speed for asynchronous analysis in complex modes and frequency response analysis. Also required for static analyses. (Real, required if SYNCLFG = 'ASYNC' for complex modes and frequency response or analysis type is static analysis; no Default)

Remarks:

1. Multiple RGYRO entries with the same RID value are not allowed.
2. SPDLOW and SPDHIGH are used in complex modes analysis. SPEED is used in complex modes and frequency response analysis. For transient response analysis, the rotor spin rates are given using RSPINT entries.

3. For synchronous analysis in complex modes, the relative rotor speeds will be treated as being linearly dependent on the reference rotor speed ($\Omega = A0 + A1 \cdot \Omega_{\text{reference}}$). The scale factors A0 and A1 will be determined by a least-mean-square fit of the relative rotor speeds input on the RSPINR entry between SPDLOW and SPDHIGH of the reference rotor. If SPDLOW or SPDHIGH are outside the range specified on the RSPINR entry, the values will be extrapolated from the RSPINR entry values.
4. For synchronous analysis in frequency response, the relative rotor speeds will be determined directly from the RSPINR entries.

RSPINR - Specifies the relative spin rates between rotors for complex eigenvalue, frequency response, and static analysis.

Format:

1	2	3	4	5	6	7	8	9	10
RSPINR	ROTORID	GRIDA	GRIDB	GR	SPDUNIT	SPEED1	...	SPEEDn	

Field	Contents
ROTORID	Identification number of rotor. (Integer > 0; Required)
GRIDA/GRIDB	Positive rotor spin direction is defined from GRIDA to GRIDB (Integer > 0; Required, See Remark 2)
GR	Rotor structural damping factor, see Remark 3. (Real, Default = 0.0)
SPDUNIT	Specifies whether the listing of relative spin rates is given in terms of RPM (revolutions/minute) or frequency (revolutions/unit time). (Character; 'RPM' or 'FREQ'; required)
SPEEDi	List of relative spin rates, entries for reference rotor must be in ascending or descending order. (Real; at least one entry required, no Default)

Remarks:

1. A RSPINR entry must be present for each rotor defined by a ROTORG entry.
2. The rotor spin axis is determined from the ROTORG entries. The positive rotation vector is from GRIDA to GRIDB. GRIDA and GRIDB must be specified on the ROTORG entry.
3. Rotor structural damping specified by the GR entry will be added as equivalent viscous damping. The equivalent damping will be calculated using:

$$[B_{rotor}]_{structural} = \left(\frac{GR}{WR3} \right) [K_{rotor}]$$

where $WR3$ is specified using PARAM, WR3.

- Number of entries for each rotor must be the same. Relative spin rates are determined by correlation of table entries. The i th entry for each rotor specifies the relative spin rates between rotors at $RPM_i/FREQ_i$.

Bulk Data Additions for Transient Response Analysis

RSPINT - Specifies rotor spin rates for direct transient analysis.

Format:

1	2	3	4	5	6	7	8	9	10
RSPINT	ROTORID	GRIDA	GRIDB	GR	SPDUNIT	TID			

Field	Contents
ROTORID	Identification number of rotor. (Integer > 0; Required, no Default)
GRIDA/GRIDB	Positive rotor spin direction is defined from GRIDA to GRIDB (Integer > 0; Required, see Remark 1)
GR	Rotor structural damping factor, see Remark 2. (Real, Default = 0.0)
SPDUNIT	Specifies whether the spin rates is given in terms of RPM (revolutions/minute) or frequency (revolutions(cycles)/unit time). (Character; 'RPM' or 'FREQ'; Required, no Default)
TID	Identification number of TABLEDi entry. (Integer > 0; Required, no Default)

Remarks:

- The rotor spin axis is determined from the ROTORG entries. The positive rotation vector is from GRIDA to GRIDB. GRIDA and GRIDB must be specified on the ROTORG entry.
- Rotor structural damping specified by the GR entry will be added as equivalent viscous damping. The equivalent damping will be calculated using:

$$[B_{rotor}] = \left(\frac{GR}{WR3} \right) [K_{rotor}]$$

where $WR3$ is specified using PARAM, WR3.

UNBALNC -- Specifies an unbalance load for transient analysis in terms of a cylindrical system with the rotor rotation axis as the z-axis.

Format:

1	2	3	4	5	6	7	8	9	10
UNBALNC	RID	MASS	GRID	X1	X2	X3			
	ROFFSET	THETA	ZOFFSET	T _{on}	T _{OFF}	CFLAG			

Field	Contents
RID	Identification number of RGYRO entry. Selected by Case Control command, RGYRO. (Integer; Required; no Default)
MASS	Mass of imbalance (real or integer; if integer must be > 0; see Remark 3; Required, no Default)
GRID	Grid identification number of applying imbalance. The grid must appear on a ROTORG entry. (Integer .0; Required, no Default)
X1, X2, X3	Components of the vector, from GRID, in the displacement coordinate system of GRID, which is used to define a cylindrical coordinate system centered at GRID, see Remark 4. (Real, Required, no Default)
ROFFSET	Offset of mass in the radial direction of the unbalance coordinate system (real or integer; if integer, must be > 0; see Remark 3; Required, Default = 1.0)
THETA	Angular position of the mass in the unbalance coordinate system. (Real; Required, Default = 0.0)
ZOFFSETX	Offset of mass in the z-direction of the unbalance coordinate system. (Real or Integer; if integer, must be > 0; see Remark 3; Required, Default = 0.0)
T _{ON}	Start time for applying imbalance load. (Real \geq 0.0; Default = 0.0)
T _{OFF}	Time for terminating imbalance load. (Real > 0.0; Default = 999999.0)
CFLAG	Correct flag to specify whether 1) the mass will be used to modify the total mass in the transient response calculations, 2) the effect of the rotor spin rate change will be included in the transient response calculation, or 3) both, see Remark 4. (Character: NONE, MASS, SPEED, or BOTH, Default = NONE)

Remarks:

1. Multiple UNBALNC entries with the same RID value may be used.
2. The imbalance load will be generated based on the mass value, offset values, and the rotor spin speed.
3. If the entry is a real number, the value is considered constant. If the entry is an integer number, the value references a table entry that specifies the value as a function of time.
4. The cylindrical coordinate system used for the initial position of the mass unbalance has the positive z-axis direction from GRIDA to GRIDB specified on the RSPINT entry. Theta is measured from the plane defined by the z-axis and the user specified vector (X1, X2, X3). Theta = 0.0 is in the direction of the user-specified vector.
5. If CFLAG is not “NONE”, the GRID must be in the analysis set (A-set) of the model. If it does not belong to the analysis set, a FATAL error will be issued.

Parameters

The rotor dynamic parameters can be added in the Case Control or Bulk Data Sections.

GYROAVG - Used to specify one of two formulations for frequency response analysis using the rotordynamic capability. The default is to determine any frequency-dependent terms for each frequency. This option activates the frequency-dependent looping option. Setting the value < 0 uses an ‘average’ frequency formulation. This option avoids using the frequency-dependent looping and results in a shorter execution time. For this option, PARAM,WR3 and PARAM,WR4 must be specified to include rotor damping. See “[Equations Used in Analyses](#)” on page 970 for detailed information. (Default=0)

PARAM, GYROAVG, -1 will bypass frequency-dependent looping in frequency response analysis. See “[Equations Used in Analyses](#)” on page 970 for the equation. Default is 0, which activates frequency looping.

PARAM, WR3 Specifies “average” excitation frequency for calculation of rotor
PARAM, WR4 damping and circulation terms. See “[Equations Used in
Analyses](#)” on page 970 for equations. Default is 0.0, no rotor
damping or circulation terms.

13.13.2 Model Restrictions

The rotors must be modeled as a detached line model in the G-set of the residual structure (superelement 0). Rotors may be modeled as 3-D structures using an upstream superelement and reduced to a line model using static reduction. Note that only static reduction may be used, dynamic reduction using component mode synthesis is not supported.

