MSC.Nastran 2005

Release Guide

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, Shinjyku-Ku
Tokyo 160-0023, JAPAN
Telephone: (03)-6911-1200
Fax: (03)-6911-1201

Worldwide Web

www.mscsoftware.com

NA*V2005*Z*Z*Z*DC-REL

C O N T E N T S

MSC.Nastran 2005 Release Guide

3
Numeric Enhancements

4
Elements

7
Rotor Dynamics

8
DDAM Processor

9
Miscellaneous

10

Upward Compatibility

INDEX

Preface

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

Munich, Germany
Telephone: (49) (89) 43 19 87 0
Fax: (49) (89) 43 61 71 6

Rome, Italy
Telephone: (390) (6) 5 91 64 50
Fax: (390) (6) 5 91 25 05

Moscow, Russia
Telephone: (7) (095) 236 6177
Fax: (7) (095) 236 9762

Frimley, Camberley
Surrey, United Kingdom
Telephone: (44) (1276) 67 10 00
Fax: (44) (1276) 69 11 11

Tokyo, Japan
Telephone: (81) (03) 6911 1200
Fax: (81) (03) 6911 1201

Paris, France
Telephone: (33) (1) 69 36 69 36
Fax: (33) (1) 69 36 45 17

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.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.

CHAPTER

1

MSC.Nastran 2005 Release Guide

- Release Guide Introduction

1.1 Release Guide Introduction

MSC.Nastran 2005 introduces new capabilities and enhancements to existing capabilities while improving solution accuracy and performance. This guide covers all new functionality that has been added to the MSC.Nastran program since the MSC.Nastran 2004 r1 release in September 2003, and includes 2004 r2, 2004 r3, and 2005 r1. In addition, some of the new capabilities discussed in this guide are initial implementations, considered to be in beta form and offered for trial purposes.

Nonlinear

The horizons of MSC.Nastran continue to broaden as development pushes the program forward to take on more advanced and complex types of analysis. The release of MSC.Nastran 2004 brought a new dimension to the analysis capabilities of MSC.Nastran, with a new implicit nonlinear solution sequence (SOL 600) that embodies MSC.Marc, taking the product into the realm of complex highly nonlinear calculations, with contact and advanced materials. The trend continues this year with the preliminary release of the explicit nonlinear solution sequence (SOL 700), comprising MSC.Dytran and LS-Dyna, to the MSC.Nastran family of solvers.

This new capability, offered in beta form with MSC.Nastran 2005, allows complex explicit nonlinear calculations, including crash and impact analysis in this initial phase of implementation. Also, many known issues have been addressed concerning implicit nonlinear SOL 600, increasing overall robustness of the solution sequence.

Also available in MSC.Nastran 2005 is another new nonlinear solution sequence, SOL 400. Offered initially as a beta release, SOL 400 will embody the current MSC.Nastran nonlinear capabilities of solution sequences 106 and 129 into a single solution sequence.

Numeric Enhancements

A new domain decomposition method is now available in beta form for the Automated Component Mode Synthesis capability, which improves performance of NVH types of analysis, particularly for models with complex geometry. You can experience further performance improvements with a new technique used in the calculation of frequency response quantities for large problems solved across a wide frequency range.

Matrix to factor diagonal reporting has also been improved for analyses that use the Lagrange Multiplier Technique.

Elements

Analysis of composite structures now extend to include temperature dependency for unsymmetric laminates. Also, post processing composite structures is now much easier with global ply results tracking, particularly for areas where ply drop off is apparent.

For bar elements, it is now possible to specify a torsional mass moment of inertia value, and for both bar and beam elements a new option to specify lumped mass with no off diagonal terms has been added. This version of MSC.Nastran also introduces an arbitrary beam cross section beta capability, allowing the specification of cross section shapes using points, and further allowing these profiles to be optimized in SOL 200.

Improvements to the QUADR element are evident with improved convergence behavior, particularly for coarsely meshed models. You can now obtain grid point force output for DMIG and GENEL type definitions.

Dynamics

Improvements to the dynamic analysis capabilities of MSC.Nastran include:

- The support of multiple boundary conditions for frequency response analysis in SOL 200
- Better handling of c-set masses during the calculation of component modes
- Enhancements to dynamic excitation processing
- Enforced motion calculations

In addition, the MODESELECT Case Control command has been extended to include easier mode selection for all selection criteria, essentially replacing many of the existing mode selection bulk data parameters, as well as adding a new criteria based on modal effective mass.

Optimization

As well as the new nonlinear solution sequences, MSC.Nastran 2005 offers a beta topology optimization capability. This addition to the existing SOL 200 optimization solution sequence allows optimization analyses to be performed that require many design variables, a typical requirement of topology optimization.

Other optimization enhancements include:

- New functions for DRESP2

- A new response type for DRESP1
- The ability to transform an approximate optimization task to a feasible design
- Increased accuracy with dynamic response optimization analyses through the use of residual vectors

Rotor Dynamics

The rotor dynamics capability introduced in MSC.Nastran 2004 has been further enhanced to include the modeling of squeeze film dampers.

DDAM

We have introduced a new dynamic design analysis method (DDAM) solution sequence (SOL 187). Widely used in the ship building industry, DDAM is a form of shock spectrum analysis used to determine the dynamic response of a component to shock loading.

Further Enhancements

We have added many other enhancements to MSC.Nastran 2005. Some are mentioned in the following list.

- Furthering the integration of MSC.Nastran with MSC.ADAMS are new capabilities that aid model checkout and the determination of component attachment points
- The new variable endian capability allows easier transfer of OP2 results data between machines with differing binary file formats
- New ACCEL Bulk Data entries allow easier specification of acceleration loads across the model
- A new complex conjugate option for matrix multiplication (MPYAD) has been added, facilitating matrix manipulation
- SPC and SPCD entries are now stored in machine precision
- New PKS and PKNLS flutter options in aeroelastic analysis

CHAPTER

2

Nonlinear Analysis

2.1 MSC.Nastran Explicit Nonlinear -- SOL 700 - Beta Capability

Introduction

MSC.Nastran Explicit Nonlinear, also known as SOL 700, is a new capability introduced as a preliminary capability in MSC.Nastran 2005. This is the first phase of adding explicit nonlinear dynamics to MSC.Nastran.

The Phase 1 effort primarily is intended to solve highly nonlinear structural crash and impact problems. Fluids, air bags, seat belts, and passengers are not a part of Phase 1 and will be added in subsequent phases. Although Phase 1 primarily addresses crash and impact loading, usually described by initial velocity input, other types of dynamic loading are also supported.

SOL 700 works in a manner similar to SOL 600, but instead of spawning MSC.Marc, a special version of the well-known LS-Dyna program is spawned. Like SOL 600, SOL 700 contains an internal translator. MSC.Nastran input data is translated to MSC.Dytran input data during the IFP portion of MSC.Nastran. Later in IFP, LS-Dyna is spawned. Special subroutines have been added to LS-Dyna to accept MSC.Dytran format input data similar to LSTC's keyword input data in the standard version of LS-Dyna. Inside LS-Dyna, the MSC.Dytran input data is stored directly in memory and converted to a structured LS-Dyna style input file (which is the old style type of input without headers). LS-Dyna then continues with computations to produce output results for highly nonlinear problems.

SOL 700 is primarily intended for engineers and analysts who have constructed an MSC.Nastran finite element model for a purpose other than crash, but who wish to use the model for crash. This avoids having to read the MSC.Nastran model into a GUI, translate it to LS-Dyna or MSC.Dytran, and thus risk losing or not properly translating some MSC.Nastran input data.

Once you have completed the LS-Dyna portion of the execution, standard LS-Dyna results files such as d3plot, as well as standard MSC.Nastran files such as op2, xdb, punch and f06, are available for postprocessing. The capability will be delivered with the MSC.Nastran 2005 r2 release.

SOL 700 contains an internal translator that creates an MSC.Dytran file. If the original MSC.Nastran input file is named jid.dat (or jid.bdf), the MSC.Dytran file that is created is named jid.dytr.dat. The translator examines and converts Executive Control statements, Case Control commands and Bulk Data entries to MSC.Dytran.

The MSC.Nastran input can contain as many subcases as desired; however only one may be selected for use in any particular SOL 700 analysis. This is done using the Case Control commands, SKIP ON or OFF, to pick the desired subcase.

Linear and Nonlinear Analysis

SOL 700 is a dynamic analysis program that can perform linear transient analyses (such as SOL 109) as well as nonlinear transient analyses (such as SOL 129). It is also set up to perform linear or nonlinear static analyses (such as SOL 101 and SOL 106). Because of the numerical integration approach used within LS-Dyna, very small time steps are required to maintain accuracy and solution stability. The penalty for taking small time steps is partially offset by not having to decompose a stiffness matrix. See the upcoming *MSC.Nastran Explicit Nonlinear User's Guide* for further theoretical details. The small time step requirements are not a great problem when simulating events that occur quickly, such as impact or crash. However, for longer events such as low frequency dynamics or static analysis, the run time sometimes is too great for explicit methods and implicit analyses need to be employed.

SOL 700 has three ways of solving static problems:

- Dynamic relaxation -- The input is applied as a step function and large damping is added. The solution is run until approximate steady-state values are obtained.
- Slow buildup -- The static load is ramped slowly from zero to full value over a period of time long enough that no important natural frequencies are excited. No extra damping is added.
- Slow buildup with extra damping -- This method is like the previous method except that some extra damping is added; thus, the final run time can often be reduced.

Which method to use depends on the problem being solved and whether extra damping affects the solution or not. The second method produces the "exact" results but may take excessive computer time in some cases. The first method is the "classic" method of solving static problems using explicit analysis techniques.

New entries have been added to MSC.Nastran to make it easier to describe crash and impact. Examples are the new TICD entry, that adds a from-thru-by grid ID description so that initial velocity input can be described by one line rather than numerous Grid point lines. This allows an input file setup for something else to be edited and quickly changed to a crash analysis.

For those familiar with the LS-Dyna material descriptions, about 25 of the most important and commonly used LS-Dyna materials may be input to SOL 700 directly.

Contact is described using the same entries as SOL 600; however there is a new entry to easily describe a rigid wall used for car crash simulation. That entry is the MSC.Dytran WALL entry (see new entry section below).

In addition, for those familiar with MSC.Dytran, several important MSC.Dytran Parameters have been added.

Description of the SOL 700 Executive Control Statement

Format:

SOL 700,ID PATH= COPYR= OUTR= STOP= NP= NOERROR

Example:

SOL 700,129 PATH=3 OUTR=OP2 NP=4

(700,129 request nonlinear transient dynamics, path=3 requests use of the dytran-lsdyna script, outr=op2 requests that an op2 file, np=4 requests that 4 processors be used)

Summary:

SOL 700 is a new Executive Control statement like SOL. It normally activates an explicit nonlinear transient analysis integration scheme using dytran-lsdyna. It may also be used for implicit static analyses using LS-Dyna's Dynamic Relaxation or slow buildup options. The calculations will not be performed directly within MSC.Nastran. Instead, SOL 700 will use a separate solver based on LS-Dyna, which is spawned from MSC.Nastran. This client-server approach is similar to SOL 600, using MSC.Marc. For linear analyses such as normal modes, frequency response, or linear direct transient response, MSC.Nastran should be used.

For the first phase of this project, the SOL 700 statement will spawn dytran-lsdyna, which uses an MSC.Dytran text input interface to LS-Dyna. dytran-lsdyna is a 3D, explicit nonlinear analyses code with DMP (parallel processing domain decomposition) capabilities.

For Phase 2 of the project and beyond, fluid coupling, airbags, seat belts, and dummy passengers will be added. Then the user will be able to select the standard MSC.Dytran program or an enhanced dytran-lsdyna program.

Inputs and outputs will be the same as, or similar to, the familiar MSC.Nastran inputs and outputs or, at the user's request, LS-Dyna type outputs will be available. The LS-Dyna style outputs are the default for SOL 700.

For ID=129 or NLTRAN, SOL 700 will generate a dytran-lsdyna input data file “jid.dytran.dat”, where “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”=abcd).

Unless explicitly specified using the STOP= option, dytran-lsdyna will be executed from MSC.Nastran on any computer system capable of doing so (which includes most UNIX systems and Windows systems). For dytran-lsdyna to run, it must be installed, properly licensed, and accessible from the directory where the MSC.Nastran input data resides, MSC_BASE must be provided in the environment.

Executive Control Parameters:

The required ID may be one of several valid solution sequence integers or names shown in “**Solution Sequences**” on page 144 of the *MSC.Nastran Quick Reference Guide* for the SOL statement. Examples are 129 and NLTRAN.

The following solutions are available for Phase I of this project: 101, 106, 109, 129 (and their equivalent names).

All items on the SOL 700,ID after ID itself, may be specified in environmental variables. This may be done in any manner environmental variables can be set. They may be set by the MSC.Nastran user at run time or by the system administrator when MSC.Nastran is installed. Any values specified on the SOL statement override those in the environment. Environmental variables are fully described in the ***MSC.Nastran 2005 Installation and Operations Guide***. A keyword file is available to describe the format of each variable. The variable is normally set in the system-wide rc file, a user’s rc file, a local rc file or in a script used to submit MSC.Nastran. Any string or value listed on the SOL 700,ID statement is also valid as an environmental variable. If the environmental variables are placed in the system-wide rc file, they may be used by a company for all MSC.Nastran users and even hide the fact that dytran-lsdyna is being spawned if so desired.

The following describes the various options for PATH. We suggest the use of PATH=3 for Linux and UNIX and path=1 for Windows.

PATH=1 (May be used with Windows and Linux only)

If PATH=1 is specified, MSC.Nastran will determine the proper command to execute a serial dytran-lsdyna run. To aid MSC.Nastran in determining where dytran-lsdyna is located, the dynrun.pth file must be located in the same directory where the

MSC.Nastran input file resides. The dynrun.pth file must contain one line providing the location (complete path) of the dytran-ls-dyna run script. A typical example of the line in the file dynrun.pth follows.

Windows c:\sol700\

Linux: /msc/users/sol700

A string is appended to this path to form the complete command used to execute dytran-lsdyna:
"dytran-lsdyna jid=name.dytr.dat
O=name.dytr,d3hsp G=name.dytr.d3plot D=name.dytr.d3dump
F=name.dytr.d3thdt
A=name.dytr.runrsf B=name.dytr.d3drfl

For Windows, MSC.Nastran will spawn dytrna-lsdyna using the following command assuming the MSC.Nastran input data is named enf2e.dat. (Although the example appears as if it is on multiple lines, it is actually on a single line.)

c:\sol700/dytran-lsdyna i=enf2e.dytr.dat O=enf2e.dytr.d3hsp G=enf2e.dytr.d3plot D=enf2e.dytr.d3dump F=enf2e.dytr.d3thdt A=enf2e.dytr.runrsf B=enf2e.dytr.d3drfl

PATH=2 (May be used with Windows and Linux only)

If PATH=2 is specified, it is expected that the directory with the dytran-lsdyna run script is on the PATH. If PATH=2 is specified, dytran-lsdyna will be executed from inside MSCNastran using the commands for the PATH=1 option except that dynrun.pth is not required.

PATH=3 (Applicable to all computer systems)

If PATH=3 is specified, a script or batch file to execute dytran-lsdyna, located in the same directory as the dytran-lsdyna executable, will be executed. The name of the script or batch files is run_dytran (or run_dytran.bat). This directory and name of the script is determined by the first line in a file named sol700.pth. Options are specified on subsequent lines of the sol700.pth file.

Available PATH=3 options for Windows PC systems are as follows:

exe	The full path to the executable for dytran-lsdyna that is to be used.
	Optional -- If exe= is omitted, the directory where the script or batch file resides (first line of sol700.pth) will be used and dytran-lsdyna for UNIX/Linux and dytran-lsdyna.exe for windows will be appended. If exe= is used, it must be the second line in the sol700.pth file.
nproc	Number of processors.
	(Default is to used NP on the SOL 700 line. If NP and nproc are omitted, the default is 1). For parallel execution, the directory where the MSC.Nastran input file exists must be shared with read/write privileges. If wdir is used, it must also be shared (see below). The directory where the dytran-lsdyna executable resides must also be shared for parallel execution. In addition, all rules for MPICH must be followed properly, (see your system administrator to be sure all computers are properly configured for parallel execution using MPICH). The version of MPICH to use is 1.2.5 as of the initial SOL 700 release. It can be obtained from ftp.mcs.anl.gov if necessary.
bat	Run in background or forground (default).
debug	Output many messages from the script or batch file.
memory	Amount of memory. Example: memory=20m.
steps	Number of steps (1 or 2; default is 2).
	Two steps means that lsdyna is executed twice: once to form the “structured input file” and again to analyze it. Although steps=1 is faster, there are some models that fail using the steps=2 option.
wdir	Working directory. For parallel execution, this directory must be shared with read/write privileges. Default is directory where MSC.Nastran input resides.
copy	Yes or no. Input and output files are copied from wdir to the input directory. Default is yes.
delete	Yes or no. LS-Dyna scratch files are deleted or not. Default is yes.

machine	Machines and number of processors to use in the form: machine1#2+machine2#4 (use 2 processors on machine 1 and 4 processors on machine 2)
host	file name. Name of a hostfile containing the same information as "machine" The format of hostfile is as follows for the example for machine: machine1 2 machine2 4

A Windows example of the file sol700.pth for the PATH=3 case follows.

```
e:\sol700\dytran-lsdyna\run_dytran
exe=f:\latest_dytran-lsdyna\dytran-lsdyna.exe
nproc=4
memory=20m
steps=2
wdir=f:\temp
delete=yes
machine=pc01#2+pc02#2
```

For the above example, MSC.Nastran will create the following command to spawn dytran-lsdyna assuming your input file is named abcd.dat. (Although the example appears like it is on multiple lines, it is actually on a single line.)

```
e:\sol700\dytran-lsdyna\run_dytran exe=f: \latest_dytran-lsdyna\dytran-
lsdyna.exe jid=abcd.dytr nproc=4 memory=20m wdir=f:\temp delete=yes
machine=pc01#2+pc02#2
```

Available PATH=3 options for UNIX/Linux systems follows.

exe	The full path to the executable for dytran-lsdyna that is to be used. (Optional -- If exe= is omitted, the directory where the script or batch file resides (first line of sol700.pth) will be used and dytran-lsdyna for UNIX/Linux and dytran-lsdyna.exe for windows will be appended.) If exe= is used, it must be the second line in the sol700.pth file.
nproc	Number of processors. (Default is to use NP on the SOL 700 line. If NP and nproc are omitted, the default is 1.)
bat	Yes or no. Run in background or forground (default). Leave out for steps=2
debug	Yes or no. Outputs many messages from the script or batch file.

memory	Amount of memory; example: memory=20m (20 MB).
steps	Number of steps (1 or 2; default is 2). Two steps means that lsdyna is executed twice: once to form the “structured input file” and again to analyze it. Although steps=1 is faster, there are some models that fail using the steps=2 option.
wdir	Working directory. Default is directory where MSC.Nastran input resides.
copy	Yes or no. Input and output files are copied from wdir to the input directory. Default is yes.
delete	Yes or no. LS-Dyna scratch files are deleted or not. Default is yes.
cluster	Yes or no. If yes is specified, the job will be initiated on the machine that the user is logged on to, but the analysis is performed on the cluster nodes that are specified in machinefile. If the default of off is used, the job will run on the local machine and the machines listed in the machine file depending on the number of processors specified. This option is not available for early SOL 700 releases. Default is no.
mpipath	The MPI install directory if you wish to used a non-default MPI directory.
mpirun	The MPI run command you want to use. If entered, it overrides the default MPI run command on your machine as well as the command in mpipath.

A UNIX/Linux example of the file sol700.pth for the PATH=3 case follows.

```
/users/joe/sol700/run_dytran
nproc=4
memory=20m
steps=2
wdir=/tmp/dyna
```

For the above example, MSC.Nastran will create the a command similar to the following to spawn dytran-lsdyna assuming your input file is named abcd.dat

```
/users/joe/sol700/run_dytran \
exe=/users/joe/sol700/dytran-lsdyna \
jid=abcd.dytr nproc=4 memory=20m steps=2 wdir=/tmp/dyna
```

If PATH is not specified, a special version of dytran-lsdyna will normally be used. This version will be located in a subdirectory named dyna/machine below the MSC.Nastran base directory (MSC_BASE). The machine directory will be aix, alpha, hpux, etc. If MSC_BASE is not available for a particular computer system, PATH=1, 2 or 3 must be specified.

STOP

STOP is an optional item. STOP is used to prevent execution of dytran-lsdyna or prevent execution of MSC.Nastran after IFP 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.

STOP=1

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 dytran-lsdyna with ID=129).

STOP=3

If STOP=3, MSC.Nastran is stopped after IFP and dytran-lsdyna is not executed. This would be the normal STOP option if the user wants to examine a dytran-lsdyna input file, make some changes and then execute dytran-lsdyna manually.

The following dytran-lsdyna files are potentially affected by the OUTR option.

OUTR=OP2,XDB,F06,PCH (Not available in Version 2005 r1)

Choose one or more or omit -- translate dytran-lsdyna jid.dytr.d3plot output to MSC.Nastran. This option requires the use of the MSC.Nastran Toolkit. A license is not needed for the Toolkit as it is imbedded in standard SOL 700 licensing. The conversion between LS-Dyna's d3plot and op2.xdb.f06,punch is made using MSC.Patran's DRA/DAC together with a special version of the toolkit. The special toolkit executable is spawned from the original MSC.Nastran job after dytran-lsdyna completes and if any of the OUTR options are specified.

NP

NP=the number of processors (domains) for parallel processing. The default is one. In order to use more than one domain, MPI, Lam, POE, or whatever parallel program is needed must be properly installed on all computers involved and a hostfile designating which computers are to be used for each domain must have been setup prior to running the job. It is required that if NP>1, PATH=3 be used and a file named sol700.pth be located in the same directory as the MSC.Nastran input data. The sol700.pth file should contain all commands necessary to run dytran-lsdyna in parallel. This file must have execute permissions.

NOERROR

NOERROR is an optional item. If NOERROR is specified, errors due to features that are available in MSC.Nastran but not available in dytran-lsdyna, and/or features not yet supported by the translator will be ignored. If NOERROR is entered and STOP=2 (or 3) is not specified, dytran-lsdyna will be executed even though the complete MSC.Nastran model may not have been completely translated.

NOERROR only be used by experienced analysts and then only with extreme caution.

Table 2-1 Case Control Commands Available in SOL 700

Item	Case Contol Commands Available in SOL 700
$	Y
ACCELERATION	Y
BCONTACT	Y
BEGIN BULK	Y (Other BEGIN forms are not allowed)
DISPLACEMENT	Y
DLOAD	Y
ECHO	Y
ELFORCE see FORCE	Y
ENDTIME	Y (new)
FORCE & ELFORCE	Y (automatically produced in d3plot files no user control)
GROUNDCHECK	Y (MSC.Nastran f06 only)
IC	Y
INCLUDE	Y
LABEL	Y (MSC.Nastran f06 only)
LINE	Y (MSC.Nastran f06 only)
LOAD	Y (for dynamic pseudo-statics only)

Table 2-1 Case Control Commands Available in SOL 700

Item	Case Contol Commands Available in SOL 700
LOADSET	Y
MAXLINES	Y (MSC.Nastran f06 only)
MPC	Y
NLPARM	Y (Psuedo static analysis only)
NLSTRESS	Y (Changed to STRESS)
PAGE	Y (In MSC.Nastran only)
PARAM	Y (Only applicable parms are used)
PRESSURE	Y
SET	Y
SET – OUTPUT(PLOT)	N
SKIP	Y (Required if multiple subcases are present)
SPC	Y
STRAIN	Y
STRESS	Y
SUBCASE	Y (See note)
Note: Only one subcase can be selected for a particular SOL 700 analysis. Many subcases may be entered in the input file, but the one to be used must be selected using the SKIP ON and SKIP OFF Case Control commands. If the SKIP ON / OFF commands are not found or are in the wrong place, the first subcase encountered will be used and the others ignored.	
SUBTITLE	Y
TITLE	Y
TSTEP	Y (Same as TSTEPNL)
TSTEPNL	Y
VELOCITY	Y
WEIGHTCHECK	Y (In MSC.Nastran only)

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
AXIC	N	Y
AXIF	N	Y
AXSLOT	N	Y
BAROR	Y	
BCBPDY	Y	
BCHANGE	N	Y
BSURF	Y	
BCBOX	Y	
BCPROP	Y	
BCMATL	Y	
BCONP	N	
BCTABLE	Y (Revised)	
BLSEG	N	Y
CBAR	Y	
CBEAM	Y	
CBEND	N	Y
CBUSH	N	Y
CCONEAX	N	Y
CDAMP1D	Y (New)	
CDAMP2D	Y (New)	
CELAS1D	Y (New)	
CELAS2D	Y (New)	
CFLUID	N	Y
CGAP	N	Y
CHACAB	N	Y
CHEXA	Y (8 Nodes only)	
CONM2	Y	

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
CONROD	Y	
CORD1C	Y	
CORD1R	Y	
CORD1S	Y	
CORD2C	Y	
CORD2R	Y	
CORD2S	Y	
CORD3G	N	Y
CPENTA	Y (5 Nodes only)	
CQUAD4	Y	
CQUAD8	Y (4 Nodes only)	
CQUADR	Y	
CQUADX	N	Y
CREEP	N	Y
CROD	Y	
CSHEAR	N	Y
CTETRA	Y (4 Nodes only)	
CTRIA3	Y	
CTRIA6	Y (3 Nodes only)	
CTRIA3R	Y	
CTRIAX	N	Y
CTRIAX6	N	Y
CTUBE	Y	
CVISC	Y	
CWELD	N	Y
CSPOT	Y (New – LS-Dyna Weld)	
CFILLET	Y (New – LS-Dyna Weld)	

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
CBUTT	Y (New – LS-Dyna Weld)	
CCRSFIL	Y (New – LS-Dyna Weld)	
COMBWLD	Y (New – LS-Dyna Weld)	
DAMPGBL	Y (New for Dynamic Relaxiation)	
DAREA	Y	
DEFORM	N	Y
DELAY	N	Y
DMI	N	Y
DMIAX	N	Y
DMIG	N	Y
DPHASE	N	Y
DTI	N	Y
ECHOOFF	Y	
ECHOON	Y	
ENDDATA	Y	
EOSPOL	Y (New – Equation of state)	
FORCE	Y	
FORCE1	N	Y
FORCE2	Y	
FORCEAX	N	Y
GENEL	N	Y
GRAV	Y	
GRDSET	Y	
GRID	Y	
INCLUDE	Y	
IPSTRAIN	N	Y
ISTRESS	N	Y

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
LOAD	Y	
LSEQ	Y	
MAT1	Y	
MAT2	Y	
MAT3	Y	
MAT8	Y	
MATDxxx	Y (New LS-Dyna materials)	
MATD20M	Y (New Rigid Material Merge)	
MATEP	N	Y
MATHE	N	Y
MATHED	N	Y
MATF	N	Y
MATHP	Y	
MATS1	Y	
MATVE	N	Y
MATORT	N	Y
MATVORT	N	Y
MATVP	N	Y
MATG	N	Y
MFLUID	N	Y
MOMAX	N	Y
MPC	Y	
MPCAX	N	Y
NLPARM	Y (For pseudo statics)	
NLRGAP	N	Y
NOLINi	N	Y
NTHICK	N	Y

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
PANEL	N	Y
PBAR	Y	
PBARL	N	Y
PBCOMP	N	Y
PBEAM	Y	
PBEAML	N	Y
PBEND	N	Y
PBUSH	N	Y
PCOMP	Y	
PDAMP	Y	
PDAMP5	N	Y
PELAS	Y	
PELAST	N	Y
PGAP	N	Y
PHBDY	N	Y
PINTC	N	Y
PINTS	N	Y
PLOAD	Y	
PLOAD1	N	Y
PLOAD2	Y	
PLOAD4	Y (Continuation supported)	
PLOADX1	N	Y
PLPLANE	N	Y
PLSOLID	N	Y
PMASS	N	Y
PRESPT	N	Y
PROD	Y	

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
PSHEAR	N	
PSHELL	Y	
PSOLID	Y	
PTUBE	Y	
PVISC	Y	
RBAR	Y	
RBE1	N	Y
RBE2	Y	
RBE3	Y (Changed to RBE3D)	
RESTART	Y	Y
RFORCE	Y (CID, METHOD, continuation line not supported)	
RLOADi	N	Y
RROD	N	Y
RSPLINE	N	Y
RTRPLT	N	Y
SLOAD	N	Y
SPC	Y	
SPC1	Y	
SPCADD	Y	
SPCAX	N	Y
SPCD	Y	
SUPAX	N	Y
TABLED1	Y	
TABLED2	Y	
TABLED3	Y	
TABLES1	Y	

Table 2-2 Bulk Data Entries Available in SOL 700

Item	Bulk Data Entries Available in SOL 700	Fatal Error
TEMP	N	Y
TEMPD	N	Y
TIC	Y	
TICD	Y (New with increment options)	
TIC3	Y (New MSC.Dytran type entry)	
TLOAD1	Y	
TLOAD2	Y	
TSTEP	Y (Changed to TSTEPNL)	
TSTEPNL	Y	
WALL	Y (New rigid wall entry)	

Summary of New or Changed Bulk Data Entries for SOL 700

BCTABLE	Contact table – many new fields have been added for “slaves”
CDAMP1D	Scalar damper connection
CDAMP2D	Scalar damper connection
CELAS1D	Scalar spring connection
CELAS2D	Scalar spring connection
CSPOT	Spot weld in the LS-Dyna style (replaces CWELD for SOL 700)
CFILLET	Fillet weld in the LS-Dyna style (replaces CWELD for SOL 700)
CBUTT	Butt weld in the LS-Dyna style (replaces CWELD for SOL 700)
CCRSFIL	Cross-fillet weld in the LS-Dyna style (replaces CWELD for SOL 700)
COMBWLD	Complex combined weld in the LS-Dyna style (replaces CWELD for SOL 700)
DAMPGBL	Defines values to use for dynamic relaxation
EOSPOL	Defines equation of state to use for solids in combination with certain materials.
MATD001	LS-Dyna material 1 – isotropic elastic
MATD2OR	LS-Dyna material 2 – orthotropic

MATD2AN	LS-Dyna material 2 – Anisotropic
MATD003	LS-Dyna material 3 – isotropic with kinematic hardening
MATD005	LS-Dyna material 5 – soil and foam
MATD006	LS-Dyna material 6 – viscoelastic
MATD007	LS-Dyna material 7 – nearly incompressible rubber
MATD012	LS-Dyna material 12 – low cost isotropic plasticity model for solids
MATD013	LS-Dyna material 13 – non-iterative plasticity model with failure
MATD014	LS-Dyna material 14 – soil and foam with failure
MATD015	LS-Dyna material 15 – Johnson-Cook strain and temperature sensitive plasticity
MATD018	LS-Dyna material 18 – isotropic plasticity with rate effects
MATD019	LS-Dyna material 19 – strain-rate dependent material model
MATD020	LS-Dyna material 20 – rigid material
MATD20M	merges several rigid materials defined using MATD020
MATD022	LS-Dyna material 22 – orthetropic material with brittle failure (composites)
MATD024	LS-Dyna material 24 – elasto-plastic material with arbitrary stress-strain curves and strain-rate dependency
MATD026	LS-Dyna material 26 – Anisotropic honeycomb and foam
MATD027	LS-Dyna material 27 – Two-variable rubber model
MATD028	LS-Dyna material 28 – Elasto-plastic resultant formulation
MATD030	LS-Dyna material 30 – Shape-memory superelastic material
MATD031	LS-Dyna material 31 – Frazer-Nash rubber
MATD054	LS-Dyna material 54 – Enhanced composite material model
MATD057	LS-Dyna material 57 – Highly compressible low density foams
MATD059	LS-Dyna material 59 – Shell or solid composite models
MATD062	LS-Dyna material 62 – Confor viscous foam model
MATD063	LS-Dyna material 63 – Crushable foam with damping
MATD064	LS-Dyna material 64 – Strain-rate dependent plasticity with power law hardening