The rotor line model must be attached to ground or flexible bearings using RBEi elements. If connection between the rotor and bearing or the support structure are detected, a fatal error will be issued. It is important to have the rotor detached from the support structure because the inertial and stiffness properties for the calculation of the gyroscopic terms are determined using the G-size matrices. Connections to ground or support elements may cause incorrect determination of the gyroscopic terms. For example, the following rotor consists of four beam elements with a concentrated mass at the center grid. The end points are connected to ground with springs through RBE2 elements.

```

$
$ ROTOR MODEL
$
GRID, 101, 0, 1.0, 0.0, 0.0
GRID, 102, 0, 2.0, 0.0, 0.0
GRID, 103, 0, 3.0, 0.0, 0.0
GRID, 104, 0, 4.0, 0.0, 0.0
GRID, 105, 0, 5.0, 0.0, 0.0
$
CBEAM, 1001, 1000, 101, 102, 0.0, 1.0, 0.0
CBEAM, 1002, 1000, 102, 103, 0.0, 1.0, 0.0
CBEAM, 1003, 1000, 103, 104, 0.0, 1.0, 0.0
CBEAM, 1004, 1000, 104, 105, 0.0, 1.0, 0.0
PBEAML, 1000, 1001, , ROD,
, 0.01
MAT1, 1001, 2.0E+7, , 0.3, , 1.0E-4
$
CONM2, 2000, 103, 1.0E-3,
, 2.0E-2, , 1.0E-2, , , 1.0E-2
$
$ SUPPORT TO GROUND
$
GRID, 201, 0, 1.0, 0.0, 0.0, 1456
GRID, 205, 0, 5.0, 0.0, 0.0, 1456
$
CELAS1, 3001, 3000, 201, 2
CELAS1, 3002, 3000, 201, 3
CELAS1, 3003, 3000, 205, 2
CELAS1, 3004, 3000, 205, 3
PELAS, 3000, 100.0, 0.02
$

```

```

$ CONNECT ROTOR TO SUPPORT SPRINGS
$
RBE2, 4001, 201, 123456, 101
RBE2, 4002, 205, 123456, 105
$
ENDDATA

```

New Case Control Options

As part of the rotordynamic implementation, the following Case Control commands and parameters can be changed for each subcase:

1. K2PP, M2PP, and B2PP (external matrices)
2. TFL (transfer functions)
3. RGYRO (rotordynamic specification)
4. SDAMP (modal damping)
5. PARAM,G (structural damping)
6. PARAM, ALPHA1 and PARAM, ALPHA2 (Reyleigh damping)
7. DLOAD (dynamic loads)
8. FREQ (frequency range)
9. CMETHOD (complex eigenvalue extraction method)

All the above items can be different in each subcase, but for efficiency reasons, subcases with the same values for items 1 through 6 should be grouped together.

13.13.3 Examples

Following are examples that demonstrate the use of the new rotordynamics features.

Frequency Response

Frequency response analyses are used to determine the rotor/support response to arbitrary excitation with the rotors spinning at user specified rates or excitation that is synchronous with the reference rotor.

Asynchronous Option

Calculation of response with the rotors spinning at set speeds is performed using the asynchronous option (see [Eq. 13-134](#)). The user specifies the rotor speeds, the gyroscopic terms are determined and added to the equation of motion, and the response is determined. Using the model above, the case control and additional bulk data for complex modes analysis with the rotor spinning at 100 RPMs are:

Case Control:

FREQUENCY= 100
 RGYRO= 200

Bulk Data:

```
$
$ FREQUENCY RANGE SPECIFICATION (10 - 500 Hz)
$
FREQ1, 100, 10.0, 10.0, 49
$
$ RGYRO ENTRY
$ specify spin rate in SPEED field
$
```

```
$
$ For gyro term look-up at each frequency, specify reference rotor and spin rate
$
RGYRO    RID    SYNCFLG    REFROTR    SPDUNIT    SPDLOW    SPDHIGH    SPEED
$
RGYRO,   200,   ASYNC,    1,         RPM,       ,          ,    100.0
$
```

or

```
$
$ PARAM, GYROAVG, -1
$
RGYRO    RID    SYNCFLG    REFROTR    SPDUNIT    SPDLOW    SPDHIGH    SPEED
$
RGYRO,   200,   SYNC,    1,         RPM,       0.0,      100.0
$ and specify values for Parameters WR3 and WR4 if required.
PARAM, WR3, 5.0
PARAM, WR4, 5.0
```

\$ ROTORG ENTRY TO SPECIFY ROTOR GRIDS

\$

ROTORG	ROTORID	GRID1	GRID2	...	GRIDn
--------	---------	-------	-------	-----	-------

ROTORG, 1, 101, 102, 103, 104, 105

\$

\$ RSPINR ENTRY

\$

RSPINR	ROTORID	GRIDA	GRIDB	GR	SPDUNT	SPEED1	...	SPEEDn
--------	---------	-------	-------	----	--------	--------	-----	--------

RSPINR, 1, 101, 102, 0.01, RPM, 0.0, 100.0

\$

Synchronous Option

Calculation of frequency response with the excitation synchronous with the reference rotor is performed using the synchronous option. The user specifies the rotor speed range, the gyroscopic terms are determined for each frequency and added to the equation of motion, and the response is determined. There are two methods for determining the gyroscopic terms. The first method calculates the terms for each frequency and adds them to the equation of motion (see [Eq. 13-135](#)). The second method estimates the gyroscopic terms using a least-mean-squared fit of the terms within the analysis range (see [Eq. 13-136](#)). Using the model above, the case control and additional bulk data for frequency response analysis are:

Case Control:

```
FREQUENCY= 100
RGYRO= 200
```

Bulk Data:

```
$
$ FREQUENCY RANGE SPECIFICATION (10 - 500 Hz)
$
FREQ1, 100, 10.0, 10.0, 49
$
$ RGYRO ENTRY
$
```

```
$
$ For gyro term look-up at each frequency, specify reference rotor only
$
RGYRO    RID    SYNCFLG    REFROTR    SPDUNIT    SPDLOW    SPDHIGH    SPEED
RGYRO,   200,   SYNC,      1,          RPM,       0.0,     100.0
$
```

OR

```
$
$ For least-mean square estimate of gyro terms
$ set param, gyroavg to -1
$
PARAM, GYROAVG, -1
$
$ and specify speed range (0 to 100 RPMs)
$
RGYRO    RID    SYNCFLG    REFROTR    SPDUNIT    SPDLOW    SPDHIGH    SPEED
RGYRO,   200,   SYNC,      1,          RPM,       0.0,     100.0
$
```

ROTORG	ROTORID	GRID1	GRID2	...	GRIDn
--------	---------	-------	-------	-----	-------


```

$
$ ROTORG ENTRY TO SPECIFY ROTOR GRIDS
$

```

```
ROTORG, 1, 101, 102, 103, 104, 105
```

```

$
$ RSPINR ENTRY
$

```

RSPINR	ROTORID	GRIDA	GRIDB	GR	SPDUNT	SPEED1	...	SPEEDn
--------	---------	-------	-------	----	--------	--------	-----	--------

```
RSPINR, 1, 101, 102, 0.01, RPM, 0.0, 100.0
```

```
$
```

Complex Modes

Complex modes analyses are used to determine whirl frequencies and critical speeds. Whirl frequencies are modes of a rotor/support system with the rotors spinning at specified rates. Critical speeds are whirl frequencies that coincide with a rotor spin rate.

Asynchronous Option

Calculation of whirl frequencies is performed using the asynchronous option (see [Eq. 13-138](#)). The user specifies the rotor speeds, the gyroscopic terms are determined and added to the equation of motion, and the complex eigenvalues are determined. Using the model above, the case control and additional bulk data for frequency response analysis with the rotor spinning at 100 RPMs are:

Case Control:

```

CMETHOD= 100
RGYRO= 200

```

Bulk Data:

```

$
$ COMPLEX EIGENVALUE EXTRACTION SPECIFICATION
$
EIGC, 100, CLAN,
, 0.0, 0.0, , , , , 20
$
$ RGYRO ENTRY
$ specify spin rate in SPEED field
$

```

RGYRO	RID	SYNCFLG	REFROTR	SPDUNIT	SPDLOW	SPDHIGH	SPEED
-------	-----	---------	---------	---------	--------	---------	-------

```
RGYRO, 200, ASYNC, 1, RPM, , , 100.0
$
$ ROTORG ENTRY TO SPECIFY ROTOR GRIDS
$
```

ROTORG	ROTORID	GRID1	GRID2	...	GRIDn
--------	---------	-------	-------	-----	-------

```
ROTORG, 1, 101, 102, 103, 104, 105
$
$ RSPINR ENTRY
$
```

RSPINR	ROTORID	GRIDA	GRIDB	GR	SPDUNT	SPEED1	...	SPEEDn
--------	---------	-------	-------	----	--------	--------	-----	--------

```
RSPINR, 1, 101, 102, 0.01, RPM, 0.0, 100.0
$
```