MATD077	LS-Dyna material 77 – General Christensen rubber model
MATD080	LS-Dyna material 80 – Ramberg-Osgood plasticity
MATD081	LS-Dyna material 81 – Elasto-visco-plastic with arbitrary stress-strain curve
MATD100	LS-Dyna material 100 – Material for spot welds
MATD127	LS-Dyna material 127 – Arruda-Boyce rubber
MATD181	LS-Dyna material 181 – Simplified rubber and foam model
RBE3D	MSC.Dytran-style RBE3
TICD	Initial conditions like TIC with from-thru-by incrementing
WALL	Rigid wall

Summary of New Bulk Data Parameters for SOL 700

DYENDTIM	Determines how to translate TSTEPNL to MSC.Dytran
DYMATS1	Determines how to translate MATS1 to MSC.Dytran
DYLDKND	Designates type of stress-strain curve
DYCOWPRD	ID of Cowper Symonds strain rate equation
DYCOWPRP	P in Cowper Symonds strain rate equation
DYNAMES	Control of output file names - d3plot or jid.dytr.d3plot
DYSTATIC	Method to simulate static analysis (see above three methods)
DYBLDTIM	Number of seconds a static load is built up
DYBULKL	Value of the linear coefficient in the bulk viscosity equation
DYINISTEP	Initial time step
DYCONSLSFAC	Default scale factor for contact forces
DYCONRWPNAL	Scale Factor for rigid wall penalty value
DYCONPENOPT	Penalty stiffens option
DYCONTHKCHG	Whether or not shell thickness change is considered in contact
DYCONENMASS	Treatment of mass of eroded grids
DYCONECDT	Time step size for eroding contact
DYCONIGNORE	Flag to ignore initial penetration or not

DYCONSKIPRWG	Controls whether or not to generate a few extra nodes to visualize a rigid wall
DYHRGIHQ	Default hourglass viscosity type
DYHRGQH	Default hourglass viscosity coefficient
DYENERGYHGEN	Hourglass energy calculation option
DYTERMNENDMAS	Percent change in mass to end calculation
DYTSTEPERODE	Flag which determines whether elements will be eliminated at time TSMIN
DYTSTEPDT2MS	Time step size for mass scaling
DYMAXSTEP	Maximum allowable time step
DYMINSTEP	Minimum time step that terminates the analysis
DYSHELLFORM	Default shell formulation
DYSHTHICK	Specifies whether or not shell thickness changes with membrane straining
DYSTEPFCTL	Scale factor for internally calculated time step
DYRBE3	Control of RBE3 translation to MSC.Dytran RBE3 or RBE3D
DYSHNIP	Number of integration points for SOL 700 shells
DYNEIPH	Control of integration point data output for solids
DYNEIPS	Control of integration point data output for shells
DYMAXINT	Another control of integration point data output for shells
DYSTRFLG	Control of strain tensor output
DYSIGFLG	Flag to include stress tensor in binary output file
DYEPSFLG	Flag to include effective plastic strain in binary output file
DYRLTFLG	Flag to include stress resultants in binary output file
DYENGFLG	Flag to include internal energy and thickness in binary output file
DYCMPFLG	Flag determining coordinate of output stress and strain tensors
DYIEVERP	Flag determining if more than one output state can be written to d3plot files
DYBEAMIP	Number of beam integration points for output

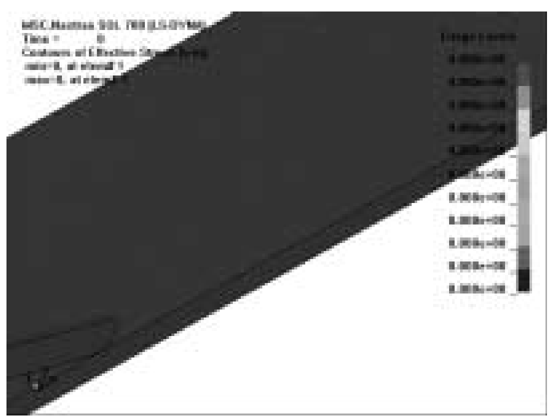
DYDCOMP	Flag controlling elimination of rigid body output
DYSHGE	Flag controlling output hourglass energy density
DYSTSSZ	Flag controlling output of shell element time step and mass
DYN3THDT	Flag controlling output of material energy
DYNINTSL	Number of solid element integration points for output

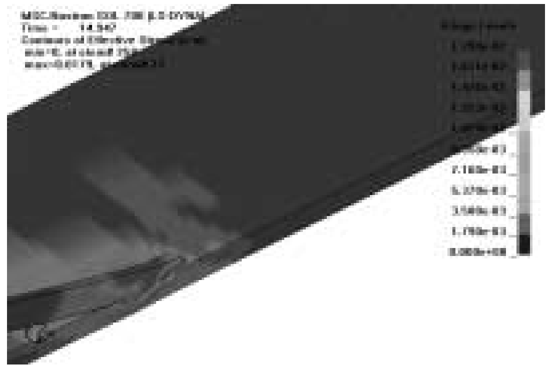
Example: Projectile Hitting a Plate with Failure

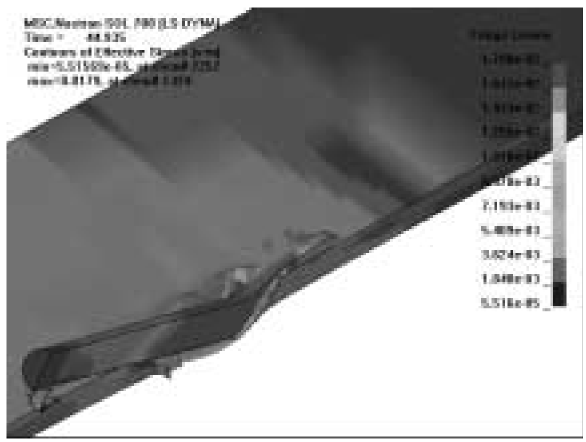
One typical example of SOL 700 Phase 1 is a projectile hitting a plate at an oblique angle. The initial velocity of the projectile is large enough that over time various elements in the plate fail. Depending on the postprocessor used, if it can account for failed elements, the failed elements are removed from the model.

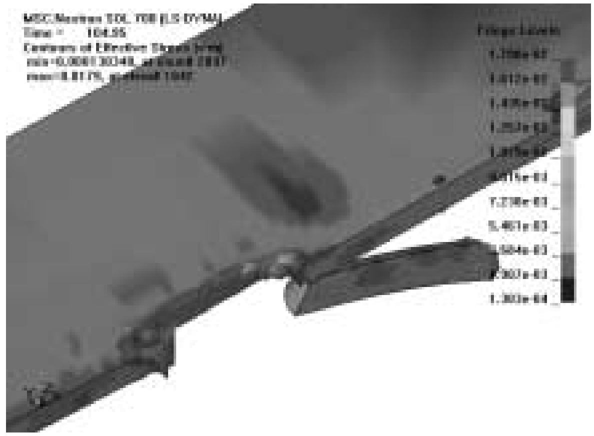
This model involves contact between the projectile and the plate. SOL 600-style contact is used. It also involves the use of LS-Dyna material MATD024 (elasto-plastic material with arbitrary stress- strain curves and strain-rate dependency). This problem takes about 20 minutes to run on a 2.4 GHz PC.

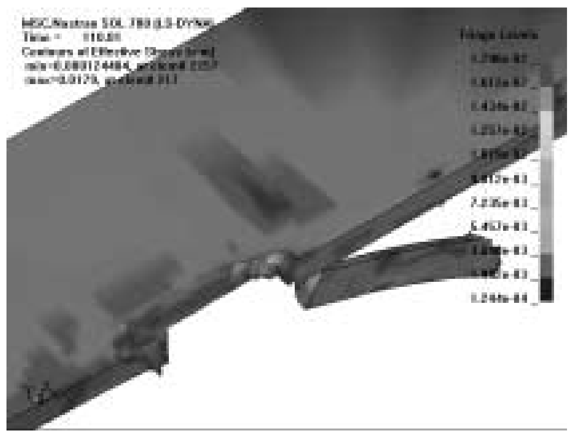
The following plots show various time slices for the analysis:











Portions the MSC.Nastran input file named projtl.dat are shown and discussed below:

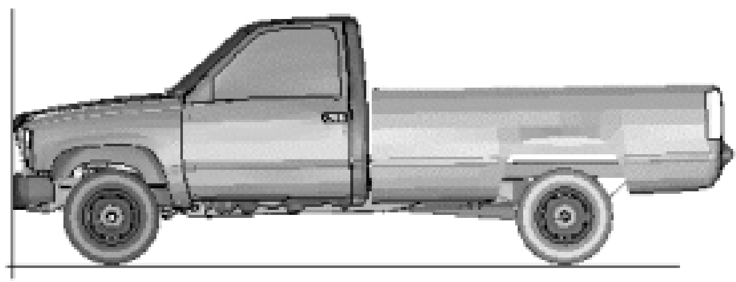
```
SOL 700,NLTRAN path=1 stop=1
TIME 10000
CEND
  ECHO = NONE
  DISPLACEMENT(SORT1,print,PLOT) = ALL
  Stress(SORT1,PLOT) = ALL
  Strain(SORT1,PLOT) = ALL
  accel(print,plot)= ALL
  velocity(print,plot)= ALL
  echo=both
  SPC = 2
  IC=1
  TSTEPNL = 20
  BCONTACT = 1
  weightcheck=yes
  page
BEGIN BULK
TSTEPNL       20      10      11       1               5       10        +
+                                                                        +
```

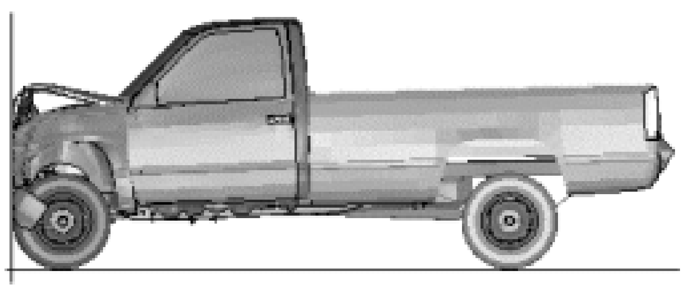
```
+                      0
PARAM,DYDTOUT,5
PARAM*,DYCONSLSFAC,1.0
PARAM,OGEOM,NO
PARAM,AUTOSPC,YES
PARAM,GRDPNT,0
param,dyendtim,1
param,dymats1,1
param,dyldknd,0
$
BCTABLE  1                      4
         SLAVE   3
+
                                       YES
         MASTER  4
$
BCBODY   3       3       DEFORM  3       0
BCBODY   4       3       DEFORM  4       0
$
BCPROP   3       2
BCPROP   4       1
$
$
$ ========== PROPERTY SETS ==========
$
$            * projectile *
$
PSOLID         1       1
$
$            * plate *
$
PSOLID         2       2
$
$
$ ========= MATERIAL DEFINITIONS ==========
$
$
$
$ -------- Material MAT_PLASTIC_KINE.2 id =2
MATD024        1   18.62    1.17     .22  0.0179             0.8
$ -------- Material MAT_PLASTIC_KINE.1 id =1
MATD024        2   7.896     2.1    .284    0.01             0.8
$
$
$
$ ======== Load Cases =======================
$
$
$ ------- Initial Velocity BC ini -----
$
TICD           1       1       1          0.1246    2586       1
TICD           1       1       3        -0.03339    2586       1
.
.
.
ENDDATA
```

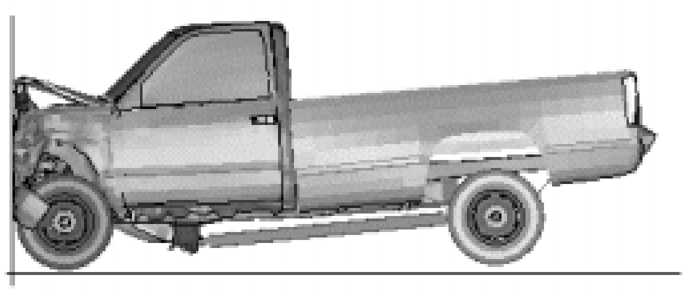
All of the previous input data are described in the compatible *MSC.Nastran Quick Reference Guide* and summarized above. Note that it was only necessary to add BCONTACT=1 to the Case Control, a few new Bulk Data parameters and a few contact entries to the Bulk Data to an existing file that would be used in MSC.Nastran SOL 101, 106, 109 or 129 analyses.

Example: Pickup Truck Crash Test

Another example involves crash testing of a pickup truck against a rigid wall. The input data file for this example is quite large and can be provided on request. It is a typical example of what can be done using a full car or truck model, developed originally for NVH analysis, in SOL 700 for crash simulation.







Where Can I Find More Information:

MSC.Nastran Explicit Nonlinear Analysis, SOL 700, is documented in the following manuals and guides:

- *MSC.Nastran Quick Reference Guide*
- *MSC.Nastran Explicit Nonlinear User's Guide*
- *MSC.Patran User's Guide*
- *MSC.Patran – MSC.Nastran and MSC.Dytran Preference Guides*
- *LS-Dyna Keyword User's Manual, Version 970* (available from LSTC)
- *LS-Dyna Theoretical Manual* (available from LSTC)
- *MSC.Dytran Reference Manual*
- *MSC.Dytran Theory Manual*

2.2 MSC.Nastran Implicit Nonlinear - SOL 600

Between the release of MSC.Nastran 2004 and MSC.Nastran 2005, there have been many improvements, a few new capabilities have been added and several errors have been corrected. This section summarizes the most important of these items.

Additions to the SOL 600 Executive Control Statement:

The new executive control statement is as follows:

SOL 600, ID PATH= COPYR= NOERROR OUTR=op2,xdb,pch,f06,eig,dmap,beam NOEXIT STOP= CONTINUE=

New items are dmap, beam and CONTINUE. An explanation of these items follows:

dmap	The user will enter his own DMAP to create whatever type of output that is desired, such as op2, xdb, punch, f06. For all other options, DMAP is generated as needed internally by MSC.Nastran.
beam	The beam option must be specified if op2,xdb,pch. or f06 options are specified and beam internal loads are to be placed in any of these files. The beam and eig options are mutually exclusive (you cannot specify both).

CONTINUE= is an option that specifies how MSC.Nastran will continue its analysis after MSC.Marc finishes. To continue the analysis, do not enter any STOP or OUTR options. It is possible to perform more than one of these operations if necessary.

0	MSC.Nastran will continue the current solution sequence as normal. For example, if SOL 600,106 is entered, SOL 106 will continue as normal after MSC.Marc finishes. Only 3D contact or materials supported by SOL 106 may be used.
1	MSC.Nastran will switch to SOL 107 to compute complex eigenvalues. MSC.Marc will generate DMIG matrices for friction stiffness (and possibly damping) on a file specified by pram,marcfil1,name and time specified by param,marcstif,time. This is accomplished by making a complete copy of the original MSC.Nastran input file and spawning off a new job with the SOL statement changed and an INCUDE statement for the DMIG file.

2	MSC.Nastran will switch to SOL 107 to compute complex eigenvalues. MSC.Marc will generate OUTPUT4 matrices for friction stiffness (and possibly damping) on a file specified by pram,marcfil2,name and time specified by param,marcstif,time, This is accomplished by making a complete copy of the original MSC.Nastran input file and spawning off a new job with the SOL statement changed and an INCLUDE statement for the DMIG file.
3	MSC.Nastran will switch to SOL 111 to compute modal frequency response. MSC.Marc will generate natural frequencies and mode shapes that are read into MSC.Nastran from a file specified by param,marcfil3,name
4	Same as option 3, except that SOL 112 for linear transient response will be used.
5	MSC.Nastran will switch to the solution sequence given in field 9 of the MDMIOUT entry. In addition, the DMIG entries specified by MDMIOUT will be included in a separate MSC.Nastran execution spawned from the original execution. Case Control and Bulk Data will be added to the original input to properly handle these matrices in the spawned MSC.Nastran execution.

An example of input using the continue=1 option is as follows:

```
SOL 600,106 path=1 stop=1 continue=1
TIME 10000
CEND
param,marcbug,0
  ECHO = sort
  DISP(print,plot) = ALL
  STRESS(CORNER,plot) = ALL
  STRAIN(plot) = ALL
  SPC = 1
  LOAD = 1
  NLPARM = 1
  CMETHOD=101
BEGIN BULK
param,marcfil1,dmig002
param,mrmtxnam,kaax
param,mrspawn2,tran
param,mrrcfile,nast2.rc
PARAM,OGEOM,NO
PARAM,AUTOSPC,YES
PARAM,GRDPNT,0
EIGC, 101, HESS, , , , ,50
NLPARM         1      10            AUTO       1              P      YES
PLOAD4         1     121   -800.
PLOAD4         1     122   -800.

(rest of deck is the same as any other SOL 600 input file)

CQUAD4       239       2     271     272     293     292
CQUAD4       240       2     272     273     294     293
ENDDATA
```

The full input for this example can be obtained from MSC.Nastran development. The name of the input file continu2.dat

Support of Complex Eigenvalue Analysis

SOL 600 now supports complex eigenvalue analysis via the CMETHOD Case Control command and the EIGC Bulk Data entry. In addition, four new Bulk Data parameters have been introduced:

param,marcfil1,dmig002	This means that a file named dmig002 will be used. It contains stiffness matrix terms (possibly from a set of unsymmetric friction stiffness matrices)
param,mrmtxnam,kaax	This means that in the dmig002 file, use DMIG matrix terms labeled kaax (or KAAX – case does not matter).

param,mrspawn2,tran	This means that the primary MSC.Nastran run will spawn another MSC.Nastran run to compute the complex eigenvalues. The name of the command is nastran (nas is always used and the characters specified by this parameter are added to the end of nas. Thus, we get nas+tran=nastran).
param,mrrcfile,nast2.rc	This is the name of the rc file to be used for the second (spawned) MSC.Nastran run.

The flow of the run is as follows:

- Create a primary MSC.Nastran SOL 600 input file (we will name it jid.dat for this example)
- Submit MSC.Nastran in the standard fashion. For this example, the following command is used:

  nastran jid rc=nast1.rc

  The nast1.rc file contains items such as scratch=yes, memory=16mw, etc.
- The primary MSC.Nastran run creates an MSC.Marc input file named jid.marc.dat
- The primary MSC.Nastran run spawns MSC.Marc to perform nonlinear analysis. MSC.Marc generates the required DMIG matrices for this example.
- The nonlinear MSC.Marc analyses completes and generates standard files.
- Control of the process returns to MSC.Nastran. A new MSC.Nastran input file named jid.nast.dat will be created from the original input file. This file will contain the CMETHOD Case Control command and EIGC Bulk Data entry, all of the original geometry and additional entries to read the dmig002 file.
- A second MSC.Nastran job will be spawned from the primary MC.Nastran run using the command

  nastran jid.nast rc=nast2.rc

  The nast2.rc file can be the same as nast1.rc or can contain different items. Usually memory will need to be larger in nast2.rc than in nast1.rc.
- The second MSC.Nastran run computes the complex eigenvalues and finishes.
- Control of the process returns to the primary MSC.Nastran run and it finishes.

The first portion of the dmig002 file is as follows:

```
$2345678 2345678 2345678 2345678 2345678 2345678 2345678 2345678 234567812345
DMIG    KAAX           0       1       2       0                     324
DMIG*   KAAX                           6               1
*                      6               1 3.014712042D+05
*                      6               2 4.204709763D+08
*
DMIG*   KAAX                           6               2
*                      6               1 1.204709763D+05
*                      6               2 3.014712042D+05
*
DMIG*   KAAX                           6               3
*                      6               1-4.616527206D+04
*                      6               2-4.616527206D+04
*                      6               3 1.308497299D+05
DMIG*   KAAX                          17               1
*                      6               1 6.239021038D+04
*                      6               2-2.528344607D+03
*                      6               3-6.239758760D+03
*                     17               1 5.939989945D+05
*
```

Temperature-Dependent Stress Strain Curves

MSC.Nastran 2005 offers the capability of stress-strain curve dependence as a function of temperature. The user specifies these stress strain curves at different temperatures and then specifies the temperature to use for each subcase. Linear interpolation between the supplied curves is used to determine the appropriate curve at the temperature specified for a particular subcase. MSC.Marc's AF-Flowmat capability is used for this capability; therefore, user subroutines do not have to be supplied. This capability is best explained with an example (this example can be obtained from MSC.Nastran development. The name of the file is mattep20.dat).

```
SOL 600,NLSTATIC path=1 stop=1
TIME 10000
CEND
  ECHO = NONE
  DISPLACEMENT(plot) = ALL
  SPCFORCE(PLOT) = ALL
  Stress(PLOT) = ALL
  Strain(PLOT) = ALL
  SPC = 1
  NLPARM = 2
  temp(init)=10
subcase 1
  temp(load)=11
  LOAD = 100
subcase 2
  temp(load)=12
  LOAD = 200
subcase 3
  temp(load)=13
  LOAD = 300
```

```
BEGIN BULK
param,mrafflow,mymat0
param,mrtabls1,4
param,mrtabls2,1
NLPARM         2      10             AUTO       1      20      P
PARAM,LGDISP,1
tempd, 10, 70.
tempd, 11, 110.
tempd, 12, 700.
tempd, 13, 1100.
$LOAD, 20, 1.0, 2.0, 1, 1.0, 2
load, 100, 1., 1., 1
load, 200, 1., -.5, 1
load, 300, 1., 1.1, 1
PLOAD4         1       1    -15.
.
.
.
$ Constraint Set 1 : Untitled
SPC            1       1  123456      0.
SPC            1       8  123456      0.
SPC            1      15  123456      0.
SPC            1      22  123456      0.
SPC            1      29  123456      0.
$  Property 1 : Untitled
PSHELL         1       1   0.125       1               1              0.
$  Material 1 : AISI 4340 Steel
MATEP, 1,TABLE, 35000., 2,CAUCHY,ISOTROP,ADDMEAN
MAT1           1  2.9E+7             0.327.331E-4  6.6E-6     70.         +MT    1
+MT    1 215000. 240000. 156000.
MAT4           14.861E-4   38.647.331E-4
$      1       2       3       4       5       6       7       8       9
$2345678 2345678 2345678 2345678 2345678 2345678 2345678 2345678 2345678
MATTEP         1                      21
MATT1          1                                       7
TABLEM1        7
+         70.0    6.6E-6   1000.  6.5E-6   1200.  6.4E-6  1500.   6.3E-6
+         2000.   6.2E-6    ENDT
$2345678 2345678 2345678 2345678 2345678 2345678 2345678 2345678 2345678
TABLEST       21
+         70.0        31   1000.      32   1200.      33   1500.      34
+         2000.       35    ENDT
TABLES1, 31
  , 0., 15000., 1.0, 16000., 10., 25000., 100., 30000.,
  ,  99999., 40000., ENDT
TABLES1, 32
  , 0., 13000., 1.0, 14000., 10., 23000., 100., 28000.,
  ,  99999., 28000., ENDT
TABLES1, 33
  , 0., 11000., 1.0, 12000., 10., 21000., 100., 26000.,
  ,  99999., 25000., ENDT
TABLES1, 34
  , 0.,  9000., 1.0, 10000., 10., 19000., 100., 22000.,
  ,  99999., 24000., ENDT
TABLES1, 35
  , 0.,  5000., 1.0,  7000., 10.,  9000., 100., 13000.,
  ,  99999., 15000., ENDT
GRID           1       0      0.      0.      0.       0
.
```

```
.
.
CQUAD4
.
.
.
ENDDATA
```

In this input, the stress strain curves are specified by TABLES1 entries. The collection of stress-strain curves to be used is specified in the TABLEST entry and the corresponding temperatures at which they apply is specified in the TABLEM1 entry. The TABLEM1 ID is called out in field 7 of the MATT1 entry and the TABLEST ID is called out in field 5 of the MATTEP entry. TABLEST must list the stress strain TABLES1 IDs in order of increasing temperature and the first ID must be at the lowest temperature specified anywhere in the analysis. In this example, it is a temperature of 70 corresponding to temp(init)=10 in the Case Control. Similarly, the temperatures in the TABLEM1 entry must be in increasing order. The stress-strain curves should cover the entire range of temperatures for the analysis so that no extrapolation is needed. The actual temperatures for each subcase are given by the temp(load) specifications for each subcase.

There is one parameter that is critical to this analysis:

param,mrafflow,mymat0	Name of the file containing temperature dependent stress versus plastic strain curves in MSC.Marc's AF_flowmat format. This file can be generated from the current MSC.Nastran run using TABLEST and TABLES1 entries or a pre-existing file can be used depending on the value of PARAM,MRAFFLOR. The extension ".mat" will be added to Name. If this is a new file, it will be saved in the directory from which the MSC.Nastran execution is submitted. If a pre-existing file is to be used, it can either be located in the directory where the MSC.Nastran execution is submitted or in the MSC.Marc AF_flowmat directory.

Improved Parallel Processing for SOL 600

In previous versions of SOL 600, the basic MSC.Marc input file had to be split up into as many MSC.Marc input files as processors to be used. MSC.Nastran 2005 incorporates the capability to use a new MSC.Marc feature called the Single File Parallel file. For this to work properly, the user must obtain MSC.Marc 2003 beta 2 or later to run in combination with MSC.Nastran. The interface to use this capability specifies KIND=0 or blank on the PARAMARC entry as shown below (the other options are still available but should be considered obsolete).

Format:

1	2	3	4	5	6	7	8	9	10
PARAMARC	ID	KIND	NPROC						

Example: To create 4 parallel processes using MSC.Marc's single file input

PARAMARC	51		4						

Field	Contents
ID	Identification number of the PARAMARC entry -- Not presently used. (Integer)
KIND	Designates how parallel domains are created. (Integer $\geq$ 0, Default = 0) 0=Parallel processing is accomplished using MSC.Marc's single file input. MSC.Marc Version 2005 and subsequent versions must be used. The command line to execute MSC.Marc is changed from -np N (or -nprocd N) to -nps N where N is the number of processors. The maximum number of processors for MSC.Marc is 256. Continuation lines may not be entered for KIND=0.

A similar option to create a single-file MSC.Marc t16 file is available starting with MSC.Marc 2005. This option is picked using Bulk Data PARAM,MARCOUTR,1. Be sure to have MSC.Marc 2005 or later before using this option since it does not work properly with earlier versions.

Support for Intel and Digital Visual Fortran PC Version of MSC.Marc

Starting with MSC.Marc 2005, an Intel version as well as a Digital Visual Fortran (also known as Comopaqa Visual Fortran) is available. MSC.Nastran 2004 r3 and MSC.Nastran 2005 support both versions. Distinctions between the two versions are only necessary for specifying the PATH to MSC.Marc and for OUTR results processing. The PATH can be handled as normal by using one of the path options on the SOL 600 Executive statement. The OUTR options will be processed internally by MSC.Nastran if the MSC.Marc Intel version is used and will require an external t16op2.exe program if the Digital Visual Fortran version is used. Beam / Bar loads, stresses and strains are available using the Intel version and are not available using the Digital Visual Fortran version. The PC version for OUTR processing can be selected by the new MSC.Nastran Bulk Data parameter:

MARCWIND Integer, Default=0 Determines which Windows version of MSC.Marc (Digital Visual Fortran or Intel) is to be used for those versions of MSC.Marc that support both versions. This option is only necessary if any OUTR option is used. If the Intel version is used, all t16op2 work is accomplished inside MSC.Nastran. If the DF version is used, a separate t16op2.exe program is required and must be on the PATH. This parameter applies only to SOL 600 used in combination with MSC.Marc 2005 or later versions. The MSC.Marc PC version prior to 2005 always used the Digital Visual Fortran version.

0 The Digital Visual Fortran version of MSC.Marc is used.

1 The Intel version of MSC.Marc is used.

Other SOL 600 Items:

- PBUSHT support has been added for nonlinear springs. TABLED1 can be used to specify the load-deflection curve. The PBUSHT/TBLED1 data is mapped to MSC.Marc's SPRINGS option with table-driven force-deflection curves. This capability will work with MSC.Marc 2003 or 2005.
- Buckling: In prior SOL 600 versions, only one nonlinear load case could be run prior to a buckling eigenvalue extraction. Now, multiple nonlinear load cases can be run before requesting buckling eigenvalue extraction.
- Improved beam offsets: MSC.Nastran 2005 supports beam and shell/plate offsets - they were not supported in some earlier versions and only partially supported in recent versions. The offsets are handled by adding extra nodes at the offset coordinates and connecting RBE2 between original and offset coordinates. Beam type 98 must be used rather than beam type 14 for offsets since beam type 14 does not have the fully 6 DOFs required for RBE2. That means beam plasticity cannot be combined with the beam offset. The numbering of the extra modes starts at the highest node ID, and increments by one above that.
- CBEAM and CBAR internal loads, stresses and strains were added for op2, xdb, punch and f06 results. This addition is applicable to UNIX/Linux systems and to Windows systems if an Intel Marc 2005 (or beyond) version is used.

- Bolt elements were added to support MSC.Marc both outside the USA and within the USA. MSC.Nastran Bulk Data entries, MBOLT and MBOLTUS reflect these new additions.
- The stacking direction for 3D composites were added with a new entry, MSTACK.
- Licensing for SOL 600 was changed to eliminate an extra license to spawn MSC.Marc. The MSC.Marc license is adequate for that purpose. In addition, a standard MSC.Nastran Nonlinear (SOL 160, 129) license was sometimes required -- this has now been eliminated.
- Improved support for element material coordinate systems
- Improved non-constant pressure loading, which means that P1, P2, P3, P4 on the PLOAD4 entry will be used. Before, P1 was applied as a constant pressure n the element face.
- The following errors have been corrected:
    - Work hardening slope specified by MATEP was ignored.
    - Gasket materials could not be defined with other solid element materials.
    - Eigenvalue analysis after a nonlinear analysis sometimes gave wrong results.
    - FTYPE variable of the BCPARA entry was not translated.
    - More than one NLSTRAT entry could not be provided.
    - RFORCE did not translate properly.
    - MSC.Marc’s ORIENTATION option was not employed correctly.
    - CPENTA pressures were sometimes applied to the wrong face.
    - BEGINBULK caused an abort.

2.3 Pre-release of the Nonlinear Transient Analysis in SOL 400

Introduction

In order to improve the nonlinear solution procedure in MSC.Nastran, a general nonlinear solution sequence, SOL 400, has been introduced since MSC.Nastran 2004 to support nonlinear static analysis. In MSC.Nastran 2005, a new capability, the nonlinear transient analysis, has been added to SOL 400. Since some capabilities are still under development and the V&V test has not been completed, this is only a pre-release and its intention is to get user feedback. Eventually, this solution sequence will include all nonlinear analyses, such as the nonlinear static analysis, the nonlinear transient analysis, the nonlinear buckling analysis, the nonlinear normal modes analysis, and other related solution procedure such as the linear static and transient analysis into one general solution procedure. In the final run, SOL 400 is going to replace the current nonlinear static solution sequence SOL 106 and the nonlinear transient solution sequence SOL 129. In addition, the nonlinear heat transfer solution sequence, SOL 153 and SOL 159 will also be included in SOL 400 in the future. Up until now, SOL 400 does not replace SOL 106 and 129.