Synchronous Option

Calculation of whirl critical speeds is performed using the synchronous option (see). The user specifies the rotor speed range, the gyroscopic terms are determined and added to the equation of motion, and the complex eigenvalues are determined. Using the model above, the case control and additional bulk data for critical speed analysis are as follows:

Case Control:

```
CMETHOD= 100
RGYRO= 200
```

Bulk Data:

```
$
$ COMPLEX EIGENVALUE EXTRACTION SPECIFICATION
$
EIGC, 100, CLAN,
, 0.0, 0.0, , , , , 20
$
$ RGYRO ENTRY
$ specify speed range (For Example, 0 to 100 RPMs)
$
```

RGYRO	RID	SYNCFLG	REFROTR	SPDUNIT	SPDLOW	SPDHIGH	SPEED
-------	-----	---------	---------	---------	--------	---------	-------

```
RGYRO, 200, SYNC, 1, RPM, 0.0, 100.0
$
$ ROTORG ENTRY TO SPECIFY ROTOR GRIDS
$
```

ROTORG	ROTORID	GRID1	GRID2	...	GRIDn
--------	---------	-------	-------	-----	-------

```

ROTORG, 1, 101, 102, 103, 104, 105
$
$ RSPINR ENTRY
$

```

RSPINR	ROTORID	GRIDA	GRIDB	GR	SPDUNT	SPEED1	...	SPEEDn
--------	---------	-------	-------	----	--------	--------	-----	--------

```

RSPINR, 1, 101, 102, 0.01, RPM, 0.0, 100.0
$

```

Static Analysis

For static analysis, the RFORCE entry will generate moments on the rotor grids due to the change of the rotor rotational direction relative to the inertial reference frame, such as when a vehicle pitches or yaws. The RFORCE specifies the rate of change of the maneuver. The loads are calculated and added to the other applied loads.

Case Control:

```

RGYRO= 100
LOAD= 200

```

Bulk Data:

```

$
$ RGYRO ENTRY
$ specify speed of spin rate (100 RPMs)
$

```

RGYRO	RID	SYNCFLG	REFROTR	SPDUNIT	SPDLOW	SPDHIGH	SPEED
-------	-----	---------	---------	---------	--------	---------	-------

```

RGYRO, 200, ASYNC, 1, RPM, , , 100.0
$
$ ROTORG ENTRY TO SPECIFY ROTOR GRIDS
$

```

ROTORG	ROTORID	GRID1	GRID2	...	GRIDn
--------	---------	-------	-------	-----	-------

```

ROTORG, 1, 101, 102, 103, 104, 105
$
$ RFORCE ENTRY
$

```

RFORCE	SID	G	CID	A	R1	R2	R3
--------	-----	---	-----	---	----	----	----

```

RFORCE, 200, 0, 0, 10.0, 1.0, 0.0, 0.0
$

```

Transient Analysis

Transient analysis with rotordynamics was enhanced to include gyroscopic terms and more easily specify rotor imbalance for blade-off analyses. Both nonlinear and direct linear transient response solutions internally calculate the gyroscopic and unbalance force terms for each time step. The mass correction terms (CFLAG option on UNBALC entry) can also be included in the response calculation. Unlike its use in the frequency response, complex modes, and static solutions, the RGYRO Case Control does not reference a RGYRO Bulk Data entry. Instead, the RGYRO Case Control command references an UNBALNC Bulk Data entry. This entry selects the imbalance loading, if any. The ROTORG entries are required to identify the rotor grids and the RSPINT entries are required to specify the rotor spin rate histories.

Case Control:

```
RGYRO= 100
TSTEP= 200
```

Bulk Data:

```
$
$ ROTORG ENTRY TO SPECIFY ROTOR GRIDS
$
```

ROTORG	ROTORID	GRID1	GRID2	...	GRIDn
--------	---------	-------	-------	-----	-------

```
ROTORG, 1, 101, 102, 103, 104, 105
```

```
$
$ SPECIFY SPIN RATE HISTORY
$
```

RSPINT	ROTORID	GRIDA	GRIDB	GR	SPDUNT	TID
--------	---------	-------	-------	----	--------	-----

```
RSPINT, 1, 101, 102, 0.01, RPM, 1000
TABLED1, 1000, 0.0, 0.0, 5.0, 1000.0, 10.0, 6000.0, 20.0, 6000.0, ENDT
```

```
$
$ SPECIFY UNBALANCE LOADING WITH MASS CORRECTION CALCULATION
$
```

UNBALNC	RID	MASS	GRID	X1	X2	X3
	ROFFSET	THETA	ZOFFSET	T _{on}	T _{OFF}	CFLAG

```
UNBALNC, 100, .01, 103, 0.0, 1.0, 0.0,
, 1.0, 30.0, 3.0, 0.0, 100.0, -1
```

13.13.4 Equations Used in Analyses

Frequency Response

For frequency response with asynchronous excitation, Ω for each rotor is constant and can be determined from the rotation speed of the reference rotor, Ω_{ref} , and relative rotation rates specified by the user. The equation of motion to be solved is:

$$\begin{aligned} & \left(-\omega^2 M + \left(i\omega B_s + B_r + \sum_{j=1}^n \Omega_j (\Omega_{ref}) B_j^G \right) \right. \\ & \left. + \left((1 + ig)K_s + iK4_s + \sum_{j=1}^n \left((1 + ig_{rj})K_{rj} + iK4_{rj} + \Omega_j (\Omega_{ref}) \left(K_f^{Cv} + \left(\frac{g_{rj}}{\omega} \right) K_j^{Cgr} + \left(\frac{1}{\omega} \right) K_j^{Cge} \right) \right) \right) \right) u(\omega) = F(\omega) \end{aligned} \quad \text{Eq. 13-134}$$

where the subscript j references the individual rotors.

$1 / \omega$ will be determined for each excitation frequency, similar to frequency-dependent elements.

For the option to bypass the frequency-dependent lookup of rotor speeds, the equation of motion to be solved is:

Eq. 13-135

$$\begin{aligned} & - \left(\omega^2 M + i\omega \left(B_s + B_r + \sum_{j=1}^n \left(\Omega_j (\Omega_{ref}) B_j^G + \left(\frac{g_{rj}}{WR3} \right) K_{rj} + \left(\frac{1}{WR4} \right) K_{rj} \right) \right) \right) \\ & + \left((1 + ig)K_s + iK4_s + \sum_{j=1}^n \left(K_{rj} + \Omega_j (\Omega_{ref}) \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cgr} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) \right) \right) u(\omega) = F(\omega) \end{aligned}$$

where the subscript j references the individual rotors.

WR3 and WR4 are specified using PARAM, WR3 and PARAM, WR4 respectively.

For frequency response with synchronous excitation, the excitation frequency is equal to the spin rate of the reference rotor, $\omega = \Omega_{ref}$. The spin rates of the additional rotors can be determined from the relative spin rates specified by the user. The equation of motion to be solved is:

$$\begin{aligned}
& \left(-\Omega_{ref}^2 M + i\Omega_{ref} \left(B_s + B_r + \sum_{j=1}^n \Omega_j(\Omega_{ref}) B_j^G \right) \right) \\
& + \left((1 + ig)K_s + iK4_s + \sum_{j=1}^n \left((1 + ig_{rj})K_{rj} + iK4_{rj} + \Omega_j(\Omega_{ref}) \left(K_j^{Cv} + \left(\frac{g_{rj}}{\Omega_{ref}} \right) K_j^{Cgr} + \left(\frac{1}{\Omega_{ref}} \right) K_j^{Cge} \right) \right) \right) u(\Omega_{ref}) = F(\Omega_{ref})
\end{aligned}$$

Eq. 13-136

where the subject j references the individual rotors.

$\Omega_j(\Omega_{ref})$ will be determined for each excitation frequency, similar to frequency-dependent elements.

For the option to bypass the frequency-dependent lookup of rotor speeds, Ω for each rotor is written as a linear function dependent on the reference rotor spin rate ($\Omega_j(\Omega_{ref}) = \alpha_j + \beta_j \Omega_{ref}$). The scaling factors, α_j and β_j , are determined from the relative spin rates specified by the user on the RSPINR entries. The Ω_{ref} in the $1/\Omega_{ref}$ terms are replaced by the values of user parameters WR3 and WR4. The equation of motion to be solved is:

Eq. 13-137

$$\begin{aligned}
& \left(-\Omega_{ref}^2 \left(M - i \sum_{j=1}^n \beta_j \beta_j^G \right) \right) \\
& + i\Omega_{ref} \left(B_s + B_r + \sum_{j=1}^n \left(\alpha_j B_j^G + \left(\frac{g_{rj}}{WR3} \right) K_{rj} + \left(\frac{1}{WR4} \right) K4_{rj} - i\beta_j \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cgr} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) \right) \right) \\
& + \left((1 + ig)K_s + iK4_s + \sum_{j=1}^n \left(K_{rj} + \alpha_j \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cgr} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) \right) \right) u(\Omega_{ref}) = F(\Omega_{ref})
\end{aligned}$$

where the subscript j references the individual rotors. WR3 and WR4 are specified using PARAM, WR3 and PARAM, WR4 respectively.

Complex Modes

For complex modes with asynchronous excitation, the spin rate Ω for each rotor is constant and can be determined from user input and the reference rotor spin rate, Ω_{ref} . The equation of motion to be solved is:

$$\begin{aligned} & \left(-\omega^2 M + i\omega \left(B_s + B_r + \sum_{j=1}^n \left(\Omega_j(\Omega_{ref}) B_j^G + \left(\frac{g_{rj}}{WR3} \right) K_{rj} + \left(\frac{1}{WR4} \right) K4_{rj} \right) \right) \right) \\ & + \left((1 + ig)K_s + iK4_s + \sum_{j=1}^n \left(K_{rj} + \Omega_j(\Omega_{ref}) \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cgr} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) \right) \right) u(\omega) = 0 \end{aligned} \quad \text{Eq. 13-138}$$

where the subscript j references the individual rotors.

For complex modes analysis with synchronous excitation, the excitation frequency is equal to the spin rate of the reference rotor, $\omega = \Omega_{ref}$. Ω for each rotor is written as a linear function dependent on the reference rotor spin rate ($\Omega_j(\Omega_{ref}) = \alpha_j + \beta_j \Omega_{ref}$). The scaling factors, α_j and β_j , are determined from the relative spin rates specified by the user. The equation of motion to be solved is:

$$\begin{aligned} & \left(-\Omega_{ref}^2 \left(M - i \sum_{j=1}^n \beta_j B_j^G \right) \right. \\ & \left. + i\Omega_{ref} \left(B_s + B_r + \sum_{j=1}^n \left(\alpha_j B_j^G + \left(\frac{g_{rj}}{WR3} \right) K_{rj} + \left(\frac{1}{WR4} \right) K4_{rj} - i\beta_j \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cgr} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) \right) \right) \right) \\ & + \left((1 + ig)K_s + iK4_s + \sum_{j=1}^n \left(K_{rj} + \alpha_j \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cge} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) \right) \right) u(\Omega_{ref}) = 0 \end{aligned} \quad \text{Eq. 13-139}$$

and the subscript j references the individual rotors.

Nonlinear Transient Response

For nonlinear transient response, Ω is time dependent, the equation of motion to be solved is:

$$\begin{aligned} & M\ddot{u}(t) + \left(B_s + \left(\frac{g}{W3} \right) K_s + \left(\frac{1}{W4} \right) K4_s + \sum_{j=1}^n \left(B_r + \left(\frac{g_{rj}}{WR3} \right) K_r + \left(\frac{1}{WR4} \right) K4_{rj} + \Omega_j(t) B_j^G \right) \right) \dot{u}(t) \\ & + \left(K_s + \sum_{j=1}^n \left(K_{rj} + \Omega_j(t) \left(K_j^{Cv} + \left(\frac{g_{rj}}{WR3} \right) K_j^{Cgr} + \left(\frac{1}{WR4} \right) K_j^{Cge} \right) + \Omega_j(t) K_j^T \right) \right) u(t) = F(t) \end{aligned} \quad \text{Eq. 13-140}$$

where the subscript j references the individual rotors.

The MSC.Nastran Newmark-Beta Method discretizes the following standard equation,

$$M\ddot{u}(t) + B\dot{u}(t) + F(u(t), \dot{u}(t), t) = P(u(t), \dot{u}(t), t) \quad \text{Eq. 13-141}$$

Comparing terms between and Eq. 13-141, the stiffness and gyroscopic terms are incorporated into F .

$$F(u(t), \dot{u}(t), t) = \sum_{j=1}^n (\Omega_j(t) B_j^G) \dot{u}(t) + \left(K + \sum_{j=1}^n (\Omega_j(t) K_j^{Cv} + g_{rj} K_r^{Cgr} + K_j^{Cge} + \dot{\Omega}_j(t) K_j^T) \right) u(t) \quad \text{Eq. 13-142}$$

The gyroscopic matrix, B^G in the above equations, is a skew-symmetric matrix generated with the following matrix defined in the rotor coordinate system:

$$B_r^G = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{11} \\ 0 & 0 & 0 & 0 & -I_{11} & 0 \end{bmatrix} \quad \text{Eq. 13-143}$$

I is determined for each grid along the axis of rotation by taking the appropriate diagonal mass term from the 6X6 mass matrix for the grid. The 6X6 gyroscopic matrix is repeated for each grid along the axis of rotation (individual B^G matrixes are placed along the diagonal of global matrix resulting in a global skew-symmetric matrix).

The viscous damping 'circulation' matrix, K^{Cv} , in the above equations, is based on the rotor-damping matrix. It is formed using the following equation.

$$K_j^{Cv} = (T_I B_j + B_j T_I) \quad \text{Eq. 13-144}$$

The structural damping 'circulation' matrices, K^{Cgr} and K^{Cge} , in the above equations, is based on the rotor-stiffness matrix. It is formed using the following equation.

$$K_j^{Cgr} = (T_I K_{rj} + K_{rj} T_I) \quad \text{Eq. 13-145}$$

$$K_j^{Cge} = (T_I K_{4rj} + K_{4rj} T_I) \quad \text{Eq. 13-146}$$

T_I is a skew-symmetric matrix with the following form in the rotor coordinate system:

$$[T_I]_r = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0 \\ 0 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & -0.5 & 0 \end{bmatrix} \quad \text{Eq. 13-147}$$

The 6X6 T_I matrix is repeated for each grid along the axis of rotation (individual T_I matrixes are placed along the diagonal of global matrix resulting in a global skew-symmetric matrix).

The tangential matrix, K^T in the above equations, is a skew-symmetric matrix generated with the following matrix defined in the rotor coordinate system:

$$K_r^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{11} \\ 0 & 0 & 0 & 0 & -I_{11} & 0 \end{bmatrix} \quad \text{Eq. 13-148}$$

I_{11} is determined for each grid along the axis of rotation by taking the appropriate diagonal mass term from the 6X6 mass matrix for the grid transformed to the rotor coordinate system. The 6X6 Tangent matrix is repeated for each grid along the axis of rotation (individual K_r^T matrixes are placed along the diagonal of global matrix resulting in a global skew-symmetric matrix).

In addition to the applied time-varying forces $P(t)$ in [Eq. 13-141](#). The time-varying forces and moments due to a mass imbalance at grid k , in the rotor-force coordinate system, are:

$$\begin{Bmatrix} \bar{F}_1(t) \\ \bar{F}_2(t) \end{Bmatrix}_k = WTMASS \cdot \begin{Bmatrix} m_k(t)r_k(t)(\Omega_j^2(t)\cos\theta_{jk}(t) + \dot{\Omega}_j(t)\sin\theta_{jk}(t)) \\ m_k(t)r_k(t)(\Omega_j^2(t)\sin\theta_{jk}(t) - \dot{\Omega}_j(t)\cos\theta_{jk}(t)) \end{Bmatrix} \quad \text{Eq. 13-149}$$

$$\begin{Bmatrix} \bar{M}_1(t) \\ \bar{M}_2(t) \end{Bmatrix}_k = offset_3(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{Bmatrix} \bar{F}_1(t) \\ \bar{F}_2(t) \end{Bmatrix}_k \quad \text{Eq. 13-150}$$

where:

$m_k(t)$ = specified by the user

$r_k(t)$ = specified by the user

$\Omega_j(t)$ = specified by the user

$\dot{\Omega}_j(t)$ = is derived from $\Omega_j(t)$

$\theta_{jk}(t) = \int_0^t \Omega_j(t) dt + \theta_k$, θ measured from axis 1 to axis 2