Benefits

The new benefits of SOL 400 are discussed this section. Some of the benefits may be subject to the limitation discussed in next section. The benefits are:

- The Case Control command STEP, which was first introduced in MSC.Nastran 2004 to allow user to input a flexible loading and solution sequence at an independent loading SUBCASE in the nonlinear static run, is also supported by nonlinear transient analysis.
- User can specify different type of ANALYSIS, such as NLSTAT and NLTRAN, at different SUBCASE's in a single run.
- The different STEP's can specify different types of ANALYSIS only when they belong to the same kind of analyses, such as static or transient analysis. For example, both NLSTAT and LNSTAT are static analyses; therefore, they can be mixed in a SUBCASE. However, NLSTAT and NLTRAN cannot be mixed in a SUBCASE.
- An improved nonlinear iteration procedure to make solution easier and faster to converge. All AUTO, TSTEP (or ITER), ADAPT and SEMI, which is the method for controlling stiffness updates, are supported in the nonlinear transient analysis.

- AUTOSPC can be executed when every time updating the matrices if user requests.
- Direct matrix input, such as K2PP, and EPOINT are supported in nonlinear transient analysis.
- Support “grid based reordering” for a faster decomposition to both nonlinear static and transient analyses.
- Support thermal excitation in the nonlinear transient analysis. Two new Bulk Data entries, TTEMP and TMPSET, have been added for this new feature in nonlinear transient analysis.
- The GPFORCE and ESE output are supported in the nonlinear static analysis.
- The same restart capability, which had been introduced in MSC.Nastran 2004, is also available to the nonlinear transient analysis now.
- A more flexible output method for the nonlinear transient analysis. User can use a new parameter NLPACK to control the output and restart time steps.
- Allow simulation of the same output logic and format of SOL 129 by specifying a negative NO on the TSTEPNL Bulk Data entry.

Limitations for the Current Release

In this pre-release, the following capabilities are not supported:

- Initial Condition in the nonlinear transient analysis.
- The nonlinear normal modes and the nonlinear buckling analysis.
- RFORCE and Creep.
- The arc-length method (input by NLPCI Bulk Data entry).
- The stress data recovery of the layer composite elements.
- Only the left-rotation-vector method, LANGLE=3, is allowed to process large rotation in nonlinear transient analysis.
- The nonlinear static analysis (ANALYSIS=NLSTAT) and the nonlinear transient analysis (ANALYSIS=NLTRAN) cannot be mixed in one SUBCASE.
- The line contact.

  The first 5 items will be supported in a future release. However, the last item ‘line contact’ will not be support in SOL 400. It will be replaced by the general contact capability in SOL 600.

In the following sections, how the new capabilities work in the current release is discussed.

Case Control Commands – SUBCASE, STEP and ANALYSIS

The STEP command was first introduced in MSC.Nastran 2004 for the nonlinear static analysis of SOL 400. In this release, it has been expanded to support the nonlinear transient analysis and again for SOL 400 only. User can specify different type of analysis at different SUBCASE and/or STEP by using the Case Control command ANALYSIS. Until now, ANALYSIS Case Control command supports the following 3 keywords. They are LNSTATIC, NLSTATIC (the default) and NLTRAN. The combination of SUBCASE, STEP and ANALYSIS commands will provide a mechanism for defining the multiple load steps, running multiple independent cases and, at same time, allowing the user to specify multiple types of analyses in one job.

The following examples illustrate the manner in which the SUBCASE, STEP and ANALYSIS commands are used.

- With one SUBCASE and multiple steps, 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. Note that it is not legal to assign two different keywords, NLSTATIC and NLTRAN, of ANALYSIS in one SUBCASE. The following is a typical example:

```
SUBCASE 1              $ This line can be omitted
   ANALYSIS = NLTRAN
   TSTEPNL = 200
   STEP 10
     DLOAD = 10
   STEP 20
     DLOAD = 20
   STEP 30
     DLOAD = 30
```

- Multiple SUBCASE’s may be executed in one job where the types of analysis, loads and boundary conditions can be changed. All SUBCASEs are independent from 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 can have different type of ANALYSIS. For example

```
SUBCASE 1
   ANALYSIS = NLSTAT   $ This line can be omitted
   NLPARM = 100
   STEP 110
     LOAD = 110
   STEP 120
     LOAD = 120
SUBCASE 2
     ANALYSIS = NLTRAN
     TSTEPNL = 200
     STEP 210
        DLOAD = 210
     STEP 220
        DLOAD = 220
```

In above example, the solutions of SUBCASE 1 and SUBCASE 2 are independent of each other. In case that the solution divergence is detected in a step, MSC.Nastran will terminate the solution of the current subcase and jump 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 placed below the step level. For example, all steps in above examples use the same Case Control command NLPARM = 100 in SUBCASE 1 and TSTEPNL = 200 in SUBCASE 2.
- 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 can be mixed in one SUBCASE. For example:

```
SUBCASE 10
     STEP 1
        ANALYSIS = LNSTATIC
        LOAD = 10
     STEP 2
        ANALYSIS = NLSTATIC
        LOAD = 20
        NLPARM = 20
```

In above 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., there is no 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 TSTEPNL in the nonlinear transient analysis. 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. These three error factors must satisfy the error tolerance rules specified by CONV, EPSU, EPSP, and EPSW on the Bulk Date entry TSTEPNL.

In computing the error factors, SOL 129 used the d-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 to account for the effect of Lagrange multipliers for the Lagrange rigid elements. 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 Quick Reference Guide, section (7.1).

Another major modification is the computation of the work error. In SOL 129, 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 are become smaller; therefore, the convergence rate of this value is proportional to the squire of the convergence rate of the solution. Thus it becomes 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. In this way, 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. Please note that this total work is only an approximation.

Solution Algorithm and Simulation of SOL 129

In SOL 400, the solution algorithm is modified in the following areas:

- Three new nonlinear iterations and stiffness update algorithm, AUTO, TSTEP (or ITER) and SEMI, are added on the Bulk Data entry TSTEPNL. (Please see the “**TSTEPNL Additions**” on page 63 for the details of TSTEPNL Bulk Data entry.) The old method, ADAPT, is still supported.
- The algorithm for load bi-sections.
- The algorithm for automatic time adjustment after converging at each time step.
- The solution divergence processing.

The stiffness update strategy as well as the direct time integration method is selected in the METHOD field of TSTEPNL Bulk Data entry. As mentioned above, there are four options:

- If the AUTO option is selected, the program automatically selects the most efficient strategy to adjust the incremental time step, perform the iterations, update the matrix, and use bisection. Please note the incremental time step adjustment is performed before the start of a time step. And bisection will be performed after solution iterations and stiffness updates when convergence cannot be achieved.
- If the TSTEP option is selected, the program updates the stiffness matrix every KSTEP iterations. The bisection will be applied only when the convergence cannot be reach by updating matrix. In this option, no automatic time step adjustment will be performed.
- If the ADAPT option is selected, the program automatically adjusts the incremental time step. If convergence cannot be achieved, then bisection will be performed. This method only updates matrix at every KSTEP convergent bisection solutions.
- If the SEMI option is selected, the program will update the stiffness matrix after the first iteration at each time step and then assume the normal AUTO option.

The stiffness matrix is always updated for a new STEP or Restart, irrespective of the option selected. The AUTO method has replace ADAPT to become the new default in the current release for ANALYSIS=NLTRAN in SOL 400.

For most of the problems we tested, SOL 400 gives equal or better performance than that of SOL 129. However, SOL 400 only output at user specified time steps in default. In order to give output similar to that of SOL 129, user can assign a negative NO on TSTEPNL to retrieve the output logic from SOL 129.

Nonlinear Iteration Summary Table for Nonlinear Transient Analysis in SOL 400

In order to allow the user to track the solution sequence during the nonlinear iteration, a detailed Nonlinear Iteration Summary Table is output. A line for each iteration is output on the F06 file during the nonlinear iteration. Due to printing of the average and the maximum displacements, the user will able to know the solution status before the end of the job. This is useful for large nonlinear problems. Even for small problems, the user will be able to know approximately how the analysis of a structural model performs by examining this table. An example of this table is given below and the descriptions of information given in this table are shown in **Table 2-3**.

Table 2-3 Nonlinear Iteration Summary Table

```
0                             N O N - L I N E A R   I T E R A T I O N   M O D U L E   O U T P U T

     STIFFNESS UPDATE TIME          0.02 SECONDS                                       SUBCASE     1130          STEP        1
            ITERATION TIME          0.00 SECONDS

              - TIME STEP -       - - ERROR FACTORS - -      CONV ITR MAT   AVG      TOTL       - - - - - DISP - - - - - -  NO. TOT TOT
    TIME      NO. BIS ADJUST ITR  DISP      LOAD      WORK    RATE DIV DIV R_FORCE   WORK      AVG        MAX      AT GRID C QNV KUD ITR

 5.00000E-02    1  0 1.0000  1 1.00E+00 3.30E-03 3.30E-03  1.00  0   1  1.1E-04 1.331E-05 2.63E-07  1.644E-06     1001 1  0   0    1
 5.00000E-02    1  0 1.0000  2 5.59E-07 1.83E-09 1.10E-06  0.00  0   1  1.0E-08 1.331E-05 2.63E-07  1.644E-06     1001 1  1   0    2
 1.00000E-01    2  0 1.0000  1 8.15E-01 4.92E-03 3.94E-03  1.00  0   1  4.3E-04 3.098E-04 1.42E-06  9.204E-06     1011 1  1   0    3
 1.00000E-01    2  0 1.0000  2 3.90E-07 3.31E-09 8.20E-07  0.00  0   1  2.8E-09 3.098E-04 1.42E-06  9.204E-06     1011 1  2   0    4
```

Table 2-4 Explanation of Information in Nonlinear Iteration Summary Table

TIME	The Current Time. It starts from 0.0 in the beginning of the 1st STEP and accumulates the value until at the end of the last STEP. To each STEP, the total time is determined by NDT and DT on the TSTEPNL Bulk Data entry.
TIME STEP NO	Number of time increment, including bisection. Initialize to 0 in the beginning of each STEP.
TIME STEP BIS	Number of bisections performed.
TIME STEP ADJUST	The ratio of the current time increment to the original DT on the TSTEPNL Bulk Data entry.
ITR	Number of iterations at each time increment.
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 EPSP on the TSTEPNL Bulk Data entry.
CONV RATE	Converge rate, which denotes 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 correction solution divergence will be taken if ITRDIV > MAXDIV.
MATDIV	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.

Table 2-4 Explanation of Information in Nonlinear Iteration Summary Table

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.
TOT KUD	Total number 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 the nonlinear transient analysis, only the following principles are listed in this release note:

- The restart must be continued at previous converged solution point in a nonlinear transient analysis by specifying a SUBCASE, STEP, and/or TIME. This is accomplished by using the Case Control command NLRESTART; please refer to “**Case Control Commands**” in Chapter 4 of the *MSC.Nastran Quick Reference Guide*.
- When a job has ANALYSIS=NLSTAT in SOL 400, it can restart at any user specified load steps (controlled by NOUT in NLPARM Bulk Data entry). The tremendous size of database should be required when ANALYSIS=NLTRAN in SOL 400 if the same logic mention above is used. To reduce the size of database and save the CPU time of I/O, a new parameter, NLPACK, is introduced in nonlinear transient analysis in SOL 400; please also see “**Outputs**” on page 52 e for the details of this new parameter. The nonlinear transient job can only restart at the closest output time step (controlled by NO on TSTEPNL Bulk Data entry), which is also the last time step of each output package (controlled by NLPACK parameter).

- 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.

The procedure to perform the restart for the nonlinear transient analysis is similar to the nonlinear static analysis in SOL 400; therefore, no further discussion here.

Temperature Excitation

A new capability, which has never been supported in the original nonlinear transient analysis (SOL 129), is added into SOL 400 when ANALYSIS=NLTRAN. It is the **time-dependent dynamic thermal effect**, which is applied to all the nonlinear elements in the residual.

The time-dependent thermal-elastic equation can be written as follows:

$$\varepsilon_T(t) = \alpha(T(t)) \cdot (T(t) - T_{ref}) - \alpha(T_0) \cdot (T_0 - T_{ref})$$

where:

$\varepsilon_T(t)$	=	the thermal strain
$T(t)$	=	the current temperature is defined in $T(t) = \{T_p\}f(t)$
		$\{T_p\}$ is the temperature field and $f(t)$ is the time function,
T_{ref}	=	the reference temperature,
T_0	=	the stress free temperature (initial temperature), and
(T)	=	the coefficient of thermal expansion

To all nonlinear elements, the temperature effect, in both static and transient, is directly handled as thermal strain in SOL 400 when computing the element forces. Unlike all the other linear and nonlinear analyses, the thermal load is not created to the nonlinear elements anymore in SOL 400.

In MSC.Nastran 2004, the thermal effect has been added into nonlinear static analysis in SOL 400. To support it in nonlinear transient analysis, two new bulk data entries are created in the current release. They are TTEMP and TMPSET. Basically, TTEMP is to define a time-dependent dynamic thermal field, T(t), in the same form as TLOAD1. At the same time, TMPSET is to define a group of grid points, which refers to the same TTEMP Bulk Data entry. Please see **“New Bulk Data Entries, TTEMP and**

TMPSET” on page 64 for the details of these two new bulk data entries. By using TTEMP and TMPSET, the whole model can be separated into finite sub-regions and each sub-region can have its own temperature distribution pattern. If it is necessary, user can also make every grid point as a independent sub-region or make the whole model as a single sub-region.

Same as nonlinear static analysis, TEMP(INIT) and TEMP(LOAD) commands are used in the Case Control file to define the temperature input in nonlinear transient analysis. The SID of TEMP(LOAD) can refer to TTEMP (and TMPSET) but not the TEMP(INIT). The temperature of any grid point, whose ID is not listed in the TMPSET, will be interpolated linearly in the same way as nonlinear static analysis. In other words, when there is no TTEMP (and TMPSET) in the Bulk Data file, the TEMP(LOAD) will refer to the TEMP (or TEMPD, TEMPP1,...,etc.) directly and a linear interpolation scheme will be used to determine the temperature filed in any specified time.

User should only set one temperature set (SID) in each STEP but this rule is only forced in the nonlinear elements. To all the linear elements, user can still use DLOAD bulk data entry to combine multiple TLOAD1 and TLOAD2’s, whose EXCITE_ID reference thermal load, to support multiple sets of temperature loads – this is also known as “static load for dynamics” to the temperature loads. However, it is user’s responsibility to explain the physical meanings.

Note that all the upper stream superelements and all linear elements in the residual are still used the original concept, “static load for dynamics” to input the thermal effect when ANALYSIS=NLTRAN in SOL 400. The TEMP(LOAD) and all its corresponding temperature related bulk data entries introduced above can only describe the thermal effect to the nonlinear elements in the residual. If there is no DLAOD Case Control command to define the temperature load in the “static load for dynamics” way, the temperature effect to the linear part of the structure will be lost.

Outputs

The outputs are requested by using the Case Control commands. All existing output requesting Case Control commands such DISPLACEMENT, VELOCITY, ACCELERATION, STRESS, NLSTRESS, OLOAD, SPCFORCE, etc., are also allowed in the nonlinear transient analysis in SOL 400.

Two special outputs, “Nonlinear Iteration Summary Table” and “PARAM, PH2OUT”, which have been introduced in MSC.Nastran 2004, are also available on the nonlinear transient analysis in SOL 400. In addition, a new output control, PARAM, NLPACK, is added in this release for nonlinear transient analysis only.

This new parameter, NLPACK (=100 is the default), is used to control the packed output in SOL 400. The value of NLPACK represents the total number of output time steps in one output package. SOL 400 will process the output procedure only after collecting all "NLPACK" output time steps or at the end of each STEP case. Note that NLPACK=-1 means to collect all output times steps in a STEP case and then output them all together, which is the same as SOL 129 output method. This parameter only used in ANALYSIS=NLTRAN and restart is only possible at the end of each "NLPACK" output time step. To nonlinear static analysis (ANALYSIS=NLSTAT), NLPACK is always equal to 1.

User Interfaces

The user interfaces, which are important or new to the nonlinear transient analyses in SOL 400, are summarized in this section. For details, please refer to the ***MSC.Nastran Quick Reference Guide***.

NASTRAN System Cells

- STPFLG (SYSTEM (366)) – Selects the SUBCASE or STEP layout when there are a number of SUBCASE commands and no STEP command in a Case Control file.
- ITRFMT (SYSTEM (401)) – Selects the convergence parameter computation method and the divergence solution checking method to simulate the **SOL 129**. If ITRFMT = -1, use method similar to SOL 129.
- TZEROMAX (SYSTEM (373)) – Controls initial time step adjustment in nonlinear transient analysis.

File Management Statements

The following File Management statements are required for restarts. Please refer to the "**File Management Statements**" in Chapter 2 of the *MSC.Nastran Quick Reference Guide* or Chapter 12 of the *MSC.Nastran Reference Manual* for details.

- ASSIGN – Assigns physical file names to database files that are used by a Nastran data file to run a job.
- RESTART – Requests that data stored in a previous run be used in the current run.

Executive Control Statement

- 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. The default value is 3 for the nonlinear transient analysis.
- PARAM, LGDISP – Requests a geometric nonlinear analysis.
- PARAM, FOLLOWK – Requests whether the follower force stiffness will be used in a geometric nonlinear analysis.
- PARMA, FKSYMFAC – 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 Bulk Data entry NLPARM.
- PARAM, PH2OUT – Requests phase II outputs for a nonlinear analysis.
- PARAM, NLPACK – Control the total output time step in one output package, see “**Outputs**” on page 52.
- PARAM, NDAMP – Specifies the α value (a numerical damping) of the HHT-α method in SOL 400.

Case Control Commands

- ANALYSIS – Selects solution method for an analysis step, see “**Case Control Commands**” on page 54.
- NLRESTART – Requests a restart execution at a specific solution point for SOL 400, see “**Restart**” on page 50.
- NLSTRESS – Requests the form and type of the nonlinear element stress output.
- STEP – Delimits and identifies an analysis step, see “**Case Control Commands**” on page 54.

Bulk Data Entries

- MATHP- Specifies the hyperelastic material properties for an element.

- MATS1 – Specifies the stress-dependent material properties for an element.
- TSTEPNL – Defines a set of parameters for nonlinear transient analysis iteration strategy.
- TTEMP – Defines a time-dependent dynamic temperature distribution in nonlinear transient response.
- TMPSET – Defines a time-dependent dynamic thermal load group for use in TTEMP Bulk Data entry.

Examples

The following three examples show the inputs of the nonlinear transient analysis. The intention of these examples is to show the input structure for SOL 400. The model itself and the detailed entries in the Bulk Data file are not important.

Example 1

Example one, EX01, is simplified from the standard QA file, NLTSUB02. This model only has QUAD4 elements. It has both material nonlinearity (MATS1) and geometrical nonlinearity (PARAM, LGDISP, 1). The 1st STEP will process the output data at every 5 output time steps and the 2nd STEP do it only once because of the settings of the parameter NLPACK. All the **bold**-font statements are entries pertaining to the nonlinear analysis.

```
ID MSC, EX01    $
TIME 150          $
SOL 400           $
CEND
TITLE=ISOTROPIC MATERIAL & MATS1, ELLIPTIC CYLINDER UNDER EX01
SUBTITLE =SPC CHANGE IN EACH STEP, NLPACK's
SET 10 = 10000,11200
SET 20 = 101
  SEALL = ALL
  DISPL = ALL
  STRESS = 20
$
SUBCASE 100
  ANALYSIS=NLTRAN
STEP     10
  PARAM,NLPACK,5
  DLOAD   = 100
  SPC     = 200
  TSTEPNL = 310
STEP     20
  PARAM,NLPACK,-1
  DLOAD   = 100
  SPC     = 400
  TSTEPNL = 320
```

```
$
BEGIN BULK
PARAM   NDMAP   0.0
PARAM   LGDISP  1
TSTEPNL 310     100     0.01    10      AUTO
TSTEPNL 320     100     0.01    10      AUTO
$
PLOAD4  510     101     5.                              THRU    112
$
TLOAD1  100     510     0       0       120
TABLED1 120                                                             +TBD1
+TBD1   0.      0.      5.      1.      16.     1.      ENDT
MAT1    100     3.+7            0.3     .283-2
MAT1    101     3.+7            0.3     .283-2
MATS1   100             PLASTIC 3.+5                    500000.
$
GRID    10000           100.    0.0     10.             345
GRID    10001           100.    0.0     0.0             345
GRID    10100           99.3625 3.30491 10.             345
GRID    10101           99.3625 3.30491 0.0             345
GRID    10200           96.8149 6.51543 10.             345
GRID    10201           96.8149 6.51543 0.0             345
GRID    10300           92.5105 9.59323 10.             345
GRID    10301           92.5105 9.59323 0.0             345
GRID    10400           86.6025 12.5    10.             345
GRID    10401           86.6025 12.5    0.0             345
GRID    10500           79.2443 15.1974 10.             345
GRID    10501           79.2443 15.1974 0.0             345
GRID    10600           70.5889 17.6472 10.             345
GRID    10601           70.5889 17.6472 0.0             345
GRID    10700           60.7898 19.8111 10.             345
GRID    10701           60.7898 19.8111 0.0             345
GRID    10800           50.     21.6506 10.             345
GRID    10801           50.     21.6506 0.0             345
GRID    10900           38.3729 23.1276 10.             345
GRID    10901           38.3729 23.1276 0.0             345
GRID    11000           26.0617 24.2037 10.             345
GRID    11001           26.0617 24.2037 0.0             345
GRID    11100           13.2197 24.8406 10.             345
GRID    11101           13.2197 24.8406 0.0             345
GRID    11200           0.0     25.     10.             345
GRID    11201           0.0     25.     0.0             345
$
CQUAD4  101     100     10000   10001   10101   10100
CQUAD4  102     100     10100   10101   10201   10200
CQUAD4  103     100     10200   10201   10301   10300
CQUAD4  104     100     10300   10301   10401   10400
CQUAD4  105     100     10400   10401   10501   10500
CQUAD4  106     100     10500   10501   10601   10600
CQUAD4  107     100     10600   10601   10701   10700
CQUAD4  108     100     10700   10701   10801   10800
CQUAD4  109     100     10800   10801   10901   10900
CQUAD4  110     100     10900   10901   11001   11000
CQUAD4  111     100     11000   11001   11101   11100
```

```
CQUAD4  112     100     11100   11101   11201   11200
$
PSHELL  100     100     0.10    100             101
$
SPC1    200     16      11200   11201
SPC1    200     26      10000   10001
$
SPC1    400     16      11200   11201
SPC1    400     26      10000   10001
SPC1    400     1       10700
SPC1    400     2       10701
$
ENDDATA
```

Example 2

Example two, EX02, is modified from the standard QA file, NLTSUB02. It shows two different types of analyses in the same job. This model is similar to the Example one except adding some static loads and NLPARM's. All the **bold**-font statements are entries that show difference in two different types of analyses.

```
ID MSC, EX02 $
TIME 150          $
SOL 400           $
CEND
TITLE=TEST MIXED ANALYSES - NLSTAT AND NLTRAN                EX02
SUBTITLE =SPC CHANGE IN THE STEPS IN EACH SUBCASE
SET 10 = 10000,11200
SET 20 = 101
  SEALL = ALL
  DISPL = ALL
  STRESS = 20
$
SUBCASE 100
  ANALYSIS=NLSTAT
STEP     10
  LOAD    = 800
  SPC     = 200
  NLPARM  = 110
STEP     20
  LOAD    = 900
  SPC     = 400
  NLPARM  = 120
$
SUBCASE 200
  ANALYSIS=NLTRAN
STEP     10
  DLOAD   = 100
  SPC     = 200
  TSTEPNL = 310
STEP     20
  DLOAD   = 100
```

```
  SPC     = 400
  TSTEPNL = 320
$
BEGIN BULK
NLPARM  110     10              AUTO                        YES
NLPARM  120     10              AUTO                        YES
$
LOAD    800     0.01    1.0     510
LOAD    900     0.05    1.0     510
(… The rest is same as what in the Bulk Data Deck in the 1st Example…)
ENDDATA
```

Example 3

Example three, EX03, is modified from the standard QA file, NLTTL002. This model only has 1 QUAD4 element and 2 TRAI3 elements. Its major purpose is to show the various combinations of TTEMP and TMPSET inputs in nonlinear transient analysis for the thermal effect. All the **bold**-font statements are entries related to the temperature related inputs.

```
ID MSC, EX03 $
SOL 400
DIAG 8,15
TIME 60
CEND
SEALL = ALL
SUPER = ALL
TITLE = THERMAL LOAD TEST FOR NONLINEAR TRANSIENT ANALYSIS    EX03
SUBTITLE = Q4/T3 MODEL, TTEMP AND TMPSET
$ECHO = NONE
MAXLINES = 999999999
$
TEMPERATURE(INITIAL) = 1
SUBCASE 1
analysis=NLTRAN
step 1
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 3
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
step 2
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 4
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
SUBCASE 2
analysis=NLTRAN
step 3
```

```
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 5
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
step 4
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 6
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
SUBCASE 3
analysis=NLTRAN
step 5
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 7
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
step 6
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 8
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
SUBCASE 4
analysis=NLTRAN
step 7
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 9
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
step 8
   TSTEPNL= 1
   SPC = 2
   TEMPERATURE(LOAD) = 10
   DISPLACEMENT(SORT1,REAL)=ALL
   nlstress = all
   stress = all
$
BEGIN BULK
PARAM    POST    -1
PARAM   COUPMASS 1
PARAM    LGDISP  1
PARAM    K6ROT   100.
PARAM,NOCOMPS,-1
PARAM   PRTMAXIM YES
PARAM,COMPMATT,YES
```

```
PARAM,EPSILONT,INTEGRAL
PARAM   NLTOL    0
TSTEPNL,1,4,0.25,1,AUTO
$
PCOMP    1                                     79.     0.
*        1             .04875         0.              YES
$
MAT8     1       7.15+6  2.9+6   .29     1.4+6                   1.9-4
         2.9-6   6.-6    79.
MATT8    1       3       5       4       6
         1       2
$
TABLEM1  1                                                              +     CR
+     CR 60.     2.9-6   70.     2.9-6   80.     3.24-6  100.    3.86-6 +     CS
+     CS 120.    4.01-6  140.    3.89-6  150.    3.78-6  160.    3.68-6 +     CT
+     CT 180.    3.52-6  200.    3.47-6  220.    3.55-6  240.    3.76-6 +     CU
+     CU 250.    3.87-6  260.    3.99-6  280.    4.12-6  300.    4.24-6 +     CV
+    CV 320.   4.24-6  ENDT
$
TABLEM1  2                                                              +     CW
+     CW 60.     6.-6    70.     6.-6    80.     7.67-6  100.    1.168-5+     CX
+     CX 120.    1.341-5 140.    1.37-5  150.    1.349-5 160.    1.328-5+     CY
+     CY 180.    1.266-5 200.    1.222-5 220.    1.218-5 240.    1.259-5+     CZ
+     CZ 250.    1.296-5 260.    1.334-5 280.    1.415-5 300.    1.46-5 +     DA
+    DA 320.   1.46-5  ENDT
$
TABLEM1  3                                                              +     BX
+     BX 60.     7.15+6  70.     7.15+6  80.     7.15+6  100.    7.13+6 +     BY
+     BY 120.    7.11+6  140.    7.08+6  150.    7.07+6  160.    7.07+6 +     BZ
+     BZ 180.    7.06+6  200.    7.05+6  220.    7.05+6  240.    7.04+6 +     CA
+     CA 250.    7.04+6  260.    7.05+6  280.    7.06+6  300.    7.08+6 +     CB
+    CB 320.   7.08+6  ENDT
$
TABLEM1  4                                                              +     CM
+     CM 60.     .29     70.     .29     80.     .29     100.    .29    +     CN
+     CN 120.    .29     140.    .29     150.    .29     160.    .29    +     CO
+     CO 180.    .29     200.    .29     220.    .29     240.    .29    +     CP
+     CP 250.    .29     260.    .29     280.    .29     300.    .29    +     CQ
+    CQ 320.   .29    ENDT
$
TABLEM1  5                                                              +     CC
+     CC 60.     2.9+6   70.     2.9+6   80.     2.9+6   100.    2.82+6 +     CD
+     CD 120.    2.75+6  140.    2.68+6  150.    2.64+6  160.    2.58+6 +     CE
+     CE 180.    2.47+6  200.    2.35+6  220.    2.22+6  240.    2.09+6 +     CF
+     CF 250.    2.03+6  260.    1.95+6  280.    1.8+6   300.    1.65+6 +     CG
+    CG 320.   1.65+6  ENDT
$
TABLEM1  6                                                              +     CH
+     CH 60.     1.4+6   70.     1.4+6   80.     1.4+6   100.    1.34+6 +     CI
+     CI 120.    1.29+6  140.    1.24+6  150.    1.22+6  160.    1.2+6  +     CJ
+     CJ 180.    1.15+6  200.    1.1+6   220.    980000. 240.    870000.+     CK
+     CK 250.    810000. 260.    750000. 280.    620000. 300.    500000.+     CL
+     CL 320.    500000. ENDT
$
cquad4,1,1,1,2,5,4
ctria3,2,1,1,2,4
ctria3,3,1,2,5,4
```

```
$
GRID     1               0.00000 0.00000 0.00000
GRID     2               1.00000 0.00000 0.00000
GRID     4               0.00000 1.00000 0.00000
GRID     5               1.00000 1.00000 0.10000
$
SPCADD   2       1
SPC1     1       123456  1       2
spc1     1       123456  4
$
TTEMP,3,111,300
TMPSET,111,4,5
TTEMP,3,101,310
TMPSET,101,1,2
$
TTEMP,4,102,400
TMPSET,102,1,2,4,5,7,8,9,
,10,11,12
$
TTEMP,5,201,500
TMPSET,201,1,2,4,5
$
TTEMP,6,-1,400
$
TTEMP,7,202,700
TMPSET,202,1,2
$
TTEMP,8,204,800
TMPSET,204,1,2
$
TTEMP,9,402,900
TMPSET,402,1,2,4,5
$
TEMP     1       1       79.
TEMP     1       2       79.
TEMP     1       4       79.
TEMP     1       5       79.
$
TEMP     3       1       80.
TEMP     3       2       80.
TEMP     3       4       80.
TEMP     3       5       80.
TABLED1 300
        0.0     .9875   1.0     1.0     ENDT
TABLED1 310
        0.0     .9875   1.0     1.0     ENDT
$
TEMP     4       1       81.
TEMP     4       2       81.
TEMP     4       4       81.
TEMP     4       5       81.
TABLED1 400
        1.0     .9876542 2.0    1.0     ENDT
$
```

```
TEMP     5       1       80.
TEMP     5       2       80.
TEMP     5       4       80.
TEMP     5       5       80.
TABLED1 500
        0.0     .9875    1.0    1.0     ENDT
$
TEMP     6       1       81.
TEMP     6       2       81.
TEMP     6       4       81.
TEMP     6       5       81.
$
$
TEMP     7       1       80.
TEMP     7       2       80.
TEMP     7       4       80.
TEMP     7       5       80.
TABLED1 700
        0.0     .9875    1.0    1.0     ENDT
$
TEMP     8       1       81.
TEMP     8       2       81.
TEMP     8       4       81.
TEMP     8       5       81.
TABLED1 800
        1.0     .9876542 2.0    1.0     ENDT
$
TEMP     9       1       80.
TEMP     9       2       80.
TEMP     9       4       80.
TEMP     9       5       80.
TABLED1 900
        0.0     .9875    1.0    1.0     ENDT
$
TEMP     10      1       81.
TEMP     10      2       81.
TEMP     10      4       81.
TEMP     10      5       81.
ENDDATA
```

TSTEPNL Additions

TSTEPNL Parameters for Nonlinear Transient Analysis

Field	Contents
METHOD	Method for controlling stiffness updates and direct-time-integration strategy. See Remark 4. (Character="AUTO","TSTEP","ADAPT" or "SEMI"; Default="AUTO")
KSTEP	The criteria for the stiffness matrix update. See Remark 5. (Integer>0; Default=2)

Remarks:

4. The stiffness update strategy as well as the direct time integration method is selected in the METHOD field.

   - If the AUTO option is selected, the program automatically selects the most efficient strategy to update matrix, adjust the incremental time step and use bisection.
   - If the TSTEP option is selected, the program updates the stiffness matrix every KSTEP increments of time step. The bisection will be applied only when the convergence cannot be reach by updating matrix and no automatic time step adjustment.
   - If the ADAPT option is selected, the program automatically adjusts the incremental time step and bisection. This method only updates matrix at every KSTEP convergent bisection solutions.
   - If the SEMI option is selected, the program will update the stiffness matrix after the first iteration at each time step and then assume the normal AUTO option.

   The stiffness matrix is always updated for a new STEP or Restart, irrespective of the option selected.

5. For AUTO and SEMI options, the stiffness matrix is updated on convergence if KSTEP is less than the number of iterations that were required for convergence with the current stiffness. For ADAPT option, stiffness is updated every KSTEP converged bisection solutions. For TSTEP, stiffness is updated at every KSTEP iteration at a time step interval.

New Bulk Data Entries, TTEMP and TMPSET

TTEMP

Temperature Distribution of Transient Response for Dynamic Thermal Excitation

Define a time-dependent dynamic thermal distribution in the same form as TLOAD1

$$\{T(t)\} = \{A(T) \cdot F(t)\}$$

where $A(T)$ defines the temperature field and $T(t)$ is the temperature distribution for use in the nonlinear elements in nonlinear transient analysis.

Format:

1	2	3	4	5	6	7	8	9	10
TTEMP	SID	GROUP_ID	TID						

Example:

1	2	3	4	5	6	7	8	9	10
TTEMP	11	101	31						

Field	Contents
SID	Temperature set identification number. (Integer >0)
GROUP_ID	Temperature group identification number (Integer >0 or =-1)
TID	Identification number of TABLEDi entry that gives F(t). (Integer >0)

Remarks:

1. SID is defined in Case Control file by TEMP(LOAD)=SID.
2. This entry is used in SOL 400 only when ANALYSIS=NLTRAN (nonlinear transient analysis) and the temperature load is applied. It only applies to the nonlinear elements in the Residual (SEID=0). There should be only one temperature set for each STEP.
3. GROUP_ID determines the time-dependent distribution of temperatures. It references to TMPSET Bulk Data entry to define all grid points, which reference the same TABLEDi entry. Each grid point can have its own GROUP_ID if it is necessary. GROUP_ID=-1 means all grid points are in one group and reference to the same TTEMP Bulk Data entry.
4. TEMP(INIT) should not reference TTEMP and TMPSET.

TMPSET Temperature Group Set Definition

Define a time-dependent dynamic thermal load group for use in TTEMP Bulk Data entry.

Format:

1	2	3	4	5	6	7	8	9	10
TMPSET	ID	G1	G2	G3	G4	G5	G6	G7	

Alternate Format:

TMPSET	ID	G1	"THRU"	G2	"BY"	INC			

Example:

The Continuation Entry formats may be used more than once and in any order. They may also be used with either format above.

Continuation Entry Format 1:

	G8	G9	G10	G11	-etc.-				

Continuation Entry Format 2:

	G8	"THRU"	G9	"BY"	INC				

Example:

TMPSET	15	5	THRU	21	BY	4			
	27	30	32	33					
	35	THRU	44						
	67	68	72	75	84	93			

Field	Contents
ID	Temperature group identification number (Integer >0)
Gi	Grid point Identification numbers in the group (Integer >0)

Remarks:

1. This entry is used in SOL 400 only when ANALYSIS=NLTRAN (nonlinear transient analysis) and the temperature load is applied. It only applies to the nonlinear elements in the Residual (SEID=0).
2. GROUP_ID determines the group to a specified the time-dependent distribution of temperatures. It is used by TTEMP Bulk Data entry to define the corresponding TABLEDi entry. GROUP_ID must be unique for all the other TMPSET entries.
3. TEMP(INIT) should not reference to TTEMP and TMPSET.

2.4 Correction in the Solution Algorithm for Elasto-Plastic Material

The solution algorithm for Elasto-Plastic material in nonlinear analyses works best for single-hardening-slope in MSC.Nastran 2004 and earlier versions. When there are multiple-hardening-slopes, which defined on the TABLES1 Bulk Data entry as stress-strain curve, MSC.Nastran may produce some numerical error after loading procedure processes into the 2nd and higher hardening slopes.

This numerical error is derived from the calculation of the scalar multiplier (a Lagrange multiplier) $d\lambda$.

$$d\lambda = \frac{\left\{\frac{\partial f}{\partial \sigma}\right\}^T [D_e]\{d\varepsilon\}}{H^* + \left\{\frac{\partial f}{\partial \sigma}\right\}^T [D_e]\left\{\frac{\partial f}{\partial \sigma}\right\}} \qquad \text{Eq. 2-1}$$

where the gradient vector $\{\partial f / \partial \sigma\}$, is computed by differentiating the stress function $f(\sigma)$ representing effective stress. $[D_e]$ is the elasticity matrix, $\{d\varepsilon\}$ is the strain increment and H^* is the slope of hardening.

In Plasticity analysis, when a sub-increment along the stress-strain curve crossing two hardening-slopes, for example H_1 and H_2, the old solution algorithm will picks up one of them, depending on the estimate stress increment, to be the value of H^*. H^* is always equal to either H_1 or H_2; however, none of them can represent the "true" hardening slope in this condition - that's how the numerical error building-up. In other word, the "true" value of H^* must be the function of both H_1 and H_2 when it goes through two hardening-slopes.

A new iterative algorithm is added in this release to compute the "true" value of H^* based on the two hardening-slopes (H_1 and H_2) it passed. Simply to say, we must find H^* based on the **Eq. 2-1** and the following two equations:

$$H^* d\varepsilon^P = H_1 d\varepsilon_1^P + H_2 d\varepsilon_2^P \qquad \text{Eq. 2-2}$$

and

$$d\varepsilon^P = d\varepsilon_1^P + d\varepsilon_2^P \qquad \text{Eq. 2-3}$$

where $d\varepsilon^P$ is the total plastic strain increment in one sub-increment and $d\varepsilon_1^P$ and $d\varepsilon_2^P$ are the plastic strain increments corresponding to hardening-slope H_1 and H_2.

An iterative method has been added to MSC.Nastran when computing **Eq. 2-1** , **Eq. 2-2** and **Eq. 2-3** to obtain the best (or say converged) H^* .

Note that even though MSC.Nastran 2005 can handle multiple-hardening-slope case in Plasticity, it is still not recommended to loading too fast in nonlinear analyses. That because if a sub-increment crossing three or more hardening slopes, the converged H^* may be very hard to obtain.

Example

The following example shows the difference between Theoretical results, MSC.Nastran 2004 results and MSC.Nastran 2005 results.

```
ID    PLASTICITY
SOL   400
DIAG        8
CEND
  NLPAR           = 400
  SET 1           = 12 23 34
  SET 2           = 11 22 31
  DISPL           = 1
STEP    1
  LOAD            = 100
STEP    2
  LOAD            = 200
STEP    3
  LOAD            = 300
STEP    4
  LOAD            = 400
STEP    5
  LOAD            = 500
STEP    6
  LOAD            = 600
STEP    7
  LOAD            = 700
STEP    100
BEGIN BULK
GRID        11          0.    0.                123456
GRID        12          0.    10.               13456
CROD        1     101   11    12
PROD        101   100   1.
MAT1        100   1.+7        .3
MATS1       100   101   PLASTIC                       100.
TABLES1           101                                             +T
+T    0.          0.    1.-5  100.  2.-5  195.  3.-5  285.  +U
+U    4.-5        370.  ENDT
$TABLES1          102                                             +V
$+V   0.          0.    1.-5  100.  4.-5  370.  ENDT
FORCE       100   12          100.        1.
FORCE       200   12          150.        1.
FORCE       300   12          195.        1.
```

```
FORCE       400   12          250.        1.
FORCE       500   12          285.        1.
FORCE       600   12          330.        1.
FORCE       700   12          370.        1.
NLPARM            400   1           ITER  1                       +E
+E          1.-6  1.-12                               +B
+B    0
ENDDATA
```

The displacement results at the tip of the ROD element are listed here

STEP	Theoretical	MSC.Nastran 2004	MSC.Nastran 2005
1	1.000000E-04	1.000000E-04	1.000000E-04
2	1.526316E-04	1.526316E-04	1.526316E-04
3	2.000000E-04	2.000000E-04	2.000000E-04
4	2.611111E-04	2.611111E-04	2.611111E-04
5	3.000000E-04	3.000000E-04	3.000000E-04
6	3.529412E-04	3.635294E-04	3.529412E-04
7	4.000000E-04	4.105882E-04	4.000000E-04
100	3.000000E-05	4.058823E-05	2.999997E-05

Note that in the loading procedure, from STEP1 to STEP 7, MSC.Nastran 2004 built numerical error gradually but not in MSC.Nastran 2005 when comparing with the theoretical results. The last STEP, STEP 100, is an unloading procedure, MSC.Nastran 2005 can obtain the very close result as the theoretical one but MSC.Nastran 2004 has about 35% error in the residual displacement.

2.5 Correction for the Nonlinear Element Strain Energy

By using the Case Control command, ESE, the user can ask NASTRAN job to compute and output the element strain energy in all linear analyses for a long time. This capability has been expanded to nonlinear elements in SOL 106 since MSC.Nastran 2001 and to all nonlinear elements. The element strain energy of each element was computed in the following way.

$$SE^{i+1} = SE^{i_0} + \frac{1}{2}\left(u^{i+1} - u^{i_0}\right)\left(F^{i+1} + F^{i_0}\right) \qquad \text{Eq. 2-4}$$

where SE represents the element strain energy of each element, u represents the displacement vector, F represents the element force vector and i represents the load increment. Note that i_0 represents the most recent output load increment, which controlled by INTOUT in NLPARM Bulk Data entry and $i_0 \le i$. This formulation made a limitation - the correct result can only be obtained when INTOUT=ALL, in other words, $i_0 = i$. Otherwise, the result is approximate when INTOUT=YES or NO and $i_0 \le i$. Specially, it is very difficult to obtain a correct result when INTOUT=NO.

In order to remove the above limitation, the formulation of element strain energy has been modified in MSC.Nastran 2005. It becomes

$$SE^{i+1} = SE^{i} + \frac{1}{2}(u^{i+1} - u^{i})(F^{i+1} + F^{i}) \qquad \text{Eq. 2-5}$$

On the other hand, the element strain energy is not output-request-dependent anymore in this release. No matter if INTOUT is set to ALL, YES or NO, the correct element strain energy is calculated at the end of each loading case. By the way, the capability of ESE, together with GPFORCE, has also been added into SOL 400 when ANALYSIS=NLSTAT in MSC.Nastran 2005.

Example

The following example shows the difference of the element strain energy in MSC.Nastran 2004 and MSC.Nastran 2005 when INTOUT=ALL or NO.

```
IN MSC, gpf001a $
SOL 106
TIME 10
DIAG 8, 15 $
CEND
TITLE=GPF001A - NONLINEAR GPFORCE TEST PROBLEM
SUBTI-HEXA ELEMENTS + AXIAL FORCES = NLM
LABEL=NONLINEARITY ANALYSIS - ITER
SPCF=ALL
```

```
DISPL=ALL
$STRESS(PRINT,PUNCH)=10
GPFORCE=ALL
ESE=ALL
SPC=100
NLPARM=1
SUBCASE 10
LOAD=100
SKIPON

SUBCASE 20
LOAD=200
SUBCASE 30
LOAD=300
SUBCASE 40
LOAD=400
SUBCASE 50
LOAD=500
BEGIN BULK
PARAM,GRDEQ,0
PARAM,LGDISP,-1
NLPARM,1,4,,ITER,1,10,,ALL
$NLPARM,1,4,,ITER,1,10,,NO
$
$ 8 NODE CHEXA MODEL
$
FORCE,100,5,,100000.0,1.0
FORCE,100,6,,100000.0,1.0
FORCE,100,7,,100000.0,1.0
FORCE,100,8,,100000.0,1.0
$
FORCE,200,5,,50000.0,1.0
FORCE,200,6,,50000.0,1.0
FORCE,200,7,,50000.0,1.0
FORCE,200,8,,50000.0,1.0
$
FORCE,300,5,,10000.0,1.0
FORCE,300,6,,10000.0,1.0
FORCE,300,7,,10000,1.0
FORCE,300,8,,10000.0,1.0
$
FORCE,400,5,,-10000.0,1.0
FORCE,400,6,,-10000.0,1.0
FORCE,400,7,,-10000.0,1.0
FORCE,400,8,,-10000.0,1.0
$
FORCE,500,5,,-100000.0,1.0
FORCE,500,6,,-100000.0,1.0
FORCE,500,7,,-100000.0,1.0
FORCE,500,8,,-100000.0,1.0
$
GRID  1                       0.          0.          0.
GRID  2                                   1.
GRID  3                                   1.          1.
```

```
GRID   4                                                    1.
GRID   5                        10.
GRID   6                        10.         1.
GRID   7                        10.         1.          1.
GRID   8                        10.                     1.
SPC1   100          123         1           2           3           4
CHEXA  1            8           1           2           3           4     56
+CHEX1
+CHEX1 7            8
$
$ COMMON DATA FOR EACH PROBLEM
$
PSOLID8             1           0
MAT1   1            3.+7                    0.3
MATS1  1                        PLASTIC     1.+7        1           1    3.+4
GRDSET                                                                    456
ENDDATA
```

The results of element strain energy of the CHEXA element in the SUBCASE 100 are listed here

	MSC.Nastran 2004		MSC.Nastran 2005	
INTOUT	**ALL**	**NO**	**ALL**	**NO**
	8.583713E+04	2.175758E+04	8.583713E+04	8.583713E+04

CHAPTER

3 Numeric Enhancements

- ACMS Now Available in the Matrix (DOF) Domain
- Improvements for Geometric Domain Based ACMS
- Improved Matrix Diagonal Diagnostics for 2x2 Pivots (MAXRATIO)
- Performance Improvement in Modal Frequency Response for Large Frequency Ranges

3.1 ACMS Now Available in the Matrix (DOF) Domain

Previously, ACMS was available in the Geometric Domain. The initial domain decomposition, which divides the model into smaller sub-models, took place on the model geometry, on the set of grid points and their element connections. Now, ACMS is also available in the DOF domain, which postpones the domain decomposition until after all constraints have been eliminated, at the matrix level.

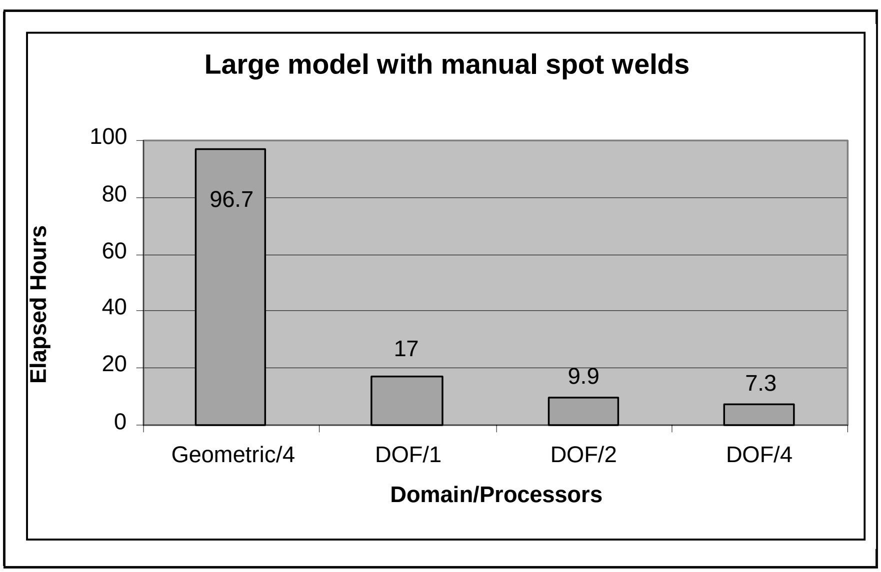
DOF Domain ACMS is invoked with the Executive level command DOMAINSOLVER “PARTOPT” option:

```
DOMAINSOLVER  ACMS (PARTOPT=DOF)
```

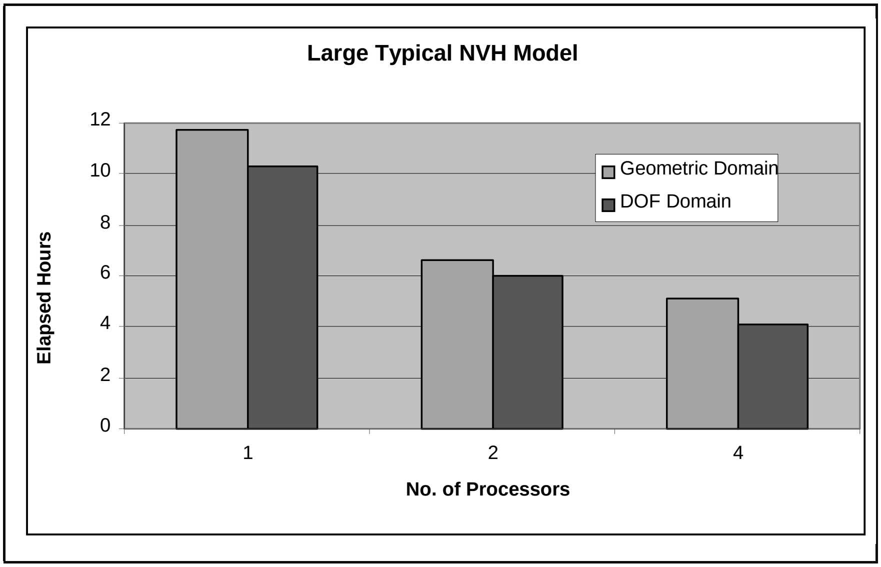
The primary advantage of DOF Domain ACMS is performance, especially for models whose complicated geometry presents problems to Geometric Domain ACMS. Examples of such modeling include the following:

- Spot welds via manual modeling technique
- Spot welds via CWELD elements
- Acoustic coupling via rigid elements and MPCs
- Large numbers of rigid elements and / or MPCs for any purpose

The following chart shows an example of the performance gain for a model that uses manual spot welding techniques.



DOF Domain ACMS performs at least as well as Geometric Domain ACMS for models that do not feature these special characteristics, as shown in the following chart.



Other important features of DOF Domain ACMS include:

- A Component Modal Synthesis theory identical to that used in ACMS and in all other MSC.Nastran CMS techniques.
- Residual vectors that are employed at every component at every level in order to maximize the accuracy of the CMS.
- DMP (Distributed Memory Parallel) available up to eight processors, providing excellent parallel speedup.
- Full compatibility with MSC.Nastran superelement techniques, including External Superelements.
- Fully integrated with the SPCD method of enforced motion in SOL 111.
- Fully integrated with Acoustic Panel Participation in SOL 111 fluid-structure interaction.
- Available in SOLs 103, 111, and 200.
- Available with external superelements

3.2 Improvements for Geometric Domain Based ACMS

Error corrections include the following:

- Improved robustness by correcting error related to free-floating (i.e., unconnected) scalar points.
- Fixed key errors in the ACMS interface with transient analysis and MAXMIN data recovery.
- Fixed errors related to CWELD elements.
- Fixed key error related to parallel execution for acoustic models.

In addition, enhancements were made to make grid-based ACMS work with the new External Superelement capability.

3.3 Improved Matrix Diagonal Diagnostics for 2x2 Pivots (MAXRATIO)

Matrix-to-factor diagonal ratios are a long-standing tool that help determine model integrity in MSC.Nastran. With the increased use of Lagrange Multiplier Technique (LMT) variables, 2-by-2 pivoting employed by the sparse direct solver in MSC.Nastran has become more common. In the event of 2-by-2 pivoting, the matrix-to-factor diagonal ratio was computed incorrectly prior to MSC.Nastran 2004 r3, with the value of 1.0 used as the factor diagonal value for that DOF. If the stiffness value was greater than maxratio, the DOF would fail the maxratio test. In these cases, MAXRATIO calculations can produce misleading or erroneous output and the analysis may terminate.

The MAXRATIO calculations have now been modified to take 2-by-2 pivoting into account. Thus, when 2-by-2 pivoting occurs, maxratio calculations are not made on the DOFs in the 2-by-2 pivot. If the maxratio vector is printed for this region it has an artificial value of 1.0 for these DOFs, meaning that they will never be listed for reasonable values of the MAXRATIO parameter. This produces MAXRATIO output, which is more a true measure of the solution that has actually taken place inside the sparse direct factorization.

For analyses such as inertia relief using the SUPORT entry selected by PARAM,INREL,-2, or the new large displacement rigid elements, or for solid elements using the interface spline elements, it should no longer be necessary to specify the BAILOUT parameter. The BAILOUT parameter is not recommended for production analysis, only for model debugging activities. It can mask models likely to produce low quality results.

When comparing results with those from a prior version, you may see fewer high ratio DOFs in the regions where 2-by-2 pivots are used. This is because the ratio messages were invalid for these DOFs on prior versions.

3.4 Performance Improvement in Modal Frequency Response for Large Frequency Ranges

Modal frequency response is a relatively inexpensive method for calculating response quantities. For large problems with large frequency ranges and thousand of modes, it may become very time consuming. A new technique for calculating frequency response quantities is activated by the Bulk Data parameter, PARAM,FASTFR.

When PARAM,FASTFR,YES is specified, the alternative technique is used. For large models with thousands of modes and a judicious use of structural damping, speedup compared to the conventional FRRD1 module is of the order 10-to-1.

CHAPTER
4

Elements

- Temperature-Dependent Composites Support Extended to Unsymmetric Laminates
- Global Ply Results Tracking
- GPFORCE and ESE Output for DMIG and GENEL
- Bar Element Torsional Mass Moment of Inertia
- PARAM,COUPMASS Lumped Mass Option
- QUADR Convergence Behavior
- Arbitrary Beam Cross Section (Pre-Release)

4.1 Temperature-Dependent Composites Support Extended to Unsymmetric Laminates

In MSC.Nastran 2004, laminated composites analysis was extended to include temperature-dependent ply materials in SOL 106 Nonlinear Analysis. This approach is based on updating the smeared laminate properties of symmetric laminates for the nonlinear QUAD4 and TRIA3 elements. The temperature-dependency of the ply materials was extended to include both orthotropic and anisotropic materials. Furthermore, the more accurate integral strain method was added to complement the default secant thermal strain method.

The symmetric laminate limitation in SOL 106 is now removed and membrane-bending coupling effects due to unsymmetric laminates are included in the nonlinear analysis.

Also, in this version, the temperature-dependent composites capability has been extended to the nonlinear QUADR and TRIAR composite elements. A nonsmeared approach is also available for the QUADR and TRIAR element types. This approach is a more general approach in that the laminate properties are not smeared as in the classical lamination theory, but evaluated during the element matrices calculation using an integration by layer. The benefit of the nonsmeared approach is that it will allow for future implementation of material nonlinear capabilities. Both the smeared and nonsmeared approaches are valid for symmetric and unsymmetric laminates. Non-uniform element grid point temperature and temperature gradient support by the QUADR/TRIAR is another difference compared to the constant element temperature QUAD4/TRIA3 element types.

Temperature-dependent QUAD4/TRIA3 composite models can be easily converted to equivalent QUADR/TRIAR models by adding the *NASTRAN QRMETH=5* command to the input file.

Note that the nonlinear QUADR/TRIAR element types are limited to temperature-dependent composites. Additional nonlinear analysis capability for these element types is planned for in future releases.

The following table summarizes the user interfaces for invoking the temperature-dependent composite capabilities for the composite element types:

Parameter	Value	QUAD4/TRIA3	QUADR/TRIAR
COMPMATT - Enable update of temperature-dependent composite properties.	Yes/Smear	Supported	Supported
	Nonsmear	Not Supported	Supported
	No (default)	Supported	Supported
EPSILONT - Select Integral or Secant thermal strain calculation method	Secant (default)	Supported	Supported
	Integral	Supported	Supported

Example:

```
ASSIGN PUNCH=OUTDIR:'shcntr2p.n',NEW,UNIT=7
$ id msc, shcntrl2.dat $ v2005 9-Jun-2004 hdp
SOL 106
TIME 600
$ Direct Text Input for Executive Control
CEND
SEALL = ALL
SUPER = ALL
TITLE = shape control demo (n2)
ECHO = NONE
MAXLINES = 999999999
$ Direct Text Input for Global Case Control Data
TEMPERATURE(INITIAL) = 1
SUBCASE 1
$ Subcase name : Default
   SUBTITLE=Default
   NLPARM = 1
   SPC = 2
   TEMPERATURE(LOAD) = 3
   DISPLACEMENT(SORT1,REAL,plot)=ALL
$ Direct Text Input for this Subcase
OUTPUT(XYOUT)
XYPUNCH DISP / 333(T3), 370(T3)
BEGIN BULK
PARAM    POST    -1
PARAM   COUPMASS 1
PARAM    LGDISP  1
PARAM    K6ROT   100.
PARAM,NOCOMPS,-1
PARAM   PRTMAXIM YES
PARAM,COMPMATT,YES
PARAM   NLTOL    0
NLPARM   1       10              ITER    1       100             YES
```

```
$ Direct Text Input for Bulk Data
$ Elements and Element Properties for region : smahcelem
$ Composite Property Record created from P3/PATRAN composite material
$ record : smahclam
$ Composite Material Description :
PCOMP    1                                       70.     0.
         1       .0045   45.     YES     2       .0045   0.      YES
         1       .0045  -45.     YES     1       .0045   90.     YES
         1       .0045   90.     YES     1       .0045   45.     YES
         2       .0045   0.      YES     1       .0045  -45.     YES
         1       .0045   90.     YES     1       .0045   0.      YES
         1       .0045   0.      YES     1       .0045   90.     YES
         1       .0045  -45.     YES     1       .0045   45.     YES
         1       .0045   90.     YES     1       .0045   90.     YES
         1       .0045  -45.     YES     1       .0045   45.     YES
$ Pset: "smahcelem" will be imported as: "pcomp.1"
CQUAD4   73      1       75      76      113     112
CQUAD4   74      1       76      77      114     113
CQUAD4   75      1       77      78      115     114
CQUAD4   76      1       78      79      116     115
CQUAD4   77      1       79      80      117     116
CQUAD4   78      1       80      81      118     117
…

…
GRID     662             8.      3.      0.
GRID     663             8.25    3.      0.
GRID     664             8.5     3.      0.
GRID     665             8.75    3.      0.
GRID     666             9.      3.      0.
$ Loads for Load Case : Default
SPCADD   2       1
$ Displacement Constraints of Load Set : cfff
SPC1     1       123456  1       38      75      112     149     186
         223     260     297     334     371     408     445     482
         519     556     593     630
$ Default Initial Temperature
TEMPD    1       70.     3       250.
$ Referenced Coordinate Frames
ENDDATA
```

PARAM,COMPMATT,YES invokes the temperature dependent composite capabilities for the composite QUAD4 elements in the model.

Absence of the EPSILONT parameter means that the default integral strain method of updating the smeared laminate properties is selected for this analysis.

This example, with the associated bulk data include files, can be found in the Test Problem Library – shcntrl2.dat, glepnast.dat, and nitnast.dat.

4.2 Global Ply Results Tracking

Idealization of large composite panels requires that each ply within a panel be accurately modeled with regard to stacking location and shape. When idealized in MSC.Nastran, this creates situations where adjacent elements may not contain the same number of plies, nor will these plies necessarily be continuous in terms of the internal ply numbering scheme. Therefore, the interpretation of results is a laborious task of manually identifying the consistent ply results when post-processing the output.

To address this limitation, user specification of global ply IDs has been introduced for easy reference of individual ply results across panels or sets of elements. A new composite property entry called PCOMPG is available for the specification of global ply IDs. The PCOMPG entry is an alternate property definition to the PCOMP entry. The global ID for each ply is included in all ply result tables.

Optionally ply-layer results can be sorted by global ply ID for a given element SET for easier results interpretation. A new Case Control command called GPRSORT is introduced to reference an element SET. Results are sorted by element ID or by global ply ID for a given element set.

This capability is available in all solution sequences that support composites except Design Optimization (SOL 200).

The following example illustrates the use of the PCOMPG Bulk Data entry to define global ply IDs.

The input file of this example, and others, can be found in the Test Problem Library – pcompg*.dat.

Shown in **Figure 4-1** is a laminated composite strip model with ply drop-off. A complete listing of this model (TPL testdeck: *pcompg1e.dat*) shown in **Listing 4-1**, illustrates specification of global ply IDs for each laminate on the PCOMPG entry. Also ply results sorted by global ply IDs are requested using the new GPRSORT Case Control command for element SET 100 (includes elements 100 and 200).

The standard ply results output is shown in **Listing 4-2** with the global ply IDs reported under the PLY ID label. **Listing 4-3** shows the ply results sorted by global ply IDs for the defined element SET 100.