θ_k = specified by the user

j = the rotor on which the grid is located

If CFLAG on the UNBALNC entry is < 0 , the forces and moments calculated in [Eq. 13-149](#) and [Eq. 13-150](#) are augmented to correct the equation of motion for the

$$\begin{Bmatrix} F_1(t) \\ F_2(t) \\ F_3(t) \end{Bmatrix}_k = \begin{Bmatrix} \bar{F}_1(t) \\ \bar{F}_2(t) \\ \mathbf{0} \end{Bmatrix}_k + WTMASS \cdot \begin{Bmatrix} -m_k(t)\ddot{u}_1(t)_k \\ -m_k(t)\ddot{u}_2(t)_k \\ -m_k(t)\ddot{u}_3(t)_k \end{Bmatrix} \quad \text{Eq. 13-151}$$

Where displacements are transformed from the global to rotor-force coordinate system using the transformation matrix stored on the UNBALNC entry.

$$\begin{Bmatrix} \ddot{u}_1(t) \\ \ddot{u}_2(t) \\ \ddot{u}_3(t) \end{Bmatrix}_k = [T]_{global \rightarrow rotor-force} \begin{Bmatrix} \ddot{u}_{T1}(t) \\ \ddot{u}_{T2}(t) \\ \ddot{u}_{T3}(t) \end{Bmatrix}_k \quad \text{Eq. 13-152}$$

The above correction accounts for the change in the translation mass due to the imbalance mass.

Eq. 13-153

Where ϕ_i is the finite rotation about the i -axis with

$$\begin{Bmatrix} \phi_1(t) \\ \phi_2(t) \\ \phi_3(t) \end{Bmatrix}_k = [T]_{global \rightarrow rotor-force} \begin{Bmatrix} \phi_{T1}(t) \\ \phi_{T2}(t) \\ \phi_{T3}(t) \end{Bmatrix}_k \quad \text{Eq. 13-154}$$

$$\begin{aligned}
\begin{Bmatrix} M_1(t) \\ M_2(t) \\ M_3(t) \end{Bmatrix}_k &= \begin{Bmatrix} \bar{M}_1(t) \\ \bar{M}_2(t) \\ \mathbf{0} \end{Bmatrix}_k + WTMASS \cdot \begin{Bmatrix} -m_k(t)(r_k(t) \sin \theta_{jk}(t))^2 \ddot{\phi}_1(t)_k \\ -m_k(t)(r_k(t) \cos \theta_{jk}(t))^2 \ddot{\phi}_2(t)_k \\ -m_k(t)r_k^2(t) \ddot{\phi}_3(t)_k \end{Bmatrix} \\
&+ \begin{Bmatrix} -2m_k(t)(r_k(t) \sin \theta_{jk}(t))^2 \Omega_j \dot{\phi}_1(t)_k \\ 2m_k(t)(r_k(t) \cos \theta_{jk}(t))^2 \Omega_j \dot{\phi}_2(t)_k \\ \mathbf{0} \end{Bmatrix} + \begin{Bmatrix} -m_k(t)(r_k(t) \sin \theta_{jk}(t))^2 \dot{\Omega}_j \phi_1(t)_k \\ m_k(t)(r_k(t) \cos \theta_{jk}(t))^2 \dot{\Omega}_j \phi_2(t)_k \\ \mathbf{0} \end{Bmatrix}
\end{aligned}$$

In the above equation, the first correction term accounts for the change in rotational inertia due to the imbalance mass, the second term corrects the gyroscopic matrix, B^G , the third term corrects the tangent matrix, K^T .

Collecting like terms in the above equation produces:

$$\begin{aligned}
\begin{Bmatrix} M_1(t) \\ M_2(t) \\ M_3(t) \end{Bmatrix}_k &= \begin{Bmatrix} \bar{M}_1(t) \\ \bar{M}_2(t) \\ \mathbf{0} \end{Bmatrix}_k && \text{Eq. 13-155} \\
&+ WTMASS \cdot \begin{Bmatrix} -m_k(t)(r_k(t) \sin \theta_{jk}(t))^2 (\ddot{\phi}_1(t)_k + 2\Omega_j(t) \dot{\phi}_1(t)_k + \dot{\Omega}_j(t) \phi_1(t)_k) \\ -m_k(t)(r_k(t) \cos \theta_{jk}(t))^2 (\ddot{\phi}_2(t)_k - 2\Omega_j(t) \dot{\phi}_2(t)_k - \dot{\Omega}_j(t) \phi_2(t)_k) \\ -m_k(t)r_k^2(t) \ddot{\phi}_3(t)_k \end{Bmatrix}
\end{aligned}$$

The forces and moments are transformed from the rotor-force coordinate system using the transformation matrix stored on the UNBALNC entry.

$$\begin{Bmatrix} F_x(t) \\ F_y(t) \\ F_z(t) \end{Bmatrix}_{global} = [T]_{global \rightarrow rotor-force}^T \begin{Bmatrix} F_1(t) \\ F_2(t) \\ F_3(t) \end{Bmatrix}_{rotor-force} \quad \text{Eq. 13-156}$$

$$\begin{Bmatrix} M_x(t) \\ M_y(t) \\ M_z(t) \end{Bmatrix} = [T]_{global \rightarrow rotor-force}^T \begin{Bmatrix} M_1(t) \\ M_2(t) \\ M_3(t) \end{Bmatrix}_{rotor-force} \quad \text{Eq. 13-157}$$

Static Analysis

For static analysis, in addition to the centripetal forces, the RFORCE entry will generate moments on the rotor grids due to the change of the rotor rotation vector relative to the inertial reference frame. The moments in the rotor coordinate system are calculated as follows:

$$\begin{Bmatrix} M_1 \\ M_2 \\ M_3 \end{Bmatrix}_r = -\Omega [I_{kk}]_r \begin{Bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{Bmatrix}_r \quad \text{Eq. 13-158}$$

Ω is the rotor spin rate. The rotation terms, $(\omega_1, \omega_2, \text{ and } \omega_3)$, are the components of the rotation vector from the RFORCE entry transformed to the rotor coordinate system. $\omega_1, \omega_2, \omega_3$ and units are radians/unit time.

$$[I_{kk}]_r = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I_{11} \\ 0 & -I_{11} & 0 \end{bmatrix} \quad \text{Eq. 13-159}$$

The moments are transformed back to the global coordinate system and added to the static loading vector. The moments can also be calculated in the global system using the following:

$$\begin{Bmatrix} M_1 \\ M_2 \\ M_3 \end{Bmatrix}_{global} = -\Omega [I_{kk}]_{global} \begin{Bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{Bmatrix}_{global} \quad \text{Eq. 13-160}$$

where Ω is the rotor spin rate, $[I_{kk}]_g$ is the transformation of $[I_{kk}]_r$ from rotor to global coordinates and the rotation terms, $(\omega_1, \omega_2, \text{ and } \omega_3)_g$, are the components of the rotation vector from the RFORCE entry in the global coordinate system.

13.14 References

1. Narayanaswami, R., and H. M. Adelman, "Evaluation of the Tensor Polynomial and Hoffman Strength Theories for Composite Materials," *Journal of Composite Materials*, Vol. II, 1977, p. 366.
2. Rivlin, R. S. and Saunders, D. W., "Large Elastic Deformations of Isotropic Material VII: Experiments on the Deformation of Rubber," *Phil. Trans. R. Soc. London*, A243, 251-288, 1951.
3. Treloar, L. R. G. *The Physics or Rubber Elasticity*, 3rd ed., Oxford: Clarendon Press, 1975.
4. Ogden, R. W. "Elastic Deformations of Rubberlike Solids: in "Mechanics of Solids," The Rodney Hill 60th Anniversary Volume, H.G. Hopkins and M. J. Sewell, eds., Oxford: Pergamon Press, pp. 499-537.
5. J. T. Oden, *Finite Elements of Nonlinear Continua*, McGraw-Hill, 1972.
6. Roark, R. J. and Young, W. C. *Formulas for Stress and Strain*. 5th edition. McGraw-Hill, Inc., 1975, p. 594.

I N D E X

MSC.Nastran Reference Manual

Numerics

3D Contact, 737

A

ACCELERATION, 28, 29
ACMODL, 780
ACMS, 898, 899
ACSRCE, 297, 756
ADAMSMNF, 906
ADAPT, 218, 834, 839, 840, 869
adaptive bisection, 710, 714
ADPCON, 209
ADUM8, 197
ADUM9, 197
ADUMi, 51
aeroelastic response, 18, 28
AEROF, 26, 28
ANALYSIS, 11, 20, 21, 22
angular acceleration, 287
angular velocity, 287
anisotropic, 737
APRESSURE, 26
arclength method, 674
Arruda-Boyce rubber, 737
ASCOUN, 753
ASETi, 333
ASSIGN, 490, 497
Augmented Lagrange Multiplier Method,
179
AUTOASGN, 493
automatic time stepping, 707
AUTOSPC, 140, 329
axial damping element, 229
axisymmetric elements, 49, 195
AXSLOT, 795

B

B2PP, 17, 32
BAR, 253, 254

bar element, 68
BAROR, 69
BCBODY, 744, 748
BCBOX, 744
BCHANGE, 744
BCMATL, 744
BCMOVE, 744
BCONP, 208
BCONTACT, 743, 748
BCPARA, 744
BCPROP, 744
BCTABLE, 744, 748
BEAM, 253, 254
beam element, 58
BEGIN BULK delimiter, 4
BFRIC, 208
BLSEG, 208
BOUTPUT, 208
BSBRCD subroutine, 88
BSBRGD subroutine, 88
BSBRID subroutine, 88
BSBRPD subroutine, 88
BSBRT subroutine, 88
BSCON subroutine, 88
BSGRQ subroutine, 88
BSMSG subroutine, 88
BSURF, 744, 748
buckling analysis, 17, 27, 670, 679
Bulk Data Section, 4
BUSH1D, 230, 231, 632
BWIDTH, 208

C

CACABSF, 755
Case Control commands for SOLs
ADAPT, 12, 16
GRID, 12
METHOD(FLUID), 16
OUTRCV, 12, 16
SDMAPING(FLUID), 16
SMETHOD, 12

- SPC, 12
- SUPPORT1, 12
- VUGRID, 16
- Case Control Section, 4, 11
- CAXIF2, 795
- CAXIF3, 795
- CAXIF4, 795
- CAXIFi, 52, 54, 56
- CBAR, 52, 54, 56, 58, 68
- CBARAO, 286
- CBEAM, 52, 54, 56, 58, 63, 65
- CBEND, 52, 54, 56, 58, 69
- CBUSH, 52, 54, 56, 225
 - frequency dependency, 225
 - load-displacement dependency, 225
- CBUSH1D, 52, 54, 56, 229
- CCONEAX, 49, 52, 54, 56
- CDAMP1, 194
- CDAMP2, 194
- CDAMP3, 194
- CDAMP4, 194
- CDAMP5, 194
- CDAMPi, 52, 54, 56
- CDUMi, 51
- CELAS1, 193
- CELAS2, 193
- CELAS3, 193
- CELAS4, 193
- CELASi, 52, 54, 56
- CEND delimiter, 4
- CFLUIDi, 52, 54, 56
- CGAP, 52, 54, 56, 203, 567
- CHACAB, 754
- CHACBR, 756
- CHBDYG, 54
- CHBDYi, 52, 56
- CHBDYP, 54
- CHEXA, 52, 54, 56, 163, 255
- CLOAD, 23, 308
- CMASS1, 194
- CMASS2, 194
- CMASS3, 194
- CMASS4, 194
- CMASSi, 52, 54, 56
- CMETHOD, 18, 684
- commands, 12
- comments, 4
- complex eigenvalue analysis, 17, 27
- component mode synthesis, 334
- Composite Rotations, 177
- Composites
 - MAT8, 532
- composites
 - data recovery, 546
 - failure criteria, 534
 - failure indices, 547
 - Hill's theory, 548
 - Hoffman's theory, 549
 - lamina, 531
 - MAT2, 532
 - material properties, 532
 - maximum strain theory, 550
 - PCOMP, 532
 - PSHELL, 532
 - Tsai-Wu theory, 549
- concentrated mass elements, 210
- CONM1, 46, 51, 210
- CONM2, 46, 51, 210
- CONMi, 52, 54, 56
- CONROD, 52, 54, 56, 58, 73
- constraint elements, 51
- contact, 655
- CONV, 658
- convergence, 655
- CONVERT, 497
- COPYR, 739
- CORD1C, 42
- CORD1R, 42
- CORD1S, 42
- CORD2C, 42
- CORD2R, 42
- CORD2S, 42
- Coupled fluid-structural analysis
 - fluid model definition, 754
 - loading, 756
 - nonmatching mesh, 753
 - panel, 754
- coupled fluid-structural analysis, 753
 - acoustic absorber, 754
 - acoustic barrier, 756
 - acoustic source, 756
 - enforced pressures, 756
 - fluid-structure interface, 757
 - matching mesh, 753
 - normal modes analysis, 757
 - output and plotting, 758
 - pressure method, 753
 - single-point constraints, 757
 - superelement analysis, 758
- COUPMASS, 58, 69, 73, 132
- CPENTA, 53, 54, 56, 163, 255
- CQUAD4, 53, 54, 56, 131, 254
- CQUAD8, 53, 54, 56, 132
- CQUADR, 53, 54, 56, 132, 254

CQUADX, 49, 195
 CRAC2D, 52, 54, 56, 197
 CRAC3D, 52, 54, 56, 197
 crack, 737
 crack tip elements, 197
 Craig-Bampton, 907
 creep, 722, 738
 CROD, 53, 54, 56, 58, 73
 CSHEAR, 53, 54, 56, 130
 CSLOT3, 795
 CSLOT4, 795
 CSLOTi, 53, 55, 57
 CTETRA, 53, 55, 57, 163, 255
 CTRIA3, 53, 55, 57, 131, 254
 CTRIA6, 53, 55, 57, 131
 CTRIAR, 53, 55, 57, 131, 254
 CTRIAX, 49, 195
 CTRIAX6, 49, 53, 55, 57, 195
 CTUBE, 53, 55, 57, 58, 73
 CVISC, 51, 53, 55, 57, 74
 CWELD, 53, 55, 57, 239, 240
 cyclic symmetry, 21

- dihedral symmetry, 804
- N-gons, 805
- reflective symmetry, 803, 806
- rotational symmetry, 804
- theory, 809
- three-dimensional bodies, 806

 CYJOIN, 823
 cylindrical coordinate system, 42, 44
 CYSYM, 823

D

DAREA, 21, 294, 295, 297, 298, 313
 database, 488
 database migration, 495
 DATAREC, 839
 DBCLEAN, 499
 DBLOAD, 497, 499, 500
 DBLOCATE, 493
 DBSETDEL, 492
 DBsets, 488

- DBALL, 488
- MASTER, 488
- SCRATCH, 488
- USROBJ, 489
- USRSOU, 489

 DBUNLOAD, 499, 500
 DCMP, 904
 DDRMM, 74, 194

DEFORM, 12, 17, 290
 DEFUSET, 338
 Degree of Freedom Sets

- Bulk Data entries, 325
- definitions, 322

 Degree-of-Freedom domain decomposition, 888
 DELAY, 294, 295, 313
 delivery database, 491
 DEQATN, 218, 842, 852
 design optimization, 22
 differential stiffness, 679, 680
 direct input matrices, 46
 Direct Matrix Abstraction Program, 486
 DISPLACEMENT, 26, 27, 28, 29
 displacement coordinate system, 42
 dissimilar meshes, 235
 Distributed Memory Parallel

- DMP, 891

 DLOAD, 18, 19, 21, 32, 294, 295, 312, 313
 DMAP, 486
 DMIG, 46, 215, 334
 DMP, 888, 897, 898, 900
 DOF based Domain Decomposition, 891
 domain decomposition, 738
 DOMAINSOLVER, 889, 892, 893, 896, 900, 902, 904

- ACMS, 894
- FREQ, 894
- MODES, 894
- STAT, 894

 DPHASE, 294
 dummy elements, 51
 DYNPCF, 410

E

EDE, 28, 29
 EIGC, 18, 684
 eigenvalue analysis, 674
 EIGR, 15, 21
 EIGRL, 15, 21
 EKE, 26, 28, 29
 element deformation, 290
 element geometry checks, 252
 element offsets, 43
 element strain energy, 719
 ELEMID, 239
 ELPAT, 239
 ELSDCON, 464, 479
 ENDDATA delimiter, 4

enforced motion, 292
 EPOINT, 45
 EPSP, 658
 EPSU, 658
 EPSW, 658
 ESE, 26, 27, 28, 29, 717, 719, 722
 Euler angles, 176
 Executive Control Section, 4
 Executive Control statements, 10