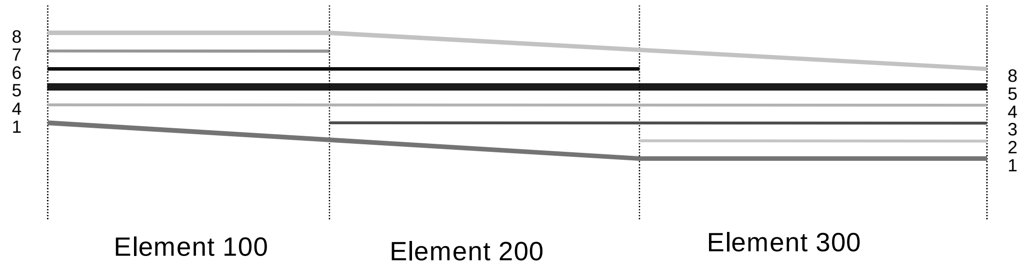


Figure 4-1 Laminated Composite Strip model with ply drop-offs

Listing 4-1 TPL testdeck pcompg1.dat

```
nastran system(361)=1
id msc, pcompg1e.dat $
time 60
sol 101
cend
title = Composite Strip with Global Ply IDs
disp =all
stress=all
set 100 = 100, 200
gprsort = 100
force=all
spcforces=all
spc=1
load=1000
begin bulk
param,k6rot,0.0
pcompg,1,,,5000.,hill,0.0,,,
,   1, 1, .0054, 45., yes
,   4, 1, .0054, 90., yes
,   5, 1, .0054, 90., yes
,   6, 1, .0054, 0.0, yes
,   7, 1, .0054,-45., yes
,   8, 1, .0054, 45., yes
pcompg,2,,,5000.,hill,0.0,,,
,   1, 1, .0054, 45., yes
,   3, 1, .0054, 0.0, yes
,   4, 1, .0054, 90., yes
,   5, 1, .0054, 90., yes
,   6, 1, .0054, 0.0, yes
,   8, 1, .0054, 45., yes
pcompg,3,,,5000.,hill,0.0,,,
```

```
,    1, 1, .0054, 45., yes
,    2, 1, .0054,-45., yes
,    3, 1, .0054, 0.0, yes
,    4, 1, .0054, 90., yes
,    5, 1, .0054, 90., yes
,    8, 1, .0054, 45., yes
$
mat8,1,2.0e7,2.0e6,.35,1.0e6,1.0e6,1.0e6,0.0,+
+,0.0,0.0,0.0,2.3e5, 1.95e5, 13000., 32000., 12000.
cquad4,100,1,1,2,6,5
cquad4,200,2,2,3,7,6
cquad4,300,3,3,4,8,7
force,1, 4,,1.,0.0,0.0,-1.0
force,1, 8,,1.,0.0,0.0,-1.0
spc1,1,12345,1,5
load,1000,0.1,1.0,1
grid, 1,, 0.0,0.0,0.0,,6
grid, 2,, 1.0,0.0,0.0,,6
grid, 3,, 2.0,0.0,0.0,,6
grid, 4,, 3.0,0.0,0.0,,6
grid, 5,, 0.0,1.0,0.0,,6
grid, 6,, 1.0,1.0,0.0,,6
grid, 7,, 2.0,1.0,0.0,,6
grid, 8,, 3.0,1.0,0.0,,6
param,autospc,yes
param,post,-1
enddata
```

Listing 4-2 Ply Results Output for Each Element in the Model

```
S T R E S S E S   I N   L A Y E R E D   C O M P O S I T E   E L E M E N T S   ( Q U A D 4 )
   ELEMENT      PLY STRESSES IN FIBER AND MATRIX DIRECTIONS    INTER-LAMINAR  STRESSES  PRINCIPAL STRESSES (ZERO SHEAR)      MAX
     ID          ID   NORMAL-1     NORMAL-2     SHEAR-12     SHEAR XZ-MAT  SHEAR YZ-MAT  ANGLE    MAJOR        MINOR        SHEAR
       100        1 -2.84204E+03 -1.02828E+03  7.92127E+02   -6.67973E+00 -3.12898E-01   69.43 -7.31044E+02 -3.13928E+03  1.20412E+03
       100        4  7.48335E+02 -8.80371E+02 -2.54907E+02   -7.95162E+00 -7.79520E-01   -8.69  7.87298E+02 -9.19334E+02  8.53316E+02
       100        5 -7.55171E+01 -4.35131E+02 -1.69516E+02   -8.54160E+00 -7.97555E-01  -21.66 -8.20804E+00 -5.02440E+02  2.47116E+02
       100        6  3.89044E+02 -7.66743E+01  8.41258E+01   -7.62246E+00 -7.54499E-01    9.93  4.03775E+02 -9.14045E+01  2.47590E+02
       100        7  1.72768E+03  2.22906E+02  3.51962E+02   -4.97689E+00 -4.53933E-01   12.53  1.80594E+03  1.44653E+02  8.30642E+02
       100        8  2.83923E+03  5.37783E+02 -6.37984E+02   -9.22442E-08  1.90999E-08  -14.50  3.00425E+03  3.72761E+02  1.31574E+03
       200        1  2.81691E+02 -2.90048E+02  3.28924E+02   -3.49256E+00 -3.51245E-01   24.50  4.31611E+02 -4.39967E+02  4.35789E+02
       200        3 -2.73870E+03  1.10669E+01  1.04747E+02   -9.62188E+00 -4.12887E-01   87.82  1.50513E+01 -2.74268E+03  1.37887E+03
       200        4  1.27601E+02 -8.43308E+01 -4.03806E+01   -9.82619E+00 -6.18361E-01  -10.43  1.35034E+02 -9.17641E+01  1.13399E+02
       200        5 -7.30090E+02  7.11099E+01  2.39860E+01   -9.62188E+00 -4.12887E-01   88.29  7.18274E+01 -7.30808E+02  4.01318E+02
       200        6  2.79715E+03 -6.88072E+01 -8.83526E+01   -3.49256E+00 -3.51245E-01   -1.76  2.79987E+03 -7.15284E+01  1.43570E+03
       200        8 -3.93539E+02  2.43778E+02 -3.62750E+02    3.93396E-16 -3.95636E-17  -65.65  4.07956E+02 -5.57717E+02  4.82836E+02
       300        1 -4.06549E+02 -1.07470E+02  1.66640E+02   -4.97689E+00 -5.17350E-02   65.95 -3.31102E+01 -4.80909E+02  2.23900E+02
       300        2 -3.15888E+02 -3.21002E+01 -9.81308E+01   -7.62246E+00 -8.59908E-02  -72.67 -1.47298E+00 -3.46515E+02  1.72521E+02
       300        3 -1.47324E+02  3.80228E+01 -1.87526E+01   -8.54160E+00 -9.08978E-02  -84.28  3.99010E+01 -1.49203E+02  9.45518E+01
       300        4  1.74069E+02  9.78514E+01  4.22159E+01   -7.95162E+00 -8.88424E-02   23.96  1.92833E+02  7.90880E+01  5.68725E+01
       300        5 -8.33154E+01  1.96665E+02  6.56791E+01   -6.67973E+00 -3.56612E-02   77.43  2.11307E+02 -9.79570E+01  1.54632E+02
       300        8  5.06968E+02  2.38382E+02 -1.75908E+02    9.22442E-08 -2.17683E-09  -26.32  5.93985E+02  1.51365E+02  2.21310E+02
```

Listing 4-3 Ply Results Sorted by Global Ply IDs

```
                              S T R E S S E S   I N   L A Y E R E D   C O M P O S I T E   E L E M E N T S
    GLOBAL  ELEMENT STRESSES IN FIBER AND MATRIX DIRECTIONS    INTER-LAMINAR  STRESSES  PRINCIPAL STRESSES (ZERO SHEAR)      MAX
    PLY ID       ID   NORMAL-1     NORMAL-2     SHEAR-12     SHEAR XZ-MAT  SHEAR YZ-MAT  ANGLE    MAJOR        MINOR        SHEAR
         1      100 -2.84204E+03 -1.02828E+03  7.92127E+02   -6.67973E+00 -3.12898E-01   69.43 -7.31044E+02 -3.13928E+03  1.20412E+03
                200  2.81691E+02 -2.90048E+02  3.28924E+02   -3.49256E+00 -3.51245E-01   24.50  4.31611E+02 -4.39967E+02  4.35789E+02
         3      200 -2.73870E+03  1.10669E+01  1.04747E+02   -9.62188E+00 -4.12887E-01   87.82  1.50513E+01 -2.74268E+03  1.37887E+03
         4      100  7.48335E+02 -8.80371E+02 -2.54907E+02   -7.95162E+00 -7.79520E-01   -8.69  7.87298E+02 -9.19334E+02  8.53316E+02
                200  1.27601E+02 -8.43308E+01 -4.03806E+01   -9.82619E+00 -6.18361E-01  -10.43  1.35034E+02 -9.17641E+01  1.13399E+02
         5      100 -7.55171E+01 -4.35131E+02 -1.69516E+02   -8.54160E+00 -7.97555E-01  -21.66 -8.20804E+00 -5.02440E+02  2.47116E+02
                200 -7.30090E+02  7.11099E+01  2.39860E+01   -9.62188E+00 -4.12887E-01   88.29  7.18274E+01 -7.30808E+02  4.01318E+02
         6      100  3.89044E+02 -7.66743E+01  8.41258E+01   -7.62246E+00 -7.54499E-01    9.93  4.03775E+02 -9.14045E+01  2.47590E+02
                200  2.79715E+03 -6.88072E+01 -8.83526E+01   -3.49256E+00 -3.51245E-01   -1.76  2.79987E+03 -7.15284E+01  1.43570E+03
         7      100  1.72768E+03  2.22906E+02  3.51962E+02   -4.97689E+00 -4.53933E-01   12.53  1.80594E+03  1.44653E+02  8.30642E+02
         8      100  2.83923E+03  5.37783E+02 -6.37984E+02   -9.22442E-08  1.90999E-08  -14.50  3.00425E+03  3.72761E+02  1.31574E+03
                200 -3.93539E+02  2.43778E+02 -3.62750E+02    3.93396E-16 -3.95636E-17  -65.65  4.07956E+02 -5.57717E+02  4.82836E+02
1    COMPOSITE STRIP WITH GLOBAL PLY IDS                                   JULY  29, 2004  MSC.NASTRAN  7/23/04   PAGE    19
      TEST GPRSORT PROCESSING
0

                  F A I L U R E   I N D I C E S   F O R   L A Y E R E D   C O M P O S I T E   E L E M E N T S
    GLOBAL  FAILURE    ELEMENT   FP=FAILURE INDEX FOR PLY    FB=FAILURE INDEX FOR BONDING   MAX OF FP,FB FOR ALL ELEMENTS  FLAG
    PLY ID   THEORY         ID  (DIRECT STRESSES/STRAINS)     (INTER-LAMINAR STRESSES)        REFERENCED BY GLOBAL PLY
         1   HILL          100            0.0055
                                                                 0.0013
             HILL          200            0.0008
                                                                 0.0007                        0.0055
         3   HILL          200            0.0003
                                                                 0.0019                        0.0019
         4   HILL          100            0.0012
                                                                 0.0016
             HILL          200            0.0000
                                                                 0.0020                        0.0020
         5   HILL          100            0.0004
                                                                 0.0017
             HILL          200            0.0000
                                                                 0.0019                        0.0019
         6   HILL          100            0.0001
                                                                 0.0015
             HILL          200            0.0002
                                                                 0.0007                        0.0015
         7   HILL          100            0.0012
                                                                 0.0010                        0.0012
         8   HILL          100            0.0047

             HILL          200            0.0013
```

4.3 GPFORCE and ESE Output for DMIG and GENEL

The effects of the GENEL element and user-supplied DMIG matrices in linear statics solution sequence (SOL 101) are now included in the Grid Point Force summations. The GPFORCE output includes individual GENEL element identifications and DMIG matrix names. Element strain energy for GENEL elements and DMIG matrices is also calculated.

The existing Case Control commands, ESE for element strain energy and GPFORCE for Grid Point Force, are used to request the output. Below is a sample F06 output illustrating the inclusion of the DMIG contribution to the grid point force summation.

```
  G R I D   P O I N T   F O R C E   B A L A N C E

    POINT-ID    ELEMENT-ID     SOURCE             T1             T2             T3             R1             R2             R3
          11                  F-OF-SPC       3.128563E+06  -9.800248E+05   1.989233E+01   4.673181E+03   5.798827E+03  -8.957655E+01
          11                  K11X          -3.128563E+06   9.800248E+05  -1.989233E+01  -4.673181E+03  -5.798827E+03   8.957655E+01
          11                  *TOTALS*       0.0           -2.328306E-10  -1.421085E-13  -3.637979E-12  -7.275958E-12   0.0
0         12            12    QUAD4         -2.193838E+06   1.597768E+05  -1.398442E+01  -1.416497E+03  -9.718338E+03  -1.905801E+00
          12                  K11X           2.193838E+06  -1.597768E+05   1.398442E+01   1.416497E+03   9.718338E+03   1.905801E+00
          12                  *TOTALS*      -9.313226E-10   4.074536E-10  -2.772893E-12   7.776180E-11   3.037712E-10  -6.550316E-14
0         13            12    QUAD4          9.914453E+05  -3.166309E+05  -5.283162E+00  -6.762946E+02   6.893261E+03  -6.008299E+01
          13            13    QUAD4         -9.914453E+05   3.166309E+05   5.283162E+00   6.762946E+02  -6.893261E+03   6.008299E+01
          13                  *TOTALS*       3.143214E-09   5.820766E-11   2.898037E-11  -3.363994E-10  -1.509761E-10  -3.552714E-14
0         14            13    QUAD4          1.665495E+06   1.313788E+06  -2.721329E+01  -1.409337E+03   4.590343E+03  -2.426860E+01
          14            14    QUAD4         -1.665495E+06  -1.313788E+06   2.721329E+01   1.409337E+03  -4.590343E+03   2.426860E+01
```

Since the DMIG matrix names are defined by character names rather than by enumerated values such as element identification, they currently cannot be individually selected by SET selection of the ESE Case Control command. The parameter, DMIGNRG is used to include the DMIG matrix energy to the Total Energy calculation and provide individual matrix contribution information. The default of DMIGNRG is NO. To include the DMIG contribution, the Bulk Data parameter, DMIGNRG should be set to YES.

Example files ab1/2/3/4.dat can be found in the TPL.

4.4 Bar Element Torsional Mass Moment of Inertia

In prior versions of MSC.Nastran, the BAR element did not include the torsional mass moment of inertia in the mass matrix. This led to different results when compared to the BEAM element for an equivalent structure. This limitation is documented as CSR7832.

The torsional mass moment of inertia can now be included in the BAR mass matrix by using 'NASTRAN SYSTEM(398)=1 or NASTRAN BARMASS=1'. Note that, by default, this term will not be included. For both values of PARAM COUPMASS, this term is added. If desired, the system cell default value can be changed via the NASTRAN rc file.

The BAR torsional mass moment of inertia term for the component of rotation about the element axis is calculated similar to the BEAM element using the following equation:

$$I_{xx} = \rho L(I_1 + I_2)$$

where:

I_{xx} = Torsional Mass Moment of Inertia

ρ = Density

L = Element of Length

I_1 and I_2 = Area Moments of Inertia

For COUPMASS=1, the axial mass will be consistent rather than coupled.

An example test file can be found in the TPL - brbm.dat.

4.5 PARAM,COUPMASS Lumped Mass Option

The *MSC.Nastran Quick Reference Guide* documentation for PARAM,COUPMASS suggests that its default value of –1 causes the generation of lumped mass matrices that contain only translational components for the elements listed therein. Notable exceptions to this are the CBAR and CBEAM elements, both of which will yield rotational and coupling terms in order to preserve the mass center when element offsets are defined. This offset mass is ‘lumped’ in the sense that it has low matrix rank, and is ‘coupled’ in the sense that there are non-zero off diagonal terms in the mass matrix. The CBEAM element will also yield a mass moment of inertia about the local X axis of the element, and if system cell 398>0, then this is also true of the CBAR element.

In order to yield a lumped mass matrix containing translational components only for the CBAR and CBEAM elements, system cell 414 has been introduced. The default value of this system cell (0) leaves the current behavior unchanged whereas a positive integer value for system cell 414, along with the default value for PARAM,COUPMASS (-1), will yield lumped mass matrices containing only translational components for both CBAR and CBEAM elements.

Inputs

FMS SECTION

```
NASTRAN SYSTEM(414) = 1
```

BULK DATA

```
PARAM,COUPMASS,–1
```

Note: This parameter may optionally be specified in the Case Control Section.

Outputs

The resulting mass matrix for the BAR and BEAM element will only contain translation terms. All off-diagonal and rotation terms will be zero.

Example:

The example file mass_bs.dat can be found in the Test Problem Library.

```
NASTRAN NLINES=99999
NASTRAN system(414) = 1
id msc, mass_bs.dat $ v2005 hdp 15-Jan-2004
$
```

```
SOL   101
DIAG  8
COMPILE SEMG $
ALTER 'ENDIF $ NOMGG>=0' $
MESSAGE//' MASS MATRIX' $
MATGPRBGPDTS,USET0,,MJJX//'G' $ MGG
EXIT $
CEND
TITLE = MASS MATRIX FOR BAR/BEAM ELEMENTS, COUPMASS = -1
BEGIN BULK
PBAR  101   100   .1    .01   .01   .02   10.
PBEAM 102   100   .1    .01   .01         .02   10.
MAT1  100   1.+7        .3    10.
PARAM COUPMASS-1
$
GRID  11          -1.   -.1   +.1
GRID  12          +1.   +.1   -.1
CBAR  1     101   11    12          1.                +A
+A                      .1                .1
$
GRID  21          -1.   -.1   +.1
GRID  22          +1.   +.1   -.1
CBAR  2     101   21    22          1.                +B
+B          2356
$
GRID  31          -1.   -.1   +.1
GRID  32          +1.   +.1   -.1
CBEAM 3     102   31    32          1.                +C
+C                      .1                .1
$
GRID  41          -1.   -.1   +.1
GRID  42          +1.   +.1   -.1
CBEAM 4     102   41    42          1.                +D
+D          2356
$
GRID  51          -1.   -.1   +.1
GRID  52          +1.   +.1   -.1
SPOINT53    54
CBEAM 5     102   51    52          1.                +E
+E          2356                                      +F
+F    53    54
ENDDATA
```

The default mass matrix is shown:

```
0                MJJX
   POINT      VALUE   POINT      VALUE   POINT      VALUE   POINT

 COLUMN           1 (          11-T1).
         11 T1  1.11095E+01        11 R3 -1.11095E+00
 COLUMN           2 (          11-T2).
         11 T2  1.11095E+01
 COLUMN           3 (          11-T3).
         11 T3  1.11095E+01        11 R1  1.11095E+00
 COLUMN           4 (          11-R1).
         11 T3  1.11095E+00        11 R1  1.11095E-01

 COLUMNS           5 (          11-R2) THRU           5 (          11-R2) ARE NULL.
 COLUMN           6 (          11-R3).
         11 T1 -1.11095E+00        11 R3  1.11095E-01
 COLUMN           7 (          12-T1).
         12 T1  1.11095E+01        12 R3 -1.11095E+00
 COLUMN           8 (          12-T2).
         12 T2  1.11095E+01
 COLUMN           9 (          12-T3).
         12 T3  1.11095E+01        12 R1  1.11095E+00
 COLUMN          10 (          12-R1).
         12 T3  1.11095E+00        12 R1  1.11095E-01
```

With System Cell 414 set to 1, the mass matrix contains only diagonal terms:

```
0                MJJX
   POINT      VALUE   POINT      VALUE   POINT      VALUE   POINT

 COLUMN           1 (          11-T1).
         11 T1  1.11095E+01
 COLUMN           2 (          11-T2).
         11 T2  1.11095E+01
 COLUMN           3 (          11-T3).
         11 T3  1.11095E+01

 COLUMNS           4 (          11-R1) THRU           6 (          11-R3) ARE NULL.
 COLUMN           7 (          12-T1).
         12 T1  1.11095E+01
 COLUMN           8 (          12-T2).
         12 T2  1.11095E+01
 COLUMN           9 (          12-T3).
         12 T3  1.11095E+01
 COLUMNS          10 (          12-R1) THRU          12 (          12-R3) ARE NULL.
```

4.6 QUADR Convergence Behavior

Introduction

A new QUADR element has been introduced in MSC.Nastran 2005 to provide more accurate results for coarsely meshed regions where accuracy would tend to degrade. In recent tests, the new QUADR element was compared to the QUAD4 element. The results of the testing showed that the new QUADR element produced more accurate results, even in regions that were coarsely meshed.

Benefits

Overall, an increase in accuracy can be expected when using the enhanced QUADR element, and improvements in accuracy carry over to areas of the model that are more coarsely meshed.

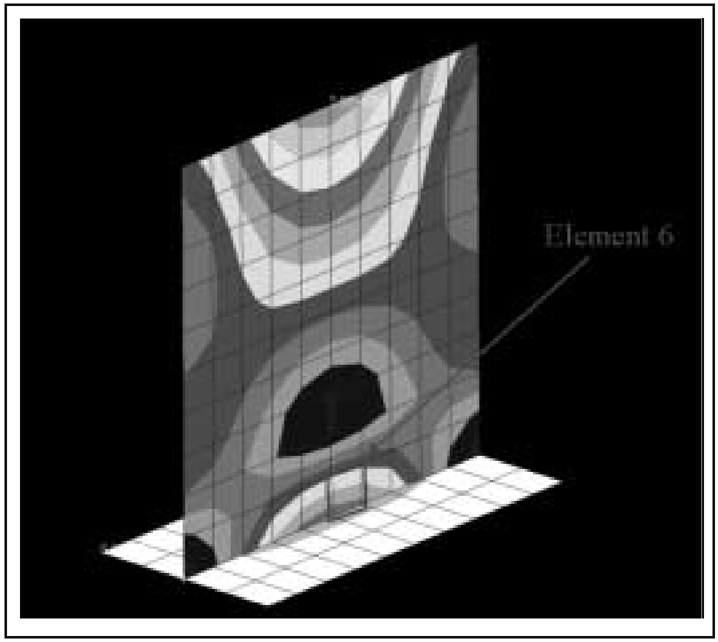
Existing QUAD4 element models can easily be converted to QUADR elements by setting a single System Cell (QRMETH) in the NASTRAN Statement.

Inputs

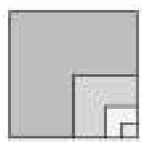
System cell QRMETH=5 will convert all QUAD4/TRIA3 elements in the model to QUADR/TRIAR.

Example

To demonstrate the difference in accuracy between the QUADR and QUAD4 elements, various mesh densities for a simple “T-Section” test model were run using MSC.Nastran 2004 r3, comparing von Mises stress results taken in the central position at Element 6, as shown.

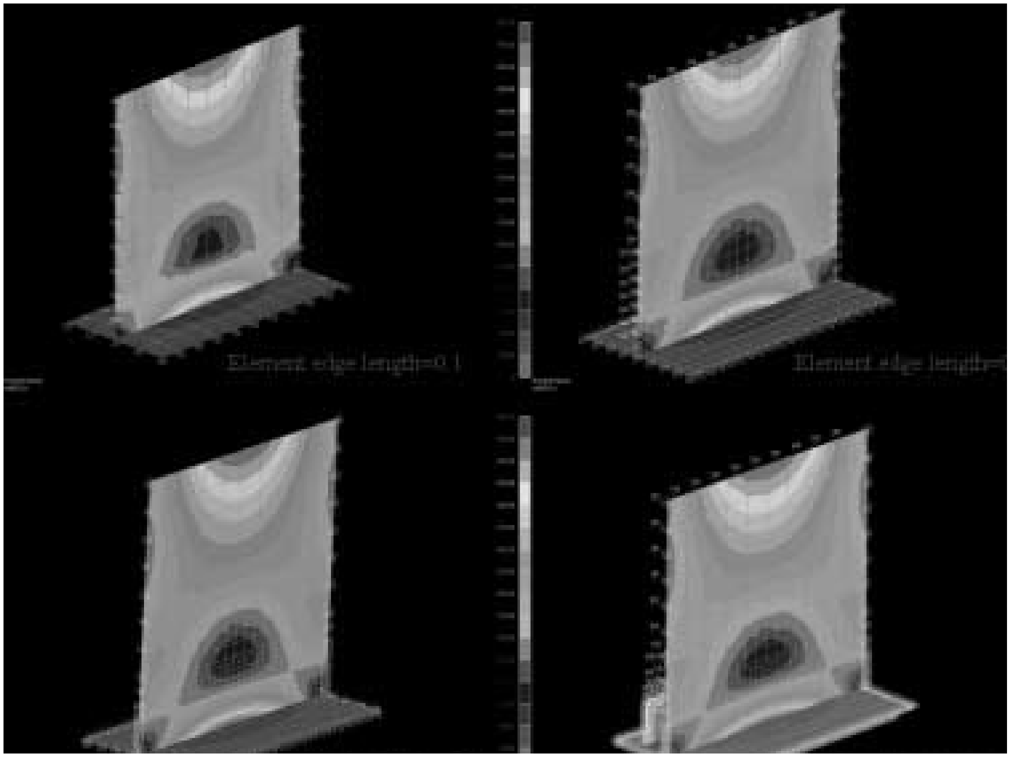


Four analysis runs were made using for element edge lengths (0.1, 0.05, 0.025, and 0.0125) to vary the number of elements (increasing the mesh density).

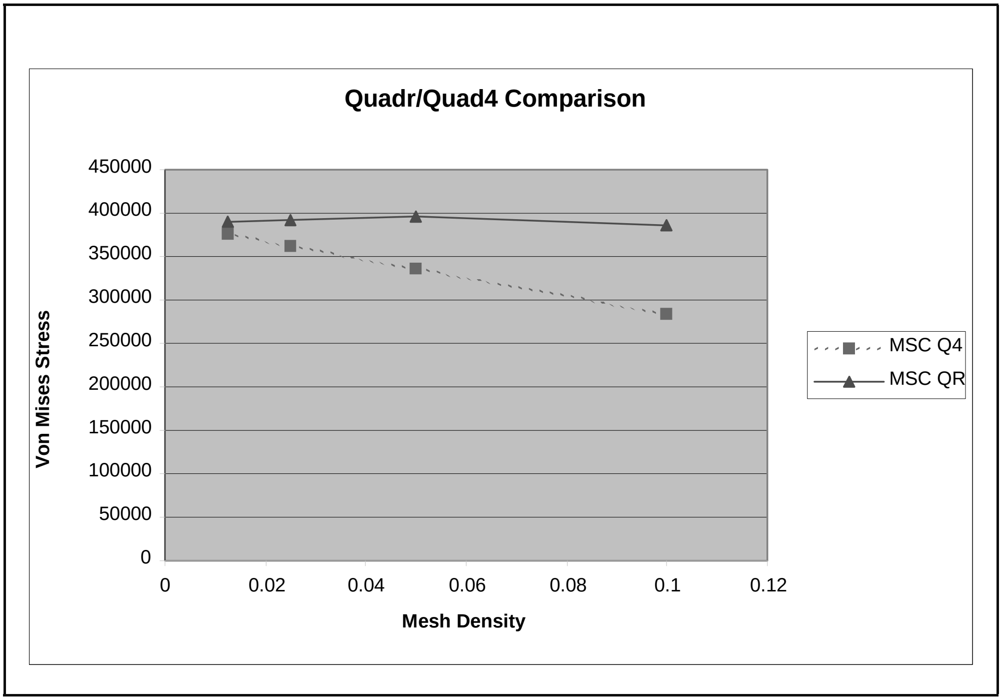


Model Resolution	Normalized DOF	Normalized Run time
0.1	1	1
0.05	4	16
0.025	16	256
0.0125	64	4096

The four element stress contour plots, below, show a very similar stress distribution for all four mesh densities; however, the actual stress values do vary slightly between the plots.



The results for each of the four MSC.Nastran runs were tabulated and plotted to compare the QUAD4 and QUADR elements.



It is evident that the MSC.Nastran QUADR element performs more consistently and accurately for each of the four element mesh densities (coarse to fine) analyzed, compared to the QUAD4 element that is less accurate for the models with larger element edge lengths and fewer elements.

4.7 Arbitrary Beam Cross Section (Pre-Release)

Introduction

Beam elements have long been a staple in MSC.Nastran. Over the years, the capability of beam element has grown steadily from constant cross section of PBAR to variable cross section of PBEAM. However, users are required to compute the sectional properties in order to utilize BAR and / or BEAM elements in the analysis. To alleviate the amount of effort from engineers, PBARL and PBEAML were added for popular cross sectional profiles. Nevertheless, engineers are still left to search for modeling alternatives for 1-D structural components with arbitrary cross sectional shapes. A new user interface for describing cross section shapes for CBAR and CBEAM element types has been developed, and will be provided in the release of MSC.Nastran 2005 r2, and is currently available for beta testing.

Development of this new capability in MSC.Nastran has been driven by the automotive industry, keen to be able to easily represent the nonstandard beam profiles commonly used in automotive design, and to use analysis tools to optimize the profile designs themselves.

Subsequent development phases are planned, which will add more advanced features to the Arbitrary Beam Section capability.

Benefits

The new user interface for describing cross section shapes of CBAR and CBEAM element types will provide users with the ability to:

- More easily model 1-D structural components with arbitrary cross sectional profiles using the MSC.Nastran BAR and / or BEAM element types for analysis in linear solution sequences
- Design an optimal cross section profile in the Design Optimization solution sequence, SOL 200 to optimize the overall model performance

This new capability, the Arbitrary Beam Section, will be delivered in the MSC.Nastran 2005 r2, but is available for beta testing in MSC.Nastran 2005 r1.

Inputs and Outputs

Essentially, the shape of the beam cross section is defined using sets of POINTs as defined on the SET1, or new SET3 Bulk Data entry (subsequent development phases will allow section definition using geometric entities - GMCURV). These sets are then referenced by new Bulk Data entries - PBRSECT for the BAR, PBMSECT for the BEAM

- used to define the cross section form parameters, and reference material properties. The types of section that can be defined include a General Section, Open Profile, and Closed Profile, with various parameters required on the PBRSECT or PBMSECT entries to define outer perimeter, inner perimeter, and branch segments where applicable.

Currently for the BEAM element, only a constant cross section beam is supported.

Once all of the bulk data has been read in, equivalent BAR and BEAM elements are created from the data supplied by the PBRSECT and PBMSECT entries. These equivalent element definitions are printed out to the .f06 output file.

Guidelines

1. BRP for CP and OP must start or end branching from OUTP. BRP must not start or end from another BRP.
2. BRP must not branch out from the end of OUTP. This rule covers both CP and OP.
3. For CP and OP, a $T = rs$, where rs denotes a positive real single precision number, must be present even if the thickness for every segment is separately defined. This thickness will be used for all segments which do not have specific thickness defined for them.
4. When PT=(id1,id2) is utilized to define the thickness of a segment, the id1 and id2 must be next to each other on the SET1 or SET3. A warning message will be issued if this guideline is not observed.

For a design optimization analysis, the PBRSECT and PBMSECT entries are referenced by the design variable property relation entries, DVPREL1. Dimensions that can be taken into the design optimization analysis include:

- Overall Width - input W for PNAME field of DVPREL1. This is available for GS, CP and OP. Overall width is computed as $X1_{max} - X1_{min}$. Both $X1_{max}$ and $X1_{min}$ are collected by examining $X1$ of all POINT entries involved.
- Overall Height - input H for PNAME field of DVPREL1. Also available for GS, CP and OP. Overall height is computed as $X2_{max} - X2_{min}$. Both $X2_{max}$ and $X2_{min}$ are collected by examining $X2$ of all POINT entries involved.
- Segment Thickness - input T or T(id) for PNAME field of DVPREL1. This is available only for CP and OP.

New PBRSECT, PBMSECT, and POINT entries are generated after each design cycle.

The stress recovery points, C, D, E, and F, are automatically selected by internal logic that will pick POINTs with extreme coordinates, that is, closest to the four corners of the rectangle defined by the overall width and height that encloses the cross section. If a POINT is on a section defined as a design variable in a design optimization analysis, then the POINT will move as the design variable changes. However, the location of the POINT itself cannot be defined as a design variable.

Example: Z-Section Beam

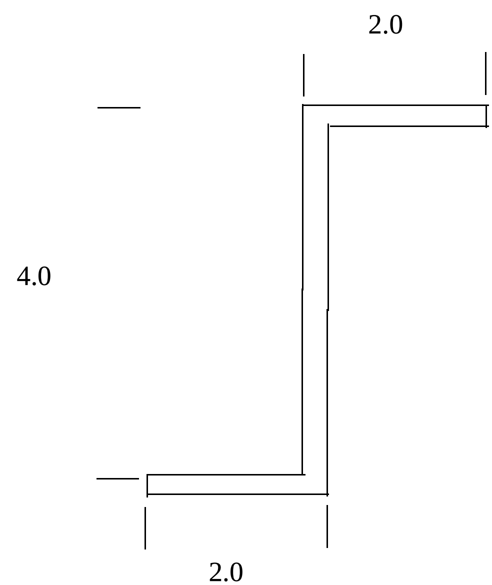


Figure 4-2 Z-Section - Uniform Thickness of 0.1

The required bulk data entries to define the above section for linear analysis is as follows:

1	2	3	4	5	6	7	8	9	10
POINT	1		0.0	0.0					
POINT	2		2.0	0.0					
POINT	3		2.0	3.9					
POINT	4		3.9	3.9					
POINT	5		3.9	4.0					
POINT	6		1.9	4.0					
POINT	7		1.9	0.1					

1	2	3	4	5	6	7	8	9	10
POINT	8		0.0	0.1					
$SET3	SID	DES	ID1	ID2	ID3				
SET3	10	POINT	1	THRU	8				

where DES (description) can be POINT, GRID, or ELEMENT.

$PBRSECT	PID	MID	FORM						
PBRSECT	1	1	GS						
	OUTP=10								

where FORM can be: GS - General Section
OP - Open Profile
CP - Closed Profile

$PBMSECT	PID	MID	FORM						
PBMSECT	2	1	GS						
	OUTP=10								

PBMSECT,2 defines a constant section beam.

The z-section example showing the OP option with thickness definition is as follows:

1	2	3	4	5	6	7	8	9	10
POINT	11		0.0	0.05					
POINT	12		1.95	0.05					
POINT	13		1.95	3.95					
POINT	14		3.9	3.95					
SET3	20	POINT	11	THRU	14				

PBRSECT	11	1	OP						
	OUTP=20, T=0.1, T(2)=[0.1, PT=(12,13)]								

PBMSECT	12	1	OP						
	OUTP=20, T=0.1								

Further examples are available in the Test Problem Library - zbr3.dat, zbr4.dat zbr5.dat, zbm3.dat, zbm4.dat, zbm5.dat.

CHAPTER

5

Dynamic Analysis

- C-Set Improvements
- Enhancements to the MODESELECT Case Control Command
- Automatic Q-Set (AUTOQSET)
- Enhancements to Dynamic Excitation Processing in DPD Module
- Enhancements to Transient Response Analysis

5.1 C-Set Improvements

When performing modal synthesis with free or mixed boundary conditions, the c-set mass is usually included in the calculation of the component modes. A new parameter, ZROCMAS, has been introduced to allow the c-set masses to be set to zero when computing component modes. For the case where the component has large masses on the c-set degrees of freedom, or when the user requests too many modes for the component, the c-set residual flexibility will become singular, causing the component reduction to fail. Setting the parameter ZROCMAS to YES will avoid this condition by excluding the c-set masses when calculating the component modes.

5.2 Enhancements to the MODESELECT Case Control Command

The MODESELECT Case Control command, which was first made available in MSC.Nastran 2004, has been greatly enhanced with the addition of several new options. The enhanced command permits the user to specify ALL data related to mode selection without the need for any parameters. The command, which can be employed for selecting either structure modes or fluid modes, offers five different and distinct options.

The details and usage of the enhanced command are clearly described in the *MSC.Nastran Quick Reference Guide*. A short description of the various options available with this command is listed below.

1. Mode selection based on arbitrary mode numbers

   This option is the same as the one (and only one) that was available in MSC.Nastran 2004.

2. Mode selection based on the number of lowest modes

   This option is similar to the usage of the LMODES / LMODESFL parameter.

3. Mode selection based on range of mode numbers

   This option can be regarded as a variation of options (1) and (2) above.

4. Mode selection based on frequency range

   This option is similar to the usage of the LFREQ / LFREQFL and HFREQ / HFREQFL parameters. However, this option is more general since it also allows for the UNCONDITIONAL inclusion or exclusion of selected modes regardless of their frequencies.

5. Mode selection based on modal effective mass fraction (MEFFMFRA) criteria

   This powerful option is the highlight of the enhancement. It allows the user to select modes based on different MEFFMFRA criteria. Further, like Option (4) above, it also allows for the UNCONDITIONAL inclusion or exclusion of selected modes regardless of their MEFFMFRA values.

There are several example problems included in the Test Problem Library that illustrate the use of MODESELECT.

For SOL 111: kmfmodea / b / c / d / s.dat

For SOL 112: kmtmodea / b / c / d / s.dat

```
id msc, kmtmoded.dat
$ID TEST,MODESELECT $ TEST PROBLEM KMTMODED
$
SOL 112
TIME 30
CEND
TITLE = TEST PROBLEM KMTMODED - MODESELECT IN MODAL TRAN. RESP. ANALYSIS
SUBTITLE = TEST OF MODESELECT WITH THE MODAL EFF. MASS FRACTION OPTION
$
MODESELECT(T1FR T2FR=0.9 T3FR=0.8)
MODALSE(SORT1, PRINT, ESORT=ASCEND, THRESH=0.0, FREQ=ALL)= all
MODALKE(SORT2, PRINT, ESORT=DESCEND, THRESH=0.0, FREQ=ALL)= all
$
SUBCASE 1
disp(plot) = all
TSTEP= 100
DLOAD = 10
METHOD = 10
$
BEGIN BULK
$
$ DYNAMIC LOADING
$
freq1,5,1.,1.,20
tload2, 10, 10, , load, 0.0, 100.0, 10.0
$
tstep, 100, 200, .005, 20
$
DAREA,10,13,3,-1.
EIGRL,10,,1000.
$
$  BASIC MODEL DEFINITION
$
GRDSET,,,,,,,6
GRID,1,,-.4,0.,0.,,123456
GRID,3,,-.4,0.9,0.
=,*2,=,=,*.9,==
=1
...
...
$
$  ELEMENTS
$
CQUAD4,1,1,1,2,4,3
=,*1,=,*2,*2,*2,*2
=1
CQUAD4,4,1,7,8,14,13
CQUAD4,6,1,9,10,20,19
=,*1,=,*1,*1,*1,*1
=2
...
...
MAT1,1,30.+6,,.3,.283
PARAM,WTMASS,.00259
PSHELL,1,1,.05,1,,1
$
ENDDATA
```

5.3 Automatic Q-Set (AUTOQSET)

Component modes (or dynamic reduction) are computed if the following items are defined in the input file:

1. Mass is present
2. EIGR or EIGRL Bulk Data entry is requested by METHOD command (or PARAM,METHCMRS)
3. Generalized coordinates (q-set degrees-of-freedom) are defined

The q-set DOFs are defined on QSETi entries (SEQSETi for superelements) and associated SPOINT or GRID entries. It is the user's responsibility to define a sufficient number of q-set DOFs to capture all of the eigenvectors in the desired frequency range defined on the EIGR or EIGRL entry and residual vectors. If too few q-set DOFs are defined then modal truncation occurs and accuracy may suffer. If too many then the dynamically reduced matrices will have null columns for the unused q-set DOFs and may result in a performance degradation.

In MSC.Nastran 2005, the user may replace all q-set related Bulk Data entries with the user parameter PARAM,AUTOQSET,YES. The number of component modes computed is determined by the frequency range and/or number of desired engenvectors specified on the selected EIGR or EIGRL Bulk Data entry.

Since the generalized coordinates are automatically defined, the following entries may not be specified: QSETi, SEQSETi, SENQSET, or PARAM,NQSET. Also, those GRID and/or SPOINT entries used to define the q-set may be left in the Bulk Data section but it is recommended that they be removed.

In superelement analysis, the calculation of component modes is attempted on all superelements including the residual structure. Also, all generalized coordinates for all superelements will become interior to the residual structure and also assigned to the q-set in the residual structure. In other words, component modes may not be assigned interior to a superelement and they may not be removed (constrained).

This feature is currently not supported with:

1. Multiple boundary conditions
2. Design optimization (SOL 200)
3. Aerodynamic analyses (SOLs 144, 145, 146)
4. Cyclic symmetry analyses (SOLs 114, 115, 116, 118)
5. Restarts

Example

In the following example the user defines six q-set DOFs for natural frequencies up to 1200 cycles per unit time.

```
SOL 103
DIAG 8,15
CEND
TITLE= AUTOQSET DEMONSTRATION PROBLEM
SUBTITLE= TWENTY CELL BEAM
SPC=1002
METHOD=1
BEGIN BULK
EIGRL 1           1200.
QSET1   0       101     THRU    106
SPOINT  101     THRU    106
GRID    10000           0.0     0.0     0.0             1246
=       *(1)    =       *(5.)   == $
=(19)
CBAR    101     100     10000   10001   0.0     0.0     1.      1
=       *(1)    =       *(1)    *(1)    == $
=(18)
PBAR    100     1000    0.31416 0.15708
MAT1    1000    3.+7            .3      7.764-4
SPC     1002    10020   3               10000   3
ENDDATA
```

The results of the f06 show that six q-set are insufficient to capture the residual vectors as shown by the messages below:

```
 R E A L   E I G E N V A L U E S
                                         (BEFORE AUGMENTATION OF RESIDUAL VECTORS)
   MODE    EXTRACTION      EIGENVALUE            RADIANS             CYCLES            GENERALIZED         GENERALIZED
    NO.       ORDER                                                                       MASS              STIFFNESS
        1         1        1.881936E+04        1.371837E+02        2.183346E+01        1.000000E+00        1.881936E+04
        2         2        3.011058E+05        5.487311E+02        8.733327E+01        1.000000E+00        3.011058E+05
        3         3        1.524259E+06        1.234609E+03        1.964941E+02        1.000000E+00        1.524259E+06
        4         4        4.816616E+06        2.194679E+03        3.492940E+02        1.000000E+00        4.816616E+06
        5         5        1.175494E+07        3.428547E+03        5.456702E+02        1.000000E+00        1.175494E+07
        6         6        2.435711E+07        4.935292E+03        7.854762E+02        1.000000E+00        2.435711E+07
        7         7        4.506449E+07        6.713009E+03        1.068409E+03        1.000000E+00        4.506449E+07
```

```
 ^^^
 ^^^ USER   WARNING MESSAGE 9144 (SEMR4)
 ^^^ THERE ARE NO Q-SET DEGREES-OF-FREEDOM LEFT TO ACCOMMODATE ANY   RESIDUAL VECTORS.
 ^^^ USER INFORMATION:   NO RESIDUAL VECTORS WILL BE COMPUTED.
 ^^^ USER ACTION:    SPECIFY AT LEAST            6 MORE Q-SET DEGREES-OF-FREEDOM.
 ^^^
 ^^^
 ^^^ USER   WARNING MESSAGE 9145 (  RESLOAD )
 ^^^ THERE ARE NOT ENOUGH Q-SET DEGREES-OF-FREEDOM DEFINED TO    ACCOMMODATE ALL OF THE COMPUTED EIGENVECTORS AND/OR
RESIDUAL
     VECTORS.
 ^^^ USER INFORMATION:   THE LAST            1 MODE(S) ABOVE WILL BE TRUNCATED.
 ^^^ USER ACTION:    SPECIFY AT LEAST            1 MORE Q-SET DEGREES-OF-FREEDOM.
```

```
R E A L   E I G E N V A L U E S
                                        (BEFORE AUGMENTATION OF RESIDUAL VECTORS)
  MODE    EXTRACTION      EIGENVALUE            RADIANS             CYCLES            GENERALIZED         GENERALIZED
   NO.       ORDER                                                                       MASS              STIFFNESS
       1         1        1.881936E+04        1.371837E+02        2.183346E+01        1.000000E+00        1.881936E+04
       2         2        3.011058E+05        5.487311E+02        8.733327E+01        1.000000E+00        3.011058E+05
       3         3        1.524259E+06        1.234609E+03        1.964941E+02        1.000000E+00        1.524259E+06
       4         4        4.816616E+06        2.194679E+03        3.492940E+02        1.000000E+00        4.816616E+06
       5         5        1.175494E+07        3.428547E+03        5.456702E+02        1.000000E+00        1.175494E+07
       6         6        2.435711E+07        4.935292E+03        7.854762E+02        1.000000E+00        2.435711E+07

^^^ USER   WARNING MESSAGE 9144 (SEMR4)
^^^ THERE ARE NO Q-SET DEGREES-OF-FREEDOM LEFT TO ACCOMMODATE ANY   RESIDUAL VECTORS.
^^^ USER INFORMATION:   NO RESIDUAL VECTORS WILL BE COMPUTED.
^^^ USER ACTION:    SPECIFY AT LEAST            6 MORE Q-SET DEGREES-OF-FREEDOM.
```

If we replace the QSETi and SPOINT entries with PARAM,AUTOQSET,YES:

```
SOL 103
DIAG 8,15
CEND
TITLE= AUTOQSET DEMONSTRATION PROBLEM
SUBTITLE= TWENTY CELL BEAM
SPC=1002
METHOD=1
BEGIN BULK
EIGRL 1           1200.
PARAM,AUTOQSET,YES
GRID    10000           0.0     0.0     0.0             1246
=       *(1)    =       *(5.)   == $
=(19)
CBAR    101     100     10000   10001   0.0     0.0     1.      1
=       *(1)    =       *(1)    *(1)    == $
=(18)
PBAR    100     1000    0.31416 0.15708
MAT1    1000    3.+7            .3      7.764-4
SPC     1002    10020   3               10000   3
ENDDATA
```

Then the results show that all of the eigenvectors and the residual vectors are now computed.

```
R E A L   E I G E N V A L U E S
                                        (BEFORE AUGMENTATION OF RESIDUAL VECTORS)
  MODE    EXTRACTION      EIGENVALUE            RADIANS             CYCLES            GENERALIZED         GENERALIZED
   NO.       ORDER                                                                       MASS              STIFFNESS
       1         1        1.881936E+04        1.371837E+02        2.183346E+01        1.000000E+00        1.881936E+04
       2         2        3.011058E+05        5.487311E+02        8.733327E+01        1.000000E+00        3.011058E+05
       3         3        1.524259E+06        1.234609E+03        1.964941E+02        1.000000E+00        1.524259E+06
       4         4        4.816616E+06        2.194679E+03        3.492940E+02        1.000000E+00        4.816616E+06
       5         5        1.175494E+07        3.428547E+03        5.456702E+02        1.000000E+00        1.175494E+07
       6         6        2.435711E+07        4.935292E+03        7.854762E+02        1.000000E+00        2.435711E+07
       7         7        4.506449E+07        6.713009E+03        1.068409E+03        1.000000E+00        4.506449E+07
```

R E A L   E I G E N V A L U E S

(AFTER AUGMENTATION OF RESIDUAL VECTORS)

MODE NO.	EXTRACTION ORDER	EIGENVALUE	RADIANS	CYCLES	GENERALIZED MASS	GENERALIZED STIFFNESS
1	1	1.881936E+04	1.371837E+02	2.183346E+01	1.000000E+00	1.881936E+04
2	2	3.011059E+05	5.487311E+02	8.733327E+01	1.000000E+00	3.011059E+05
3	3	1.524259E+06	1.234609E+03	1.964941E+02	1.000000E+00	1.524259E+06
4	4	4.816615E+06	2.194679E+03	3.492940E+02	1.000000E+00	4.816615E+06
5	5	1.175493E+07	3.428547E+03	5.456702E+02	1.000000E+00	1.175493E+07
6	6	2.435711E+07	4.935292E+03	7.854761E+02	1.000000E+00	2.435711E+07
7	7	4.506449E+07	6.713009E+03	1.068409E+03	1.000000E+00	4.506449E+07
8	8	9.003539E+07	9.488698E+03	1.510173E+03	1.000000E+00	9.003539E+07
9	9	1.442988E+08	1.201244E+04	1.911840E+03	1.000000E+00	1.442988E+08

5.4 Enhancements to Dynamic Excitation Processing in DPD Module

The DPD module generates dynamic excitations (applied loads and enforced motion information) for subsequent use in frequency response and transient response calculations. Prior to MSC.Nastran 2005, these excitations were generated in single precision and were stored in the Dynamic Loads Table (DLT). Starting with MSC.Nastran 2005, these excitations are generated in machine precision. Further, these excitations are no longer stored in the DLT, but are instead generated and held as machine precision matrices.

There is another important enhancement in the DPD module. Depending upon the time delays and phase angles specified, the columns of the excitation matrices are segregated appropriately so that each column represents a dynamic excitation with its own unique combination of time delay and phase angle. This scheme allows for the residual vectors computed in modal dynamic analysis to be more meaningful and more representative of the dynamic excitation employed in the analysis since it accounts for the time delays and phase angles associated with the dynamic excitations.

The enhancements mentioned above not only result in increased accuracy for dynamic response calculations on double precision machines, but also facilitate more efficient processing of these excitations in subsequent modules.

5.5 Enhancements to Transient Response Analysis

Increased Accuracy from TRLG Module

The TRLG module generates time dependent dynamic excitations (applied loads and enforced motion data) for subsequent use in transient response calculations. Prior to MSC.Nastran 2005, these excitations were generated in single precision. Starting with MSC.Nastran 2005, these excitations are generated in machine precision, thereby resulting in increased accuracy for transient response calculations on double precision machines.

Improved Calculations for Enforced Motion in TRLG and TRD1 Modules

The differentiation scheme employed in the TRLG module for enforced displacement and enforced velocity applications in transient analysis has been improved in MSC.Nastran 2005 by employing the same central finite difference scheme that is used in the subsequent TRD1 module which performs the solution phase for linear transient analysis (see Chapter 6 of the *MSC.Nastran Basic Dynamic Analysis User's Guide*). Appropriate enhancements have also been made to the TRD1 module to account for these TRLG changes. These improvements impact enforced motion usage in transient analysis not only for the SPC/SPCD approach, but also the large mass approach. For the enforced displacement and enforced velocity applications in transient analysis, this improved scheme yields much better correlation between the results obtained from these two approaches than existed in earlier versions. It also results in improved correlation with the corresponding results obtained from the Lagrange multiplier technique (LMT).

Initial Condition Specification for Enforced Motion Usage via SPC/SPCD

Enforced acceleration or enforced velocity usage in transient analysis via SPC/SPCD specification requires integration to compute the corresponding enforced velocities and/or displacements. This integration involves the use of initial conditions. Prior to MSC.Nastran 2005, there was no way for the user to specify the initial conditions for the enforced degrees of freedom (DOFs) in these cases. Starting with MSC.Nastran 2005, this facility is now available. With this feature, the user can specify initial displacements for enforced DOFs in the case of enforced velocity usage via SPC/SPCD and can specify initial displacements as well as initial velocities for enforced DOFs in the case of enforced acceleration usage via SPC/SPCD. The initial displacement and velocity values are specified via corresponding factors in two new fields that have been added to the TLOAD1 and TLOAD2 Bulk Data entries. Details will be clear from the description of these expanded entries in the ***MSC.Nastran Quick Reference Guide***. This enhancement will greatly help users in performing enforced motion studies with a variety of scenarios.

It should be noted here that the initial conditions for the enforced DOFs mentioned here are distinct from, and may be used in conjunction with, the initial conditions for independent DOFs that may be specified by a TIC Bulk Data entry.

CHAPTER

6 Optimization

- Composite Ply Strength Ratio Response Type for the DRESP1 Entry
- New FUNC(tions) for the DRESP2 Entry
- Transformation of Approximate Optimization Task to a Feasible Design
- Residual Vectors Based on Adjoint Loads
- Multiple Boundary Conditions for DFREQ / MFREQ in SOL 200
- Benefits of Matrix Domain ACMS in SOL 200
- ADS Optimizer
- Topology Optimization - Beta Capability
- BIGDOT Optimizer

6.1 Composite Ply Strength Ratio Response Type for the DRESP1 Entry

Introduction

The CSTRAT response type has been added to the DRESP1 entry to support the specification of composite ply strength ratios in a SOL 200 design task.

Benefits

Strength ratio output was provided as an additional available response in MSC.Nastran 2004 (see Chapter 5.6 in the *MSC.Nastran 2004 Release Guide).* This is a direct failure indicator and, as such, is an ideal response for the design of composites.

Input

The existing DRESP1 entry is used to identify the new CSTRAT response type. See the *MSC.Nastran Quick Reference Guide* for a complete description of the DRESP1 entry. The input requirements for the new CSTRAT response type are very similar to those for the existing CFAILURE response type.

Outputs

Representative output produced for the CSTRAT response based on the P2 parameter is shown in **Figure 6-1** while **Figure 6-2** shows representative formatted sensitivity output for this response.

```
                  -------------------------------------------------------------------
                 |        R E S P O N S E S    I N    D E S I G N    M O D E L       |
                  -------------------------------------------------------------------

                                (N/A - BOUND NOT ACTIVE OR AVAILABLE)

                      I N I T I A L   A N A L Y S I S   S U B C A S E =      1

                         -----    COMPOSITE LAMINA STRESS RATIO RESPONSES    -----

 ---------------------------------------------------------------------------------------------------------------
   INTERNAL   DRESP1   RESPONSE   ELEMENT  COMPONENT   LAMINA       LOWER                         UPPER
      ID        ID      LABEL        ID       NO.        NO.        BOUND          VALUE          BOUND
 ---------------------------------------------------------------------------------------------------------------
          1         3  FISUM          101          5         1        N/A       2.2166E-01       N/A
          2       333  CFAILXX        101          5         2    1.0000E+00    2.2166E-01       N/A
```

Figure 6-1 Display of the CSTRAT Response Values

```
                 ****************************************************************************
                 *                                                                          *
                 *   D E S I G N   S E N S I T I V I T Y   M A T R I X   O U T P U T        *
                 *                                                                          *
                 *                                                                          *
                 *   R E S P O N S E   S E N S I T I V I T Y   C O E F F I C I E N T S      *
                 *                                                                          *
                 ****************************************************************************

-----------------------------------------------------------------------------------------------------------------------------
 DRESP1 ID=        3          RESPONSE TYPE= CSTRAT      ELEM ID=      101     COMP NO=        5        SEID=        0
 SUBCASE  RESP VALUE  LAMINA NO.  DESIGN  VARIABLE  COEFFICIENT
-----------------------------------------------------------------------------------------------------------------------------
       1  2.2166E-01          1         1 ORIENT    -1.7345E-02

-----------------------------------------------------------------------------------------------------------------------------
 DRESP1 ID=      333          RESPONSE TYPE= CSTRAT      ELEM ID=      101     COMP NO=        5        SEID=        0
 SUBCASE  RESP VALUE  LAMINA NO.  DESIGN  VARIABLE  COEFFICIENT
-----------------------------------------------------------------------------------------------------------------------------
       1  2.2166E-01          2         1 ORIENT    -1.7285E-03
```

Figure 6-2 Formatted Sensitivity Output for the CSTRAT Response Type

Guidelines and Limitations

- The SRCOMPS parameter, which the user must set to 'YES' to obtain printed ply strength ratios, is ***not*** required for the CSTRAT response type.
- Only Item Codes 5 and 7 are available for this response.

Example (TPL: csrsens.dat)

Listing 6-1 shows a complete input file that demonstrates the new feature. Note the first DRESP1 with the CSTRAT response type has left the ATTB field blank so that the default first ply will be used for this response.

Listing 6-1 Input File with CSTRAT Response Type (csrsens.dat)

```
ID CSRSENS $
SOL 200 $
CEND
TITLE = TEST DRESP1 WITH RTYPE = CSTRAT
DSAPRT=(END=SENS)
DESOBJ = 3
DESSUB = 100
   DISP=ALL
   FORCE=ALL
   STRESS=ALL
   ANALYSIS = STATICS
   LOAD=10
$
BEGIN BULK
CQUAD4  101     101     1       2       3       4       0.
DESVAR  1       ORIENT  20.     -90.    90.0
$       ID      LABEL   RTYPE   PTYPE   REGION  ATTA    ATTB    ATT1
DRESP1  3       FISUM   CSTRAT  PCOMP           5               101
DRESP1  333     CFAILXX CSTRAT  PCOMP           5       2       101
DCONSTR 100      333     1.0    1.01
```

```
DVPREL1 1       PCOMP   101     14      -100.
        1       -1.0
FORCE   10      2               70.     1.
FORCE   10      3               70.     1.
GRID    1               0.      0.                      123456
GRID    2               1.      0.                      6
GRID    3               1.      1.                      6
GRID    4               0.      1.                      16
MAT8    1       10.     1.      .25     1.25    4.      20.     1.
                                100.    100.    1.      20.     6.
PCOMP   101                     1.+10   HILL    0.
        1       2.      -20.    YES     1       2.      -20.    YES
DOPTPRM P1      1       P2      15
ENDDATA
```

The outputs of **Figure 6-1** and **Figure 6-2** were obtained from this example. Note that the sensitivity indicates that the response in the second ply is a much weaker function of this design variable than that of the response in the first ply.

6.2 New FUNC(tions) for the DRESP2 Entry

Introduction

The DRESP2 has been extended to provide two new functions that allow for the use of the DRESP2 synthesis capability without invoking a DEQATN:

- The first new FUNC is BETA and it simplifies the input when the design task is to minimize a maximum response value.
- The second FUNC is MATCH and this provides a simplified way to specify a design task where one of the requirements is to match analysis results from MSC.Nastran with a set of results that could come, for example, from a structural test.

Benefits

The task of minimizing the maximum response has been found to be useful in a number of applications, but particularly in NVH studies where the technique can be used to minimize the peak response across a frequency range. This can be a tricky task to implement and the new BETA function provides a convenient way of specifying this design task.

Users frequently want to use SOL 200 of MSC.Nastran to obtain a better agreement between analysis and test results. This can be a tedious process and benefits from formulating the task in a way that may not be obvious to all users. The addition of the MATCH function greatly simplifies the input preparation and could result in a better agreement than would be obtained from the user's input.

Theory

Minimizing the maximum response value is a technique that has proven valuable in a number of applications and the new FUNC=BETA type on the DRESP2 entry merely simplifies the data preparation for this case by creating the following design task:

Minimize:

$$F(X_\beta) = C_1 X_\beta$$

Subject to:

$$g = \frac{r_j - \gamma X_\beta}{C_3} \leq 0$$

where C_1 and C_3 are user input values that have default values of 100.and 10.0, respectively. C_1 is used to scale the objective function and C_3 is used to offset the constraint bound from 0.

The γ quantity is computed so that the maximum constraint for all the response is equal to another user input value, C_2:

$$g_{max} = (r_{jmax} - \gamma X_\beta) / C_3 = C_2$$

Where r_{jmax} is the value of the maximum response for the initial design. The default value for C_2 is 0.005, creating a maximum constraint that is just equal to the default value of DOPTPRM parameter GMAX.

The matching of analysis results with user specified values is performed by converting the user input data to one of two user specified formulations: a. Matching Using Least Squares or b. Matching Using Minimization of the Maximum Deviation.

Matching Using Least Squares:

The least squares technique converts the user input into an objective function of the following form:

$$\phi = \sum_{j=1}^{m} \left(\frac{r_j - r_j^T}{r_j^T} \right)^2$$

where r_j is a response from a DRESP1 and r_j^T is a user defined target response.

There are no constraints spawned by this formulation but the user is permitted to specified any other desired constraints in the standard fashion.

Matching Using Minimization of the Maximum Deviation:

The second formulation is a variation on minimization of the maximum response technique described above. The objective is to minimize a spawned design variable:

$$F(X_\beta) = C_1 X_\beta$$

Subject to:

$$-\gamma X_\beta \le \frac{(r_j - r_j^T)}{\left| r_j^T \right|} \le \gamma X_\beta \qquad j = 1, 2, \ldots m$$

Because X_β can become small, it is necessary to offset the constraint in a fashion similar to the BETA method given above:

Define:

$$R_{2j} = \frac{r_j - r_j^T}{\left| r_j^T \right|}$$

And then determine R_{2max} and R_{2min}, the maximum and minimum values of R_{2j}.

The γ quantity can then be determined from user specified values of C_2 and C_3 using the following equation.

$$\frac{Max(R_{2max}, -R_{2min}) - \gamma X_\beta}{C_3} = C_2$$

Input

The new features use the existing DRESP2 entry with additional options provided for the FUNC attribute. Additionally, the user can specify three constants that are used in the two algorithms. All of these changes appear on the "parent" line of the DRESP2 entry, which now has the following form:

1	2	3	4	5	6	7	8	9	10
DRESP2	RID	Label	EQID / ***FUNC***	Region	***Method***	***C1***	***C2***	***C3***	

The modified or added terms are in italics. There are two FUNC types:

FUNC= BETA indicates that the DRESP2 specifies a min-max problem and converts the problem as shown in the "**Theory**" on page 117. This DRESP2 can only be invoked by a DESOBJ case control entry. The continuations on the DRESP2 can only supply DRESP1 data as shown in the examples section. The METHOD field in this case can be MIN (default) or MAX (maximize the minimum response).

FUNC=MATCH indicates that the DRESP2 specifies a matching task as shown in the Theory section above. The METHOD attribute can be either LS, indicating that the least-squares method is to be utilized while METHOD = BETA indicates that the minimization of the maximum normalized difference is to be performed. For METHOD=BETA, the C_i coefficients can be input, again with defaults of 100., .005 and 10.0

Outputs

There are no added outputs produced by this enhancements with the exception that the spawned design variable that is produced for FUNC=BETA or for FUNC=MATCH with METHOD=BETA, will appear in the prints of design data. The ID for the spawned design variable is the maximum number for the existing DESVAR plus 1. Similarly, spawned responses will appear as normal responses. The ID for the spawned DRESP2 starts from 100,000,001 and the response label is pre-defined as B-BETA for FUNC=BETA and as LOW-BETA or UPP-BETA for FUNC=MATCH with METHOD=BETA.

Guidelines and Limitations

- The new FUNC options on the DRESP2 can only be used on DRESP2's that are invoked by the DESOBJ entry.
- For FUNC=BETA, only DRESP1 data can be provided.
- For FUNC=MATCH, DTABLE data provides the target points for each of the responses that is specified using a DRESP1. These data must be provided in matching pairs.
- A target value of 0.0 is not recommended with FUNC=MATCH.
- Other constraints can be used in combination with the objective function specified with the new DRESP2 functionality.
- If multiple DRESP1 responses are invoked with FUNC=BETA, each must be a scalar quantity.
- If a single DRESP1 entry is invoked with FUNC=BETA, the assumption is that this one DRESP1 generates multiple responses (e.g., displacements at a series of frequencies or stresses for a number of elements).

Examples (dr2beta1, dr2mtch1 and dr2mtch2):

Three examples are provided with this delivery. The first is dr2beta1 and is similar to the acoustic optimization task of "**Acoustic Optimization**" in Chapter 7 of the *MSC.Nastran Design Sensitivity and Optimization User's Guide* modified to utilize the new BETA function on the DRESP2. The DRESP2 in this case has the form:

DRESP2	100	BETA	BETA		min	10000.	1.047	100.
	DRESP1		1					

Where C1=10000., C2=1.047 and C3=100. have been selected to produce an initial design that is similar to that in the user's guide. Note that the only user specified constraint in this case is on the weight since the remaining constraints are spawned from the DRESP2, as is the beta design variable.