F

F06, 742
 f06, 739
 F11, 742
 failure criteria, 552, 554, 555
 HILL, 552
 HOFFMAN, 554
 TSAI-Wu, 555
 Failure Index (FI), 552
 FBS, 904
 FEEDGE, 45, 216, 218, 302, 831, 838, 841, 846
 FEFACE, 45, 217, 219, 302, 831, 838, 841, 846
 FEFACES, 863
 File Management Section (FMS), 4, 9
 flexibility factors, 71
 Flexible-Flexible surfaces, 737
 Flexible-Rigid surfaces, 737
 FLUTTER, 18
 FLUX, 30, 31
 FMETHOD, 18
 foam, 737
 follower force, 709
 follower stiffness, 683
 FOLLOWWK, 677
 FORCE, 26, 27, 28, 29, 285, 306
 FORCE1, 285, 306
 FORCE2, 285, 306
 FORCEAX, 285
 FREQ, 18, 294
 FREQ1, 18, 294
 FREQ2, 18, 294
 FREQUENCY, 18, 32
 frequency dependency, 225
 frequency response, 18, 28

G

GAP, 204
 gap element, 203
 gaps, 655

Gasket, 738
 GENEL, 50, 211, 215
 general element, 211
 GEOMCHECK, 253, 255, 256
 geometric domain decomposition, 891
 GEOMIN, 216
 Gimbal angles, 176
 global coordinate system, 43
 global/local analysis, 831
 GMBC, 219, 302
 GMCORD, 45, 219, 852
 GMCURV, 45, 217, 219, 302, 831, 842, 850, 863
 GMLOAD, 219, 302, 841, 844, 852
 GMNURB, 744
 GMODES, 892
 GMSPC, 218, 302, 841, 863
 GMSURF, 45, 217, 218, 302, 831, 842, 850
 GPFORCE, 26, 27, 30, 717, 719, 722
 GPSDCON, 464, 479
 GPSTRAIN, 26, 27
 GPSTRESS, 26, 27, 29, 464, 474, 479, 482
 GRAV, 287
 gravity load, 287
 GRDPNT, 35
 GRDSET, 44
 GRID, 41, 42
 grid point force, 718
 grid point stress, 464, 466, 467, 481
 exception points, 472
 geometric method, 470
 output, 475
 topological method, 467
 grid points, 41, 46
 GRIDF, 795
 GRIDID, 239
 GRIDS, 795
 Guyan reduction, 333

H

HARMONICS, 823
 HDOT, 31
 heat transfer, 655
 h-elements, 829
 HEXA, 253
 Hierarchic Distributed Parallel Lanczos
 Method, 891
 Hierarchic Method, 889
 HOUTPUT, 824
 hydroelastic modeling, 794

hyperelastic elements, 220
 hyperelastic materials, 132
 hyperelasticity
 curve-fitting algorithm, 588
 nonlinear stress output, 616

I

IC, 19, 21
 INCLUDE statement, 4
 inertia relief, 331, 391, 416, 422
 INIT, 490
 INIT MASTER, 492
 initial conditions, 312
 INREL, 407
 interface elements, 232
 dissimilar meshes, 235
 IPSTRAIN, 744
 ISTRESS, 744

K

K2GG, 12, 15, 21, 23
 K2PP, 17, 32
 K6ROT, 140

L

Lagrange Elimination Method, 174
 Lagrange elimination method, 181
 Lagrange rigid element, 173
 Left Rotation, 177
 LGDISP, 619, 677
 line elements, 58
 Linear Elimination Method, 179
 linear steady state heat transfer, 20, 30
 LOAD, 12, 17, 20, 23
 LOADCYH, 300, 824
 LOADCYN, 300, 824
 LOADCYT, 824
 load-displacement dependency, 225
 LOADSET, 19, 23, 24, 294, 295, 297, 298, 309,
 312
 local coordinate systems, 41
 LOOPID, 20, 21, 23, 24, 714
 LSEQ, 19, 24, 294, 297, 298, 300, 308, 313

M

M2GG, 12, 15, 21, 23, 215

M2PP, 17, 32
 MacNeal/Harder test, 145
 MARCIN, 744
 MARCOUT, 744
 mass properties, 46
 MAT1, 49, 63, 69
 MAT2, 49
 MAT3, 50
 MAT8, 50, 532
 MAT9, 50
 MATEP, 744
 material properties, 49
 MATG, 744
 MATGEN module, 338
 MATGPR module, 338
 MATHE, 744
 MATHED, 744
 MATHP, 50, 220
 MATMOD module, 338
 MATORT, 744
 MATS1, 50, 58
 MATT, 744
 MATT1, 49
 MATT2, 49
 MATT3, 50
 MATT9, 50
 MATTE, 744
 MATTEP, 744
 MATTG, 745
 MATTORT, 745
 MATTVE, 745
 MATV, 744
 MATVP, 744
 maximum stress (strain), 557
 mechanism, 386
 Message Passing Interface
 MPI, 889, 893
 METHOD, 15, 21, 24, 684
 METHOD(FUID), 757
 METHOD(STRUCTURE), 757
 MODES, 15
 MOMAX, 285
 MOMENT, 285, 306
 MOMENT1, 285, 306
 MOMENT2, 285, 306
 MPC, 15, 31, 168, 327
 MPCADD, 327
 MPCAX, 327
 MPCF, 27, 28, 29
 MSC.ADAMS, 906, 908
 MSC.Marc, 655, 738, 739
 MSGLIMIT, 253

MSGTYPE, 253
 multipoint constraint, 327

N

nastran command, 2
 NASTRAN statement, 3, 8
 Newton's method, 674, 683
 Newton-Raphson method, 175
 NINC, 658
 NLAUTO, 745
 NLDAMP, 745
 NLLOAD, 31, 312
 NLPARM, 23, 652, 655, 657
 NLPCI, 662, 722
 NLRGAP, 314, 710
 NLSTRAT, 745
 NLSTRESS, 26
 NOCOMPS, 533
 NOLIN, 707, 710
 NOLIN1, 296, 710
 NOLIN2, 296, 710
 NOLIN3, 296, 710
 NOLIN4, 296, 710
 NONLINEAR, 312
 NOLINi, 230, 314
 NONLINEAR, 19, 21, 29
 nonlinear analysis, 306

- 3-D slideline contact, 639
- elastic-plastic material, 594
- element library, 567
- elements, 600
- gap element, 633
- geometric nonlinearity, 567, 599
- hyperelastic elements, 614
- hyperelastic material, 567, 585
- material nonlinearity, 567, 583
- modeling, 580
- nonlinear analysis with superelements, 715
- nonlinear buckling analysis, 670
- nonlinear elastic material, 584
- nonlinear transient response analysis, 691
- parameters, 577
- plot codes, 580
- recommendations, 582
- restarts, 700
- small strain elements, 610
- SOL 106, 570
- SOL 129, 570

SOL 153, 570
 SOL 159, 570
 static analysis, 649
 subcase structure, 572
 temperature dependent material, 598
 viscoelastic material, 598
 nonlinear static analysis, 23
 nonlinear steady state heat transfer, 20
 nonlinear transient heat transfer, 21, 31
 nonlinear transient response analysis, 24
 nonstructural mass, 258
 normal modes analysis, 15, 26
 NOUTPUT, 824
 NTHICK, 745

O

Offsets, 124
 OFREQUENCY, 19, 28, 412
 Ogden rubber, 737
 OLOAD, 26, 27, 28, 29, 30, 31
 OMITAX, 333
 OMITi, 333
 OP2, 742
 op2, 739
 orthotropic, 737
 OTIME, 412
 OUTPUT, 218, 835, 840, 859
 OUTPUT(PLOT), 26, 27, 28, 29
 OUTPUT(POST), 464, 474, 479, 482
 OUTPUT(XYPLOT), 28, 30
 OUTR, 739
 OUTRCV, 219, 835, 839

P

P2G, 12, 21, 23
 PACABS, 755
 PACBAR, 756
 PANEL, 754
 PARAM, 552

- BUCKLE, 682
- BUCKLE,2, 671, 673
- FKSYMFAC, 678, 684, 732
- FOLLOWK, 732
- GRDPNT, 920, 923
- LANGLE, 183, 731
- LGDISP, 731
- LGSTRN, 745
- LMFACT, 183
- MALIASij, 746