The dr2mtch1 file is a simple eigenvector optimization task that seeks to minimize the difference between three components of the first eigenvector while using a more traditional constraint technique to force a match between measured and analytical results for the third eigenvector. The original form of the design task was developed as degvnt01.dat and is described in Chapter 2.1 of the *MSC.Nastran 2004 Release Guide*. A fragment of the dr2mtch1 input file is provided to illustrate the use of the new MATCH function:

DRESP2	500	DIFM1	MATCH			LS
	DTABLE		TR1	TR2	TR3	
	DRESP1	511	512	513		
DTABLE	TR1	1.432e-2	TR2	1.741e-1	TR3	6.381e-1

SOL 200 easily solves this oversimplified design task.

The dr2mtch2 file is for the same design task except that now the beta method is applied to assist in the match.

6.3 Transformation of Approximate Optimization Task to a Feasible Design

Introduction

A parameter has been added to the DOPTPRM entry that transforms an infeasible optimization task to a feasible one.

Benefits

A general statement can be made that optimization algorithms perform best when the design task does not include violated constraints. In fact, some algorithms will fail if a feasible design does not exist (the algorithms used in MSC.Nastran search for the best compromise infeasible design so that the transformation to a feasible design is not a strict requirement). In order to facilitate the use of a general optimization procedure, such as the ADS code which has been installed in MSC.Nastran 2005, a simple transformation technique has been developed to ensure that the optimization task always works in the feasible domain.

Theory

The standard optimization task minimizes an objective subject to satisfying a set of constraints. The transformed problem, designated the β Method**,** involves creating a β design variable that modifies the constraints so that:

$$g_j' = g_j - \beta \qquad (j = 1, 2, \ldots ncon)$$

and the transforms the objective function so that:

$$\Phi' = \Phi + \beta * \mathrm{PENAL} * \Phi_0$$

where PENAL is the user defined penalty parameter, Φ_0 is the initial objective function and β is initialized to the maximum initial constraint value. (If this maximum value is less than CTMIN, the transformation is not applied.) In this way, the maximum constraint value is never positive while the penalty on the objective acts to force the β value to zero. If there is a feasible design, this technique should lead to it. If there is not, this will result in a compromise design where the maximum constraint violation is minimized.

Input

The only user input is the PENAL parameter that can be input on the DOPTPRM entry. A suggested value for this parameter is 2.0, with larger values serving to move the design more forcefully toward the feasible design space. Experience to date shows that a value of 2.0 works well.

Outputs

This transformation is only applied during the approximate optimization and is therefore not evident in standard reporting of the optimization results (i.e, the results produced with DOPTPRM parameters P1 and P2, sensitivity prints and the summary DESIGN HISTORY information). The prints produced when DOPTPRM IPRINT is used to display the approximate optimization results do include this transformation, so the user must be aware of this extra, final design variable, the transformed objective and constraint values, and the effects this has on the sensitivities.

Guidelines and Limitations

- The method does not apply when the design task given the optimizer is feasible.
- Setting PENAL=2.0 seems to work well, but users are free to experiment with their application.
- PENAL must be a positive real number (0.0 is valid, but has no effect).

Example (TPL: mmfdpen.dat)

TPL file mmfdpen shows the use of the PENAL parameter in conjunction with the DOT algorithm for the Modified Method of Feasible Directions. The file is the existing mmfd.dat TPL file with the DOPTPRM entry modified to:

1	2	3	4	5	6	7	8	9	10
DOPTPRM	P1	1	p2	15	DELP	0.5	DESMAX	30	
	PENAL	10.							

Even though the DOT algorithm that is used for mmfdpen and mmfd contains strategies to deal with initially infeasible designs, the use of the PENAL parameter showed a reduction in the number of design cycles and a slight (insignificant) improvement in the final design as shown in **Table 6-1**.

Table 6-1

File Name	Final Objective	Number of Design Cycles
mmfd	6.536e4	11
mmfdpen	6.529e4	8

6.4 Residual Vectors Based on Adjoint Loads

Introduction

A new RESVEC option has been introduced that creates residual vectors for use in a modal frequency response analysis that are based on adjoint load vectors which are, in turn, are created from user input DRESP1 entries that have RTYPE's that are associated with grid responses in a frequency response subcase (i.e, FRDISP, FRVELO or FRACCL).

Benefits

Residual vectors became the default in MSC.Nastran 2004 and often show dramatic improvements in dynamic response analyses. This benefits SOL 200, and the new RESVEC option creates residual vectors that are ideal for obtaining accurate sensitivity values in modal frequency response sensitivity analysis.

Input

No new input is required for this new feature, but optional describers have been added to the RESVEC Case Control command:

ADJLOD/NOADJLOD – Control calculation of residual vectors based on adjoint loads in a modal frequency response sensitivity analysis. (Default=ADJLOD)

See the ***MSC.Nastran Quick Reference Guide*** for a complete description of the RESVEC Case Control command.

Outputs

There are no new outputs.

Guidelines and Limitations

- Residual vectors from adjoint loads are only applicable in SOL 200 when ANALYSIS=MFREQ.
- If there are FREQ3,4 or 5 entries to specify excitation frequencies, the residual vectors based on adjoint loads will not be computed automatically. In this case, the user can generate these residual vectors by using the RVDOF or RVDOF1 entry with grid and components the same as those entered on the DRESP1 entry.

Example (rvadjsens.dat)

A structural model that represents a sensor on a missile is shown below. The objective is to minimize the jitter at the sensor location so that accurate sensitivities are of benefit. The example was exercised in three different runs:

1. Using ANALYSIS=DFREQ
2. Using ANALYSIS=MFREQ with 9 normal modes and 4 residual modes available in MSC.Nastran 2004 (these modes are produced from three inertia loads and one applied load.
3. Same as 2, except an additional residual vector is created based on the DRESP1 at the sensor location.

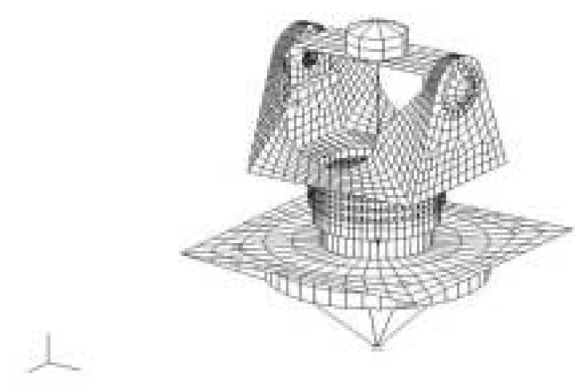


Figure 6-3 Visual Sensor Model

Figure 6-4 shows the percent error in the sensitivity results across a range of frequencies, where the error is computed as the difference in the modal frequency response compared to the direct frequency response normalized by the direct frequency response. It is seen that the single additional residual vector has a pronounced effect in reducing the error.

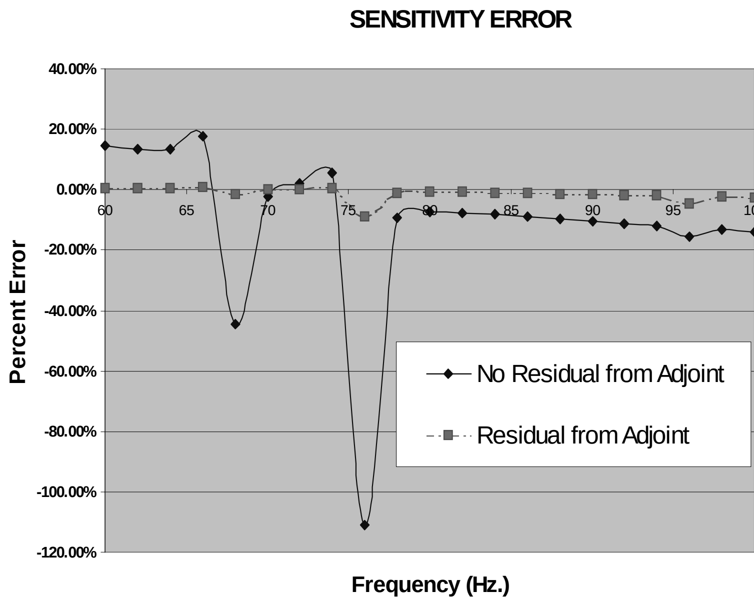

SENSITIVITY ERROR
40.00%
20.00%
0.00%
-20.00%
-40.00%
-60.00%
-80.00%
-100.00%
-120.00%
Percent Error
60
65
70
75
80
85
90
95
100
Frequency (Hz.)
No Residual from Adjoint
Residual from Adjoint

Figure 6-4

6.5 Multiple Boundary Conditions for DFREQ/MFREQ in SOL 200

SOL 200 has been capable of supporting multiple boundary conditions (MBC) for statics, normal modes, buckling and aeroelasticity for some time. However, dynamic frequency response analysis of structures subjected to multiple loads under different boundary conditions (SPC,MPC and/or SUPORT) has not been supported in SOL 200 until recently. With MSC.Nastran 2004 r3, SOL 200 is capable of performing sensitivity and optimization calculations for DFREQ and MFREQ with multiple boundary conditions.

Theory

The multiple boundary conditions for ANALYSIS = MFREQ or DFREQ in SOL 200 are implemented with the addition of a boundary condition loop in the DMAP level. Each pass of the new DMAP loop for a boundary condition has identical theoretical background.

Input/Output

The MBC for MFREQ and DFREQ in SOL 200 is implemented with no new user interface requirement. An input file can be prepared by merging several previous SOL 200 DFREQ (or MFREQ) for the same structures that have different boundary conditions or simply adding DSO related entries to a SOL 108 (or 111) file that has multiple boundary conditions.

Examples (TPL: mbc01.dat, mbc02.dat, mbc03.dat)

All three test files have analysis type of MFREQ. Portions of mbc01 will be utilized for this discussion. The subcase structure of mbc01 is shown.

```
SUBCASE =   1
     ANALYSIS = MFREQ
     LABEL =SPC Force
     SPC =      100
     DESSUB =     1
          FREQUENCY =    501
SUBCASE =   2
     DESOBJ(MIN)=1
$
     ANALYSIS = MFREQ
     LABEL =SPC Force
     SPC =      200
     FREQUENCY =    502
```

Note that the response for design objective is selected from the second subcase while responses for design constraints are from SUBCASE 1. This arrangement is only for demonstration purposes. DESOBJ with a global response, such as WEIGHT, can appear either above the SUBCASE level or in the first SUBCASE. The output for an MFREQ or DFREQ SOL 200 job with MBC is identical to those with a single boundary condition. Hence, the output example is not shown here.

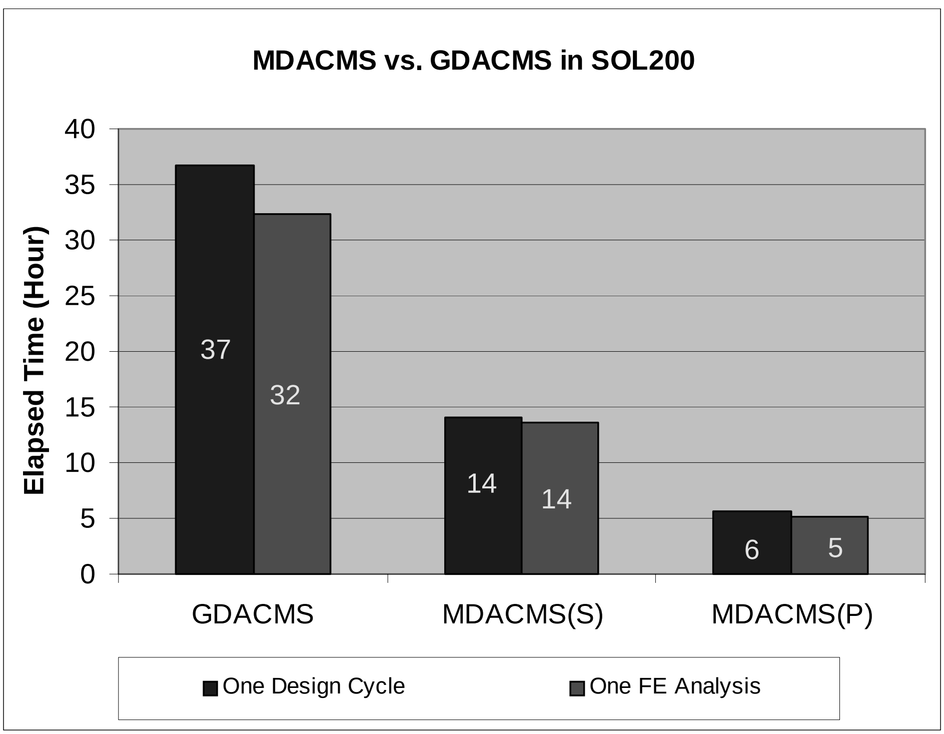
Guidelines and Limitations

- The SPC, MPC and SUPORT conditions are used to determine whether a new boundary condition has been encountered.
- DFREQ and MFREQ subcases cannot be mixed
- The restriction on a single modal transient subcase remains.

6.6 Benefits of Matrix Domain ACMS in SOL 200

Matrix Domain Automated Component Mode Syntheses (MDACMS) is the latest Lanczos solver introduced in MSC.Nastran 2005. You can find a detailed discussion in "**ACMS Now Available in the Matrix (DOF) Domain**" on page 74**.** You can use this new feature in a SOL 200 run by specifying the DOMAINSOLVER ACMS (partopt=dof) command. It is available for Analysis=MODES and / or =MFREQ and both serial and parallel processing are supported. A parallel run is activated when DMP = n and n > 1.

This new feature has been tested with an NVH optimization task with a 4.2M dofs, for an eigenanalysis up to 300 Hz that produced 1960 modes. Three separate runs were performed. The following plot shows that a parallel (4 processor) MDACMS SOL 200 job achieves more than 6 times speedup relative to a Geometric Domain ACMS (GDACMS) SOL 200 run (reduced time from 37 Hrs to 6 Hrs) while the serial MDACMS SOL 200 run reduces to 14 Hrs for a 2.5 times speedup.



6.7 ADS Optimizer

Introduction

ADS is a public domain FORTRAN program for Automated Design Synthesis developed by G.N. Vanderplaats in 1985 and is documented in Vanderplaats, G.N., **ADS – A FORTRAN Program for Automated Design Synthesis** – Version 1.10, NASA Contractor Report 177985, NASA Langley Research Center, Hampton, Virginia, 1985.

In MSC.Nastran 2005, an MSC.Software enhanced ADS version is added to support SOL 200.

Benefits

The availability of the ADS optimizer provides an alternative to the existing optimizer options that are provided in the DOT optimizer that has been the workhorse optimization algorithm for many years. As indicated below, the ADS code is a suite of optimization techniques and it may be that a particular optimization task will achieve improved results when using one of the ADS techniques. Notably, ADS contains SUMT (Sequential Unconstrained Minimization Techniques) methods that are not available with DOT. Users who have had experience in the use of the public domain ADS code should welcome having these algorithms available in MSC.Nastran. The ADS optimizer will use your current optimization license, and does not require any special licensing.

Input

There are two ways to use ADS code. One way is by modifying an executive system parameter OPTCOD in the Runtime Configuration (RC) file as shown in **Table 6-2**

Table 6-2 System Cell Summary

System Cell Number	System Name	Function and Reference
413	OPTCOD	Specifies which optimization code to be used in SOL 200. Optimization method is selected by parameter METHOD on DOPTPRM entry 0 (default) – DOT (Design Optimization Tool) 1 – Enhanced ADS code

The second way is that MSC.Nastran allows users to select an ADS optimization algorithm is by a new parameter ADSCOD added to DOPTPRM Bulk Data entry that has the options shown in **Table 6-3**

Table 6-3 DOPTPRM Design Optimization Parameters

Name	Description, Type and Default Value	
ADSCOD	Optimization Code: (Integer $\geq$ 0, Default = 0, see Remark 1)	
	= 0	DOT used
	= 1	ADS Modified Method of Feasible Directions
	= 2	ADS Sequential Linear Programming
	= 3	ADS Sequential Quadratic Programming
	= 4	SUMT Method
	= IJK	where
	I -ADS strategy options (Integer 0-9 Default = 0 see Remark 2)	
	0	None, Go directly to the optimizer
	1	Sequential unconstrained minimization using the exterior penalty function method
	2	Sequential unconstrained minimization using the linear extended interior penalty function method
	3	Sequential unconstrained minimization using the quadratic extended interior penalty function method
	4	Sequential unconstrained minimization using the cubic extended interior penalty function method
	5	Augmented Lagrange multiplier method
	6	Sequential linear programming
	7	Method of centers
	8	Sequential quadratic programming
	9	Sequential convex programming
	J - ADS optimizer options (Integer 1-5)	
	1	Fletcher-Reeves algorithm for unconstrained minimization
	2	Davidon-Fletcher-Powell (DFP) variable metric method for unconstrained minimization

Table 6-3 DOPTPRM Design Optimization Parameters

Name	Description, Type and Default Value	
	3	Broydon-Fletcher-Goldfarb-Shanno (BFGS) variable metric method for unconstrained minimization
	4	Method of feasible directions for constrained minimization
	5	Modified method of feasible directions for constrained minimization
	K - ADS one-dimensional search options (integer 1-8)	
	1	Find the minimum of an unconstrained function using the Golden Section method
	2	Find the minimum of an unconstrained function using the Golden Section method followed by polynomial interpolation
	3	Find the minimum of an unconstrained function by first finding bounds and then using polynomial interpolation
	4	Find the minimum of an unconstrained function by polynomial interpolation / extrapolation without first finding bounds on the solution
	5	Find the minimum of a constrained function using the Golden Section method
	6	Find the minimum of a constrained function using the Golden Section method followed by polynomial interpolation
	7	Find the minimum of a constrained function by first finding bounds and then using polynomial interpolation
	8	Find the minimum of a constrained function by polynomial interpolation / extrapolation without first finding bounds on the solution

Remarks:

1. If ADSCOD>0, ADSCOD will override the METHOD.
2. A more complete description of the available options can be found in the Reference cited in the introduction of this section.

Output

The outputs used with the ADS code are identical, in terms of format, to those using the DOT code with the exception of the prints produced using the DOPTPRM IPRINT parameter. The IPRINT output from ADS will differ for each option but all IPRINT results are headed by a banner the identifies the results as coming from the MSC.Software enhanced version of ADS.

Guidelines and Limitations

The ADS code has been provided in response to client requests for an alternative optimization algorithm to the DOT code. MSC.Software has performed extensive testing of using ADS on our suite of over 400 test problems. The basic conclusion from these tests is that the ADS code performs adequately on many of these tests, but that there is no compelling case to recommend the use of ADS on a general basis. It is recommended that knowledgeable users apply ADS to some of their difficult optimization tasks and see if they can obtain improved results. In particular, ADS SUMT methods can comfortably solve problems with a few thousand design variables. MSC.Software would be interested in hearing of any experience along these lines.

One guideline is that while the DOT code has techniques for dealing with infeasible designs, ADS is weak in this area. In this case, it is recommended that the new PENAL parameter on the DOPTPRM entry be used to transform the optimization task from an infeasible to a feasible one.

6.8 Topology Optimization - Beta Capability

Introduction

Unlike sizing and shape optimization, topology optimization finds an optimal distribution of material, given the package space, loads, and boundary conditions. These methods have grown rapidly in popularity and application in recent years and topology optimization methods have been discussed in a large number of publications. An overview of topology optimization can be found in a book by Bendsoe and Sigmund [1] and a review article by Rozvany *et al* [2].

MSC.Software has integrated a topology optimization capability into MSC.Nastran 2005 that is based on the increasingly popular density approach to topology optimization. In the density method, Young's modulus E and density ρ are used as intermediate design variables for each designable finite element. The actual design variable x is the normalized density that links Young's modulus E and density ρ for designable finite elements using the following relationships

$$\rho = \rho_0 x$$

$$E = E_0 x^p$$

where ρ_0 and E_0 are respectively the fully solid Young's modulus and density. A penalty factor p is introduced to enforce the design variable to be close to a 0-1 solution when p>1.0. The penalty factor p usually takes values between 2 and 4.

The general topology optimization problem available in MSC.Nastran can be stated as follows:

Minimize:

$$f(x_i)$$

Subject to:

$$g_j(x_i) \le 0.0 \qquad j = 1, \ldots, M$$

$$\eta \le x_i \le 1.0 \qquad i = 1, 2, \ldots, N$$

where g_j represents the j-th constraints and M is the total number of constraints. The constraint specification can be general in that any of the response types currently available in SOL 200 can be used. N is the total number of designable elements. η is a small positive number to prevent the stiffness matrix singularity.

Benefits

Topology optimization can generate more efficient design concepts in the early design stage, especially for load paths. Topology optimization can also be to used to obtain rib patterns and weld distribution patterns. The BIGDOT optimizer is available to solve problems with a large number of design variables and constraints that DOT struggles with due to computer memory requirements and efficiency.

Input

Topology optimization in MSC.Nastran borrows heavily from the user interface developed for sizing and shape optimization. In particular, the design objective and constraints are defined in an identical manner for topology and sizing/shape optimization. This section discusses the additional bulk data entry that has been provided to ease the creation of the design variables and then discusses other features that have been adapted for topology optimization

TOPVAR – Topological Design Variables

To select a topologically designable region, the user needs to specify a group of elements. All elements referencing a given property ID are made topologically designable with the Bulk Data entry TOPVAR. Topology design variables are automatically generated with one design variable per designable element.

The basic format for TOPVAR is:

1	2	3	4	5	6	7	8	9	10
TOPVAR	ID	LABEL	PTYPE	XINIT	XLB	DELXV	POWER	PID	

Field	Contents
ID	Unique topology design region identification number. (Integer>0)
LABEL	User-supplied name for printing purpose. (Character)
PTYPE	Property entry name. Used with PID to identify the elements to be designed. (Character: “PBAR”, “PSHELL”, etc.)
XINIT	Initial value. (Real, XLB<=XINIT). Typically, XINIT is defined to match the mass target constraint, so the initial design does not have violated constraints. For example, if the mass target is 30%, then it is suggested XINIT=0.3.
XLB	Lower bound. (Real, Default = 0.001)

Field	Contents
XLB	Upper bound (real, Default = 1.0)
DELXV	Fractional change allowed for the design variable during approximate optimization. (Real > 0.0, default = 0.2 see Remark 3).
POWER	A penalty factor used in relation between topology design variables and element Young's modulus. (Real > 1.0, default =3.0). 2.0<=POWER<=4.0 is suggested.
PID	Property entry identifier (Integer > 0)

Remarks:

1. The topologically designable element property includes PROD, PBAR, PBARL, PBEND, PBEAM, PBEAML, PSHELL, PSHEAR, PSOLID, and PWELD. Multiple TOPVARs are allowed to design different element types in a single file.
2. All designed element properties must refer to a MAT1 entry; therefore, a PCOMP cannot be used in topology optimization.
3. If DELXV is bank, the default is taken from the specification of DELX parameter on the DOPTPRM entry.

New Responses - Compliance and Fractional Mass

The existing DRESP1 entry has been extended to provide two new response types that are available exclusively for topology optimization. The format for the new responses is shown in **Table 6-4** and it is seen that both new response types require only the specification of the response type and no other attributes.

Table 6-4 New Responses for Topology Optimization

Response Type (RTYPE)	Response Attributes		
	ATTA (Integer>0)	ATTB (Integer>0 or Real>0.0)	ATTI (Integer>0)
COMP Remark 1	BLANK	BLANK	BLANK
FRMASS Remark 1,2	BLANK	BLANK	BLANK

Remarks:

1. RTYPE=COMP (compliance of structures = $p^T u$) and FRMASS (mass fraction of designed elements) entries are used for topology optimization only.
2. RTYPE=FRMASS is the mass divided by the mass calculated if all design variables are 1.0. FRMASS is calculated for designed elements only. FRMASS = 1.0 if all design variables are 1.0

The COMP and FRMASS response types are provided to facilitate the specification of the classical topology optimization task of minimizing the compliance of a loaded structure while limiting the mass to some percentage of the maximum allowable amount. In MSC.Nastran's implementation, these responses can be applied generally so that the COMP response could lead to a constraint and the minimization of FRMASS could be an objective.

New and Modified Design Optimization Parameters (DOPTPRM)

Two new design optimization parameters are added for topology optimization in SOL 200 as shown in **Table 6-5**. A new parameter TCHECK is used to turn on / off a filtering algorithm to prevent the checkerboard like material distribution. Another parameter TDMIN is introduced to achieve mesh independent solutions, control the size of members in the topology optimized design, and therefore the degree of simplicity in terms of manufacturing considerations.

In addition, a number of existing DOPTPRM parameters have different default values for topology optimization as opposed to Sizing / Shape optimization, as shown in **Table 6-6**. As described in "**BIGDOT Optimizer**" on page 147, the BIGDOT optimization algorithm is available for topology optimization problems with many (>2000) designed elements. This is selected by setting DOPTPRM parameter METHOD to 4.

Table 6-5 New DOPTPRM Design Optimization Parameters

Name	Description, Type, and Default Value
TCHECK	Topology Filtering options (integer 0 or 1) 1 Filtering algorithm is on for topology optimization (default) 0 No filtering algorithm
TDMIN	Topology minimum member diameter (real $\geq$ 0.0) in the basic coordinate system. Default =0.0 (i.e., no minimum member size control). This option is applied on 2 and 3 D elements only.

Table 6-6 Default Values for DOPTPRM Design Optimization Parameters

Parameter	Sizing/Shape	Topology
DESMAX	5	30
CONV1	0.001	1.0E-5
CONVDV	0.001	1.0E-4
DELX	0.5	0.2
DXMIN	0.05	1.0E-5

As a final comment on DOPTPRM parameters, it was necessary to change the definition of the P2 parameter that controls the amount of print that occurs at design cycles specified by P1. For sizing and shape optimization, design variables are printed for any value of P1 = 1 (or if 1 is including in the sum of the options). Since a topology optimization task can easily result in thousands of design variables, this would not be a viable option for most problems. Instead, design variable prints are turned off unless P2 value greater than 8 is specified.

Outputs

P2=1 (default) on Bulk Data entry DOPTPRM does not print topology design variables to minimize optimization output since topology optimization involves in a large number of design variables. P2>8 prints topology design variables.

Output in for the two new responses, compliance and fractional mass, and topology design variables are shown if **Figure 6-5**. Also in this figure, the design variable history shows the external element ID associated with the internal design variable ID.

```
                         -----   COMPLIANCE RESPONSES -----
 ----------------------------------------------------------------------------
        INTERNAL  DRESP1   RESPONSE          LOWER                      UPPER
           ID       ID      LABEL            BOUND         VALUE        BOUND
 ----------------------------------------------------------------------------
            1        1      COMPL              N/A        1.4162E+02      N/A


    -----     FRACTIONAL MASS RESPONSES -----
----------------------------------------------------------------------------
        INTERNAL   DRESP1   RESPONSE          LOWER                     UPPER
           ID        ID      LABEL            BOUND       VALUE         BOUND
 ----------------------------------------------------------------------------

            2        2      FRMASS             N/A     3.0000E-01   3.0000E-01



******************************************************************************
        S U M M A R Y   O F   D E S I G N   C Y C L E   H I S T O R Y

******************************************************************************
                         DESIGN VARIABLE HISTORY
 ----------------------------------------------------------------------------
 INTERNAL |  EXTERNAL  |          |
   DV. ID. |  ELEMENT ID  |    LABEL    |   INITIAL    :      1       :      2
 -----------------------------------------------------------------------------------------
---------------------------------------
        1 |          1   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
        2 |          2   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
        3 |          3   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
        4 |          4   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
        5 |          5   |  TOPVAR     |   3.0000E-01 :   3.6000E-01 :
        6 |          6   |  TOPVAR     |   3.0000E-01 :   3.6000E-01 :
        7 |          7   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
        8 |          8   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
        9 |          9   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
       10 |         10   |  TOPVAR     |   3.0000E-01 :   2.4000E-01 :
 -------------------------------------------------------------------------
```

Figure 6-5 New Output in jobname.f06

PARAMETER, DESPCH – specifies when the optimized Bulk Data entries are written to the PUNCH file for sizing and shape optimization. In topology optimization, DESPCH is used to specify when the topology optimized element density values are written to the topology element density history file jobname.des. This file can be written in one of two formats. The first format is a MSC.Patran neutral element results file that can be used with a custom template file (.res_tmpl) to display topology results on MSC.Patran. This format is obtained by default. In order to support MSC.Nastran-OptiShape users, this file can also be written in OptiShape Patran Preference format by setting PARAM,DESPCH1=-1. Thus, MSC.Nastran-OptiShape users can display and animate SOL 200 topology optimization results using the MSC.Nastran-OptiShape Patran Preference. **Figure 6-6** shows and element density history file using the OptiShape Preference format.

```
/DENSI/                          Flag  for element density file
     1                            Design cycle ID

   10
      1   0.240                 External element ID and density value
      2   0.240
      3   0.240
      4   0.240
      5   0.360
      6   0.360
      7   0.240
      8   0.240
      9   0.240
     10   0.240
```

Figure 6-6 Element Density History File jobname.des

Guidelines and Limitations

The quality of the results of a topology optimization task is a strong function of how the problem is posed in MSC.Nastran. This section contains a number of tips that have been developed based on extensive testing of this new capability.

- A new DRESP1=COMP is introduced to define the compliance of structures for topology optimizations. The response is usually used as an objective to maximize structural stiffness in static design problems.
- A new DRESP1=FRMASS is introduced to define the mass fraction of topology designed elements. The DRESP1=WEIGHT is the total weight of all structural and non-structural mass. For topology optimization tasks DRESP1=FRMASS response is recommended to define a mass reduction target in a design constraint.
- The POWER field on the TOPVAR entry has a large influence on the solution of topology optimization problems. A lower POWER often produces a solution that contains large “grey” areas (area with intermediate densities 0.3 – 0.7). A higher value produces more distinct black and white (solid and void) designs. However, near singularities often occur when a high POWER is selected.

- A parameter TCHECK on DOPTPRM is used to turn on / off the checkerboard free algorithm. The default of TCHECK=1 activates the filtering algorithm. This default normally results in a better design for general finite element mesh. However, if high order elements and / or a coarser mesh is used, turning off the filtering algorithm may produce a better result.
- The parameter TDMIN is mainly used to control the degree of simplicity in terms of manufacturing considerations. It is common to see some members with smaller size than TDMIN at the final design since the small members have contributions to the objective. Minimum member size is more like quality control than quantity control.
- XINIT on the TOPVAR entry should match the mass target constraint so that the initial design is feasible.
- Maximum design cycle DESMAX=30 (as default) is often required to produce a reasonable result. More design cycles may be required to achieve a clear 0 / 1 material distribution, particularly when minimum member size control used.
- There are many solutions to a topology optimization, one global and many local minimization. It is not unusual to see different solutions to the same problem with the same discretization by using different optimization solvers or the same optimization solver with different starting values of design variables.
- In a multiple subcase problem, a Case Control command DRSPAN can be used to construct a weighting function via a DRESP2 or DRESP3. For example, a static and normal mode combined problem, the objective can be defined as

$$\text{obj} = \text{weight1} \cdot \left\{\frac{c_1}{c_0}\right\} + \text{weight2} \cdot \left\{\frac{\lambda_0}{\lambda_1}\right\}$$

  where weight1 and weight2 are two weighting factors. c_1 is the calculated compliance and λ_1 is the calculated eigenvalue via DRESP1 definition. c_0 and λ_0 are the initial value of these responses.
- The parameter BAILOUT =0 (default) may cause the topology optimization run to exit if near singularities are detected. Users may increase the value of XLB on TOPVAR to further prevent the singularity or set BAILOUT =-1 to cause the program to continue processing with near singularities.

- To obtain a rib pattern by topology optimization, a core non-designable shell element thickness must be defined together with two designable above and below the core thicknesses. That is, add two designable elements for each regular element.
- Elements referencing the composite property PCOMP entry cannot be designed.
- Superelements are not supported.
- Topology design variable cannot used together with other type design variables
- Topology design sensitivity is not supported

Numerical problems often occur when solving a topology optimization task. The nature of the problem depends on element type, number of elements, optimization algorithm and so on. One frequent numerical problem is the so-called checkerboard effect. Checkerboard-like material distribution pattern is observed in the topology optimization of continuum, especially when first order finite elements, such as CQUAD4, are employed to analyze structural responses. It has been shown that the Checkerboard-like phenomenon is caused by the finite element formulation. The problem occurs because the checkerboard has an artificially high stiffness compared with a structure with uniform material distribution [1]. The easiest way to decrease the checkerboarding effect is to use higher order elements (such as CQUAD8). This however increase the CPU-time considerably. Another closely related phenomenon is mesh-dependent solutions. It is seen that a more detailed structure is found by increasing the number of elements. The ideas of making a finer finite element mesh is to get a better finite element solution. However, this finer meshing tends to have an increasing number of members with decreasing size. This more detailed topology solution creates a problem from a manufacturing point of view. An overview of the techniques used to avoid the checkerboarding and mesh-dependent solutions can be found in the reference [1]. In SOL 200, filtering algorithms are used to promote a checkerboard-free and mesh independent topology optimized solution.

Topology otimization is powerful tool to generate design concepts in the early design stage. Unfortunately, the topology optimzed designs usually turn out to be infeasible for certain manfacturing processes, such as casting and extrusion. This issue will be addressed in a future MSC.Nastran release.

Example 1 (topex1.dat)

This example leads to a conceptual design of a bicycle frame in a 2D situation by maximizing the stiffness for a given amount of material (70% mass reduction) that (shown in **Figure 6-7**) satisfies two boundary condition and load cases.

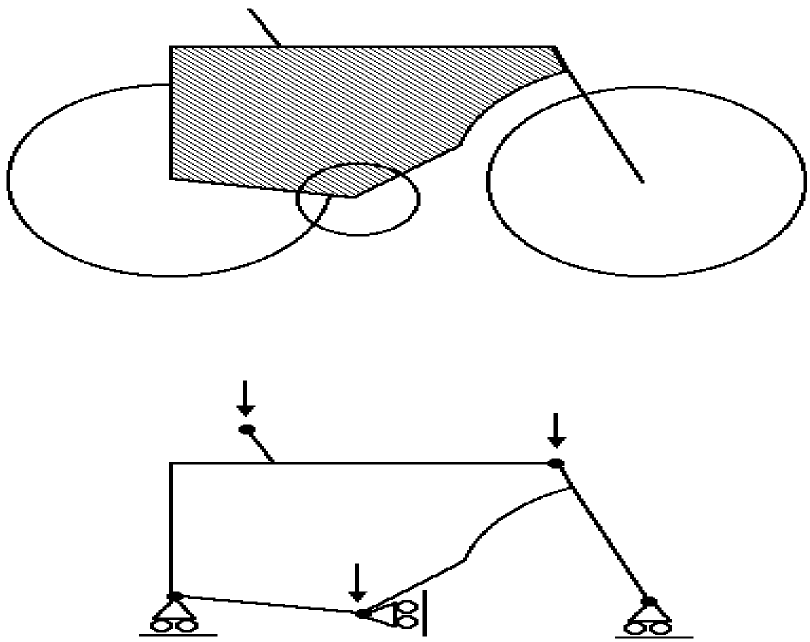
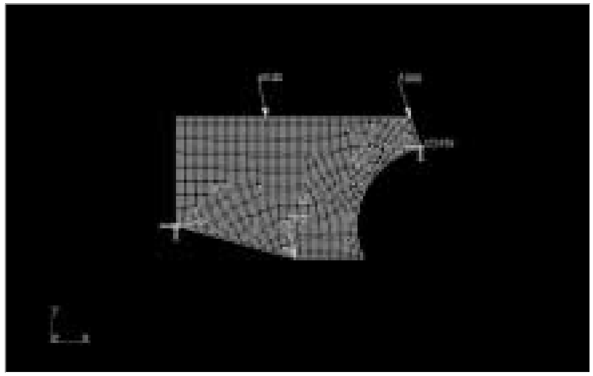


Figure 6-7 Bicycle Frame

Two loading and constraint conditions are assumed corresponding to two scenarios riding the bicycle on sitting and standing positions as shown in the figures below. There are 2442 QUAD4 elements and 2 TRIA3 elements.



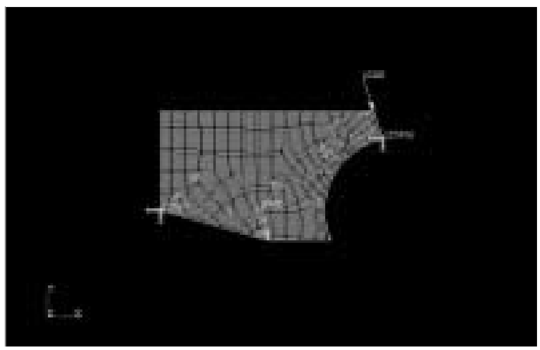


Figure 6-8 FE Model of a Bicycle Frame

The input data for this example that is related to topology optimization is listed in **Listing 6-2**. The result shown in **Figure 6-9**. is similar to existing bicycle frames.

Listing 6-2 Input File for Example 1

```
$ Topology Optimization Example 1/ XMY
$id msc, topex1 $ v2005 4-Jun-2004 xmy
SOL 200        $  OPTIMIZATION
CEND
$
SEALL = ALL
SUPER = ALL
ECHO = NONE
set 7 = 20
set 9 = 40
DESOBJ = 1
DESGLB = 1
SUBCASE 1
   SUBTITLE=LOAD CASE 1
   SPC = 2
   LOAD = 7
   DRSPAN = 7
```

```
   ANALYSIS = STATICS
SUBCASE 2
   SUBTITLE=LOAD CASE 2
   SPC = 2
   LOAD = 9
   DRSPAN = 9
   ANALYSIS = STATICS
BEGIN BULK
$
TOPVAR,    1  ,     TSHELL,     PSHELL, .3, , , ,   1
DRESP1   2      FRM      FRMASS
DRESP1, 20,     COMP1,     COMP
DRESP1, 40,     COMP2,     COMP
DRESP2   1        COMPL     SUM
DCONSTR  1       2                .3
```

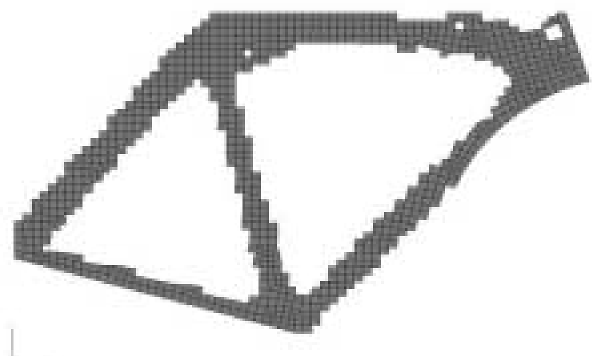


Figure 6-9 Topology result of a Bicycle Frame

References

1. Bendsoe, M.P. and Sigmund, O. *Topology Optimization Theory, Methods, and Applications*, Springer, 2003.
2. Rozvany, G.I.N., Bendsoe, M.P., and Kirsch U., *Layout Optimization of Structures*, Appl. Mech. Rev., 48, 1995, pp.41-119

6.9 BIGDOT Optimizer

Introduction

BIGDOT is an optimization algorithm that has been developed by VR&D to solve large optimization tasks. A guideline for the DOT optimizer (the workhorse optimizer in SOL 200) is that it can comfortably address problems with several hundred design variables and can be stretched to a few thousand design variables. By contrast, BIGDOT has demonstrated the ability to solve problems with tens of thousands of design variables with the maximum size approaching one million variables. A reference for the BIGDOT algorithm is: **Vanderplaats, G., 'Very Large Scale Optimization'**, presented at the 8th AIAA / USAF / NASA / ISSMO Symposium at Multidisciplinary Analysis and Optimization, Long Beach, CA September 6-8, 2000.

The BIGDOT algorithm is available in MSC.Nastran 2005 and is offered as an additional option as a royalty product that is outside the MasterKey concept. Potential users of this capability should contact their MSC sales representative to get information about the “Topology Optimization” option within MSC.Nastran. The Guidelines and Limitations section of this subchapter discusses how this new option interacts with the standard “Design Optimization” option.

Benefits

The primary benefit of including the BIGDOT option is that it enables the ability to perform topology optimization of real-world structures. As “**Topology Optimization - Beta Capability**” on page 135 indicates, topology optimization entails creating a design variable for each individual element so that one can very quickly exceed to the several thousand design variable practical limitation that is mentioned above for the DOT algorithm.

A second benefit that will be of interest to some clients is that it can be applied in sizing applications where the number of design variables is in the thousands and above.

Input

BIGDOT is available in MSC.Nastran by specifying METHOD=4 on the DOPTPRM entry. **Table 6-7** contains the meanings of the four options for this parameter.

Table 6-7 Meaning of the METHOD Parameter on the DOPTPRM Entry

Value	Description
1	Modified Method of Feasible Directions using DOT (default for non-topology optimization problems)
2	Sequential Linear Programming using DOT
3	Sequential Quadratic Programming using DOT
4	BIGDOT (default for topology optimization problems)

The remaining DOPTPRM parameters that govern the behavior of the optimizer are identical between DOT and BIGDOT so that no additional inputs are required.

Output

The output from BIGDOT algorithm itself is controlled by existing DOPTPRM parameter IPRINT. There are no other outputs that are affected by BIGDOT.

Guidelines and Limitations

The BIGDOT algorithm is intended for problems with many design variables. For problems with fewer than one thousand variables, the DOT or ADS algorithms are recommended.

As mentioned in the Introduction to this section, the BIGDOT algorithm is available to users that have purchased the “Topology Optimization” (TO) option for MSC.Nastran. This complements the existing “Design Optimization” (DO) option in the following way:

1. If the user has acquired the DO option only, this enables standard shape and sizing optimization and topology optimization with a limited number of design variables. The optimizer can be either DOT or ADS.
2. If the user has acquired the TO option only, this enables general topology optimization tasks but does not enable standard shape and sizing optimization. The optimizer is BIGDOT.
3. If the user has both DO and TO, the BIGDOT algorithm can then be applied to topology and shape and sizing optimization tasks with a large number of design variables. The optimizer can be BIGDOT or DOT or ADS.

Example

The “**Example 1 (topex1.dat)**” on page 144 (Topology Optimization) utilizes BIGDOT.

CHAPTER 7 Rotor Dynamics

- Squeeze Film Damper Nonlinear Force

7.1 Squeeze Film Damper Nonlinear Force

Introduction

The requirement for high power output from modern gas turbine engines has resulted in highly flexible light weight rotor designs. Control of vibration response in these engines is a major design problem. The use of rolling element bearings with low inherent damping makes it difficult to reduce vibration amplitudes and dynamic loads transmitted to the rotor supporting structure. Squeeze film dampers (SFDs) are therefore used to provide adequate damping to maintain low amplitude vibration levels and to reduce the dynamic loads transmitted to the bearings and rotor support structures.

The general SFD model has been sucessfully incorporated into the MSC.Nastran time-domain analysis and this new capability provides the means to design and analyze SFDs for general rotor orbits with multiple frequency content. The new capability includes static loads and models the lift-off phenomenon important in the design of free-floating dampers.

Squeeze Film Damper Model Imbedded in MSC.Nastran Transient Solution

In a coordinated effort between GEAE and MSC, the general SFD model was incorporated in MSC.Nastran for transient analysis. This was accomplished by inserting the SFD forces in the right-hand (Force Vector) side of the equations of motion – no special element was added to MSC.Nastran. GEAE provided MSC with the SFD FORTRAN code, a description of the input/output data, the variable definitions, and a checkout two-degree-of-freedom test model and results.

The SFD code lends itself to a form of a NOLIN type of element similar to NLRGAP. The NOLIN approach works with the NASTRAN time domain solutions (SOL 109 and SOL 129). The new SFD element is called NLRSFD. The Bulk Data entry NLRSFD is used to input the SFD data (journal diameter, land length, oil viscosity, etc.).

As with the NOLINS, the NLRSFD will be selected by the NONLINEAR Case Control command.

The SFD code uses as input the relative displacements and velocities x, $\dot{x}$, y, $\dot{y}$ at the connecting grids and outputs the forces $F_x(x, \dot{x}, y, \dot{y})$ and $F_y(x, \dot{x}, y, \dot{y})$ acting on the SFD damper journal grid point. Equal and opposite forces $-F_x(x, \dot{x}, y, \dot{y})$ and $-F_y(x, \dot{x}, y, \dot{y})$ are applied to the stator (SFD housing) grid point.

Referring to **Figure 7-1**, GRID I is on the damper journal and GRID J is on the damper housing. The two grids should be coincident and have parallel Cartesian coordinate systems. The forces applied to the grids are based on the relative displacements and velocities of the grids determined from the previous time steps in the NASTRAN implicit time integration. If a parallel centering spring is used, then this separate spring is entered using the CELAS2 two-ended element.

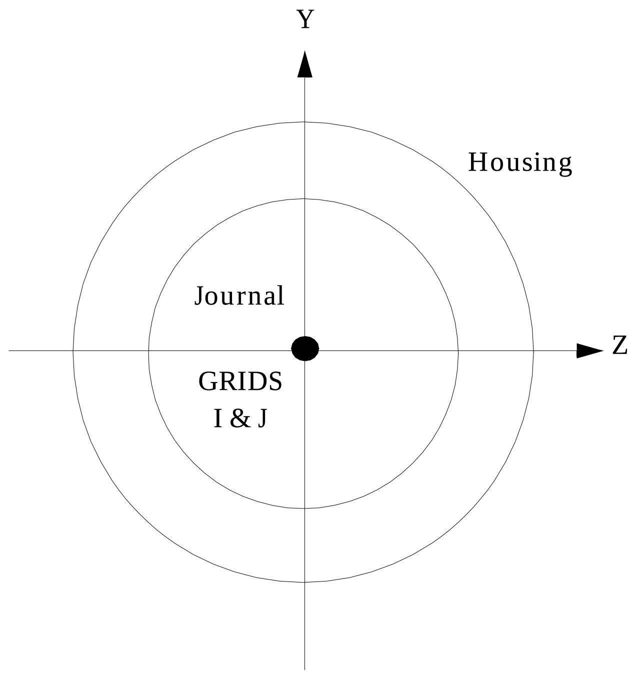


Figure 7-1 Imbedding the SFD Model in MSC.Nastran: Grid I is on the Damper Journal and Grid J is on the Damper Housing

Theory for General Squeeze Film Damper Model

The squeeze film damper model is based on work originally performed at Case Western Reserve University (CWRU). It incorporates a numerical solution of the Reynolds lubrication equation for incompressible laminar isoviscous films that is described in Reference **1.** The model is capable of handling the specified pressure boundaries at the feed (supply) and discharge (drain) ports of the SFD. The SFD pressure distribution is determined using a one-dimensional, finite difference scheme. The scheme is a 1-D adaptation of the 2-D finite difference method of Castelli and Shapiro, Reference **2.** The one-dimensional finite-difference approach permits the account of static as well as dynamic deflections and is thus capable of modeling general damper orbits with broad frequency content. The model computes the oil film forces by numerical integration of the instantaneous film pressure distribution.

Squeeze Film Damper Input Data Format

The squeeze film damper (SFD) is implemented as a nonlinear force similar to the NLRGAP. The SFD forces are activated from the Case Control Section using the NONLINEAR command.

NONLINEAR= n

The Bulk Data entry for the SFD has the following form:

1	2	3	4	5	6	7	8	9	10
NLRSFD	SID	GA	GB	PLANE	BDIA	BLEN	BCLR	SOLN	
	VISCO	PVAPCO	NPORT	PRES1	THETA1	PRES2	THETA2	NPNT	
	OFFSET1	OFFSET2							

Field	Contents
SID	Nonlinear load set identification number. (Integer > 0, Required)
GA	Inner (e.g., damper journal) grid for squeeze film damper. (Integer > 0, Required)
GB	Outer (e.g., housing) grid for squeeze film damper. (Integer > 0, Required)
PLANE	Radial gap orientation plane: XY, XZ, or ZX. See Remark 1. (Character, Default = XY)
BDIA	Inner journal diameter. (Real > 0.0, Required)
BLEN	Damper length. (Real > 0.0, Required)
BCLR	Damper radial clearance. (Real > 0.0, Required)
SOLN	Solution option: LONG or SHORT bearing. (Character, Default = LONG)
VISCO	Lubricant viscosity. (Real > 0.0, Required)
PVAPCO	Lubricant vapor pressure. (Real > 0.0, Required)
NPORT	Number of lubrication ports: 1 or 2 (Integer, no default)
PRES1	Boundary pressure for port 1. (Real > 0.0, Required if NPORT = 1 or 2)

THETA1	Angular position for port 1. See Remark 2. (0.0 ≤ Real > 360.0, Required if NPORT = 1 or 2).
PRES2	Boundary pressure for port 2. (Real > 0.0, Required if NPORT = 2).
THETA2	Angular position for port 2. See Remark 2. (0.0 ≤ Real < 360.0, Required if NPORT = 2)
NPNT	Number of finite difference points for damper arc. (Odd integer ≤ 201, Default = 101)
OFFSET1	Offset in the SFD direction 1. (Real, Default = 0.0)
OFFSET2	Offset in the SFD direction 2. (Real, Default = 0.0)

Remarks:

1. The XY, YZ, and ZX planes are relative to the displacement coordinates of GA and GB. The plane coordinates correspond to the NLRSFD directions 1 and 2. GA and GB should be coincident grids with parallel displacement coordinate systems. Wrong answers will be produced if this rule is not followed.
2. The angular measurement is counterclockwise from the displacement x-axis for the XY plane, the y-axis for the YZ plane, and the z-axis for the ZX plane.
3. OFFSET1 = Damper housing ID center offset displacement relative to OD center in the horizontal direction. Entered as a positive value for horizontally to the left (negative x-direction) displacement.
4. OFFSET2 = Damper housing ID center offset displacement relative to OD center in the vertical direction. Entered as a positive value for downward (negative y-direction) displacement. Positive entry typically used for -1 g compensation.

Note: The OFFSET2 value represents an eccentric damper housing in the vertical direction and is typically used to compensate for the -1g displacement of damper supported by a centering spring.

Squeeze-Film Damper Example

The following example demonstrates the use of the new NLRSFD nonlinear force. The model is shown in **Figure 7-2**. The MSC.Nastran input file is shown in **Listing 7-1**. The unbalance load of 20 Gm-cm is used to excite the structure. The resulting nonlinear forces are shown in **Figure 7-3**.

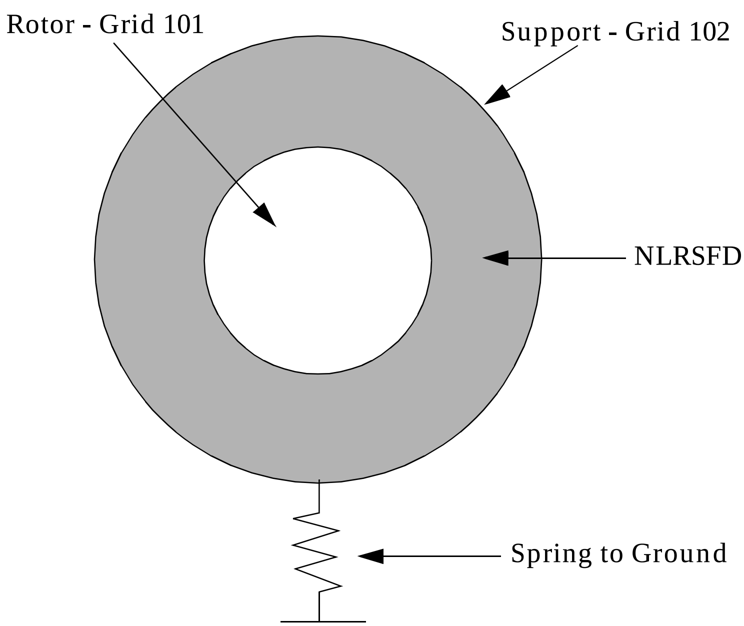


Figure 7-2 Model

Listing 7-1

```
NASTRAN
SOL 109
CEND
TITLE = Simple test model, SOL 109, No damping
$
ECHO= UNSORT
$
$------------- Results requests --------------------
SET 101 = 100,101
DISP    (PRINT,PUNCH,SORT2) = 101
SET 102 = 101,102
ELFORCE (PRINT,PUNCH,SORT2) = 102
$
TSTEP = 999
NONLINEAR=1
SUBCASE 200
 LABEL = 20 gm-in unbalance
```

```
 DLOAD = 200
$
OUTPUT(XYPLOT)
XGRID= YES
YGRID= YES
XTITLE= TIME (SEC)
YTITLE= SFD FORCE (X)
XYPLOT  NONLINER/ 101(T1)
YTITLE= SFD FORCE (Y)
XYPLOT  NONLINEAR/ 101(T2)
BEGIN BULK
$
$               1/386.4
PARAM     WTMASS258799-8
PARAM     GRDPNT       0
$
$1.....12......23......34......45......56......67......78
$
TSTEP        999   30001 .000010     100
$
$================================================================================
$
DLOAD        200     1.0    20.0     301    20.0     302
$
TLOAD2       301     301            LOAD     0.0   100.0166.6667   270.0
TLOAD2       302     302            LOAD     0.0   100.0166.6667     0.0
$
$  F(f) = UNBAL * f**2 * (1/453.5924 lbm/gm) * (2*pi)**2 / 386.08858 in/sec**2
$       = UNBAL * f**2 * 2.25243e-4 (lbf)
$         (where UNBAL is given in GM-IN, 'freq' in Hertz)
$       =  1.0  * (10000.*2*pi/60)^2 /453.6/386.4
$       = 6.2619 (for 10,000 RPM)
$
DAREA        301     101      1 6.256715
DAREA        302     101      2 6.256715
$
$================================================================================
$                         Structural Model
$
CONM2         99     101       0    100.                                (No Ip)
GRID         101       0     0.0     0.0     0.0       0    3456         onStat
GRID         102       0     0.0     0.0     1.0       0    3456         forSpinDir
$
$ CENTERING SPRINGS FOR SQUEEZE-FILM DAMPER
$
CELAS2       101 100000.     101       1     102       1                HorizK
CELAS2       102 100000.     101       2     102       2                VertK
$
$ Spring to ground
$
GRID         103       0     0.0     0.0     0.0       0  123456
CELAS2       111    1.+9     103       1     102       1                HorizK
CELAS2       112    1.+9     103       2     102       2                VertK
$
$ SQUEEZE-FILM DAMPER INPUT
$
$1.....12......23......34......45......56......67......78......89......910....10
NLRSFD         1     101     102      XY    6.44    .727    .003   SHORT+
+           7.-7     0.0       1     0.0   180.0                      31+
```

```
+           0.0     0.0
$
ENDDATA
```

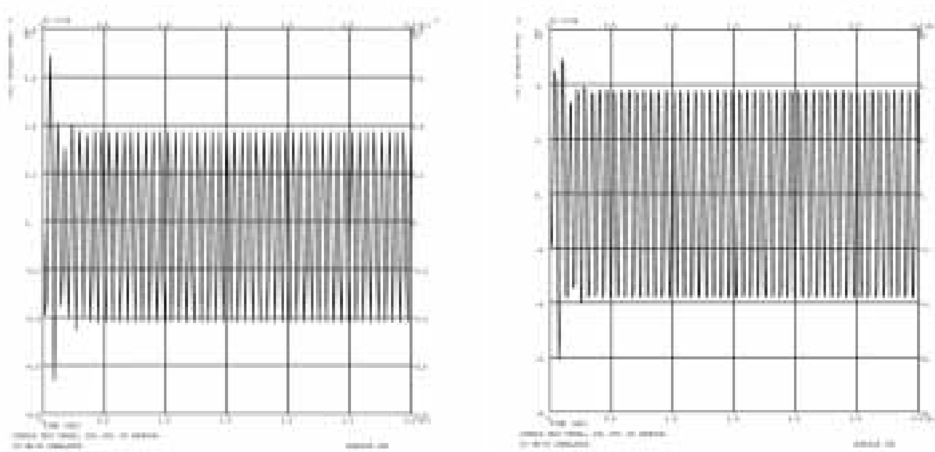


Figure 7-3 SFD Force, X and Y Direction

References

1. Adams, M. L., Padovan, J., Fertis, D. G., “Engine Dynamic Analysis With General Nonlinear Finite-Element Codes, Part 1: Overall Approach and Development of Bearing Damper Element”, ASME Journal of Engineering for Power, Vol. 104, July 1982, pp. 586-593.
2. Castelli, V., and Shapiro, W., "Improved Method for Numerical Solutions of the General Incompressible Fluid Film Lubrication Problem”, ASME Journal of Lubrication Technology, Vol. 89, No. 2, 1967, pp. 211-218.
3. Adams, M. L., Padovan, J., Fertis, D. G., “Finite Elements for Rotor / Stator Interactive Forces in General Dynamic Simulation, Part 1: Development of Bearing Damper Element”, NASA CR-165214, EDA 201-3A, October 1980.
4. Ghaby, R., “Transient / Nonlinear Vibration of Gas Turbine Engines With Squeeze Film Dampers Due to Blade Loss”, May 1984 Master of Science Thesis, Case Western Reserve University.
5. Black, G., Gallardo, V., “Blade Loss Transient Dynamics Analysis Task II-TETRA 2 User’s Manual”, NASA CR-179633, November 1986.

CHAPTER

8 DDAM Processor

- A DDAM Processor for MSC.Nastran Including an MSC.Patran Interface
- Guidelines for Effective DDAM Analysis
- Theoretical Background
- Worked Two Mass Problem
- Format of Coefficient File
- Control File Format
- User defined Shock Spectra
- MSC.Patran Interface

8.1 A DDAM Processor for MSC.Nastran Including an MSC.Patran Interface

Introduction

This chapter describes a method for the calculation of the response of a structure using the Dynamic Design Analysis Method (DDAM) in MSC.Nastran.

The Theory Section, has been implemented in MSC.Nastran and in a FORTRAN code. Due to the classified nature of specific DDAM shock environment calculations, it may be necessary to run the entire analysis, or perform that portion of the analysis on a separate computer. The DDAM package as implemented in MSC.Nastran performs the analysis in three phases as outlined in the following steps:

1. SOL 187 is used to perform a fixed-base modal analysis. Additionally, it calculates modal participation factors and modal effective mass.
2. Running the DDAM program, which accepts ASCII OUTPUT4 data from MSC.Nastran, performs shock excitation calculations using the user supplied shock coefficients and generates shock spectrum data. Additionally, it terminates the shock excitation calculations when a specified modal mass is reached. Options are provided in this step to use equations based upon the modal masses of each mode, or a user-input design spectrum to compute the shock excitations. This will be run automatically from within MSC.Nastran using the ISHELL capability unless directed otherwise.
3. Continuation of the MSC.Nastran Sequence, using output from part 2 to perform DDAM motion and stress recovery following the NRL modal summation convention. Data is calculated and output for MSC.Patran post-processing. This will automatically follow step 2 in SOL 187.

"**MSC.Patran Interface**" on page 192 describes the MSC.Patran interface that automates the process, including coefficient and file selection.

The following section describes the process in detail, with notes and documentation for all of the parameters and data required.

DDAM with SOL 187

The first step of the procedure is the calculation of the modal frequencies and participation factors that are accomplished in MSC.Nastran by SOL 187. The model must be in a "free-free" condition with the foundation degrees of freedom referenced by a SUPORT entry. The DDAM processor considers the SUPORTed degrees of freedom (and any other grids rigidly connected to them) to be a "fixed" base for the modal analysis. The information needed for the Fortran program is output from MSC.Nastran to unit 11 in ASCII format. Unit 11 will be assigned to a file using the ASSIGN statement in MSC.Nastran. The following paragraphs describe the Executive, Case Control and Bulk Data Sections of the MSC.Nastran necessary

Executive Control Data/File Management Section

The solution sequence must be SOL 187.

For example:

```
SOL 187
```

It is necessary to include ASSIGN statements to assign the plot output and DDAM output to physical files. For example:

```
ASSIGN OUTPUT2='d1.opw', UNIT=34, DELETE
```

assigns OP2 formatted post processing data to be sent to the file d1.opw. You also need to assign the NRL-summed plot data to separate units:

```
ASSIGN OUTPUT2='d1.opx', UNIT=31, DELETE
ASSIGN OUTPUT2='d1.opy', UNIT=32, DELETE
ASSIGN OUTPUT2='d1.opz', UNIT=33, DELETE
```

The following:

```
ASSIGN OUTPUT4='d1.f11', UNIT=11, FORM=FORMATTED, DELETE
```

assigns the file d1.f11 to be the DDAM output, which will be used as the input to the Fortran program. The "DELETE" qualifier tells MSC.Nastran to delete any existing versions of the file and replace them with the one generated in this submittal. The output file from the Fortran program must be assigned as well:

```
ASSIGN INPUTT4='d1.f13', UNIT=13, FORM=FORMATTED, DEFER
```

DEFER tells MSC.Nastran not to check for the existence of the file or delete an existing one.

Finally, in SOL 187, the execution of the Fortran program is directed by a control file that must be created and identified:

```
ASSIGN INPUTT4='d1.ddd', UNIT=21, FORM=FORMATTED
```

The format of this file will be discussed later.

Case Control Data

The required Case Control is similar to that required for a SOL 103 modal run, with a few caveats and exceptions. The "METHOD" command for the residual structure will determine the frequency range of the dynamic response calculations used by DDAM. Because of this, be sure to include a broad enough range to ensure that the required modal mass will be available for processing. All procedures affecting Superelement reduction to the residual degrees of freedom are allowed (i.e. component mode synthesis, GDR and Guyan reduction). However, NRL-summed results can only be generated for the residual model.

A PARAM, POST can be used with a limited number of output requests to generate data for post-processing. Case control data controls the final NRL-summed DDAM result data. Available data are STRESS, FORCE, DISPLACEMENT, VELOCITY and ACCELERATION. Other data are either not calculated in SOL 187, or are meaningless for DDAM. Mode shapes will come out by default. SOL 187 does not yet have the ability to use the XDB file for direct results access in MSC.Patran.

Bulk Data

Note: In order for units to be totally consistent in DDAM analysis, the MSC.Nastran model must be formulated in units of inches for lengths and lb-sec^2/in for mass. It is not necessary for the x, y or z axes to be correlated to any specific direction. However, the system must have individual axes that correspond to the fore/aft, athwartship and vertical directions (i.e. You cannot have a system where the basic x axis points 45 degrees between the fore/aft and vertical directions or output the whole model in a cylindrical system.)

The input data file is required to have structure geometry, property and material information (as any model should). Additionally, the following information must be present.

A SUPORT entry is necessary to define the foundation reaction point (the "fixed base") in all 3 translational degrees of freedom. If the foundation is distributed, as in most structures, rigid elements or MPCs must tie all the foundation points to a single point. This point must be in the residual structure (for Superelements) and in the A-set (this is the default for a SUPORTed degree of freedom). The translational directions of the SUPORT point define the directions for the shock