MARCASUM, 746
 MARCCONi, 745
 MARCDILT, 746
 MARCDISi, 745
 MARCEKND, 746
 MARCGAUS, 746
 MARCIAMN, 746
 MARCMEM, 745
 MARCOPT, 745
 MARCPOS, 745
 MARCPOST, 746
 MARCRBAL, 746
 MARCRBAR, 746
 MARCRBE2, 745
 MARCRBE3, 745
 MARCSHLT, 746
 MARCSINC, 746
 MARCT16, 746
 MARCT19, 746
 MARCVERS, 745
 MARSIZi, 745
 MARSUMY, 746
 MARUPDAT, 746
 MAXLP, 732
 MRALIAS, 746
 MRCONRES, 746
 MRSRING, 746
 NDAMP, 709
 NLAYERS, 732
 NLTOL, 655, 656, 659, 732
 NMLOOP, 906
 NOELOF, 719
 NOELOP, 719
 ORIGID, 183
 PENFN, 183
 PH2OUT, 732
 QRMETH, 149
 SEKD, 714
 SHLDAMP, 162
 SRCOMPS, YES/NO, 552
 STRESS, 745
 WTMASS, 920, 923
 XFLAG, 717
 PARAMARC, 745
 PARAML module, 338
 PARTOPT, 892
 PARTPAT, 239
 PBAR, 69, 75
 PBARL, 75
 PBCOMP, 58, 61, 62
 PBEAM, 58, 61, 63, 65, 75
 PBEAML, 75
 PBEND, 69
 PBUSH1D, 229
 PBUSHT, 225
 PCH, 742
 pch, 739
 PCOMP, 532
 PDAMP, 194
 PDUMi, 51
 PELAS, 193
 p-element analysis, 45
 p-elements, 216, 236, 237
 adaptivity, 834
 applications, 829
 boundary conditions, 833
 combining p- and h-elements, 830
 data recovery, 835
 elements, 832
 error estimator, 830
 example problems, 837
 geometry, 830, 831
 loads, 833
 materials, 833
 solution sequences, 831
 PENTA, 253
 PGAP, 204
 PLOAD, 307
 PLOAD1, 58, 286
 PLOAD2, 286, 307
 PLOAD4, 286, 307
 PLOADX1, 195, 286
 PLOTSUP, 503
 PLPLANE, 220
 PLSOLID, 220
 POINT, 831, 840
 POINTAX, 285
 POST, 517
 PostScript plot, 513
 PRAC2D, 198
 PRESAX, 286
 PRESSURE, 758
 prestressed normal mode, 674, 677
 PROD, 73
 PROJECT, 494
 PSET, 219
 PSHEAR, 130
 PSHELL, 132, 135
 PSOLID, 50, 163
 PTUBE, 73
 PVAL, 219, 834, 840, 869
 PVISC, 74

Q

QBDYi, 20, 21
 QHBDY, 20, 21
 QUAD4, 140, 144, 253
 QUAD8, 144
 QUADR, 140, 144, 253
 quaternion, 176
 QVECT, 20, 21
 QVOL, 20, 21

R

RANDOM, 19
 random response, 18, 28, 294
 RANDPS, 19, 294
 RANDT1, 294
 RANDTi, 19
 RBAR, 51, 135, 167, 328
 RBAR1, 184, 185
 RBE1, 51, 168, 328
 RBE2, 51, 167, 328
 RBE3, 51, 168, 328
 rectangular coordinate system, 42, 43
 REPCASE, 12, 15, 16, 20, 32
 RESTART, 494, 745
 Restart, 728
 restarts, 667
 RFORCE, 287, 722
 Right Rotation, 177
 RIGID, 183, 184
 rigid elements, 172
 rigid format, 343
 RJOINT, 184, 186
 RLOAD1, 292, 293, 294, 297
 RLOAD2, 292, 293, 294, 297
 rod element, 73
 Rodrigues parameters, 176
 rotational stiffness, 172
 RROD, 51, 167, 328
 RSPLINE, 51, 168, 328
 RSSCON, 168, 328
 RTRPL1, 184
 RTRPLT, 51, 167, 328
 RTRPLT1, 185
 rubber, 738
 RVDOF, 436
 RVDOF1, 436

S

SACCELERATION, 28, 29
 scalar elements, 193
 SDAMPING, 17, 32
 SDISP, 31
 SDISPLACEMENT, 28, 29, 31
 SEALL, 25
 SEPLOT, 503
 SETS DEFINITION, 839
 SEUSETi,U6, 436
 Shared Memory Parallel
 SMP, 888
 shear panel, 130
 shear relief, 64
 shell elements, 131
 single point constraint, 329
 Slideline contact
 guidelines, 206
 slideline contact
 input, 208
 master line, 204
 penetration, 206
 slave line, 204
 slideline contact region, 204
 SLOAD, 285, 306
 SLOOPID, 21, 24
 soil, 738
 SOL 106, 651
 SOL 129, 691
 SOL 400, 187, 721
 SOL 600, 737
 SOL 99, 694
 SOL DBTRANS, 500
 solution sequence, 342
 rigid format, 342, 343
 SORT1, 25
 SORT2, 25
 SPC, 15, 31, 292, 329
 SPC1, 293, 329
 SPCADD, 329
 SPCAX, 292, 329
 SPCD, 292
 SPCF, 26, 27, 28, 29
 SPCFORCE, 30
 spherical coordinate system, 42, 44
 split database operations, 493
 SPOINT, 44
 static analysis, 11, 26
 STATSUB, 17
 steady state heat transfer, 30
 STEP, 721, 722

STIME, 21, 24
 STRAIN, 26, 28
 strain energy, 716
 strength ratio, 566
 STRESS, 26, 27, 28, 29, 464
 stress discontinuities, 464, 483
 stress intensity factor, 198
 STRFIELD, 482
 SUBCASE, 13, 14, 16, 721, 722
 subcase combinations, 300
 subcases, 12, 15, 17
 SUBCASID, 20
 SUBCOM, 14, 32, 289, 290, 300, 808
 SUBID, 20, 23
 subroutines
 BSBRGD, 88
 BSBRID, 88
 BSBRPD, 88
 BSBRT, 88
 BSCON, 88
 BSGRQ, 88
 BSMSG, 88
 SUBSEQ, 32
 SUPAX, 331
 SUPER, 25, 33
 superelement, 24, 35
 superelement analysis, 444
 SUPPORT, 331
 SUPPORT1, 15, 31, 331
 SURFACE, 464, 475, 479, 482
 SVECTOR, 26, 27
 SVELO, 31
 SVELOCITY, 28, 29, 31
 SYM, 14, 290
 SYMCOM, 14, 289, 290, 300, 808
 system cell
 366, 731
 401, 731

T

T16, 742
 T19, 742
 TABDMP1, 17
 TABLE3D, 219
 TABLEDi, 294, 295, 313
 TABLEM1, 49
 TABLEM2, 49
 TABLEM3, 49
 TABLEM4, 49
 TABLES1, 50

TABLEST, 50
 TABPRT module, 338
 TABRNDi, 294
 tapered beam, 67
 TEMP, 20, 21, 30, 31, 46, 288, 307
 TEMP(LOAD), 289
 TEMPAX, 46, 288
 TEMPD, 20, 21, 46, 288, 307
 TEMPERATURE, 12, 17
 TEMPERATURE(INIT), 23
 TEMPERATURE(INITIAL), 20
 TEMPERATURE(LOAD), 23
 temperatures, 288
 TEMPF, 219
 TEMPP1, 288, 308
 TEMPP2, 288
 TEMPP3, 288, 308
 TEMPRB, 288, 308
 TETRA, 253
 TF, 334
 TFL, 32
 TFLAG, 159
 THERMAL, 30, 31
 thermal field, 46, 288
 thermal load, 288
 TIC, 19
 TLOAD1, 292, 293, 295, 297, 313, 709
 TLOAD2, 292, 293, 295, 297, 313
 TLOADi, 18, 21
 transient response, 19, 29
 TRIA3, 253
 TRIAR, 140, 253
 TRIM, 12
 TSTEP, 19, 32
 TSTEPNL, 21, 24, 231, 695

U

UMERGE module, 338
 UMERGE1 module, 338
 UPARTN module, 338
 USET, 338
 USET1, 338
 USETi,U6, 436

V

VEC module, 338
 VECTOR, 26, 27
 VELO, 31
 VELOCITY, 28, 29, 31

virtual mass (MFLUID), 899
virtual mass method, 794
Visco-elastic, 738
VOLUME, 464, 475, 479, 482
VUGRID, 839

W

warping, 64, 66
WHERE, 497

X

XDB, 742
xdb, 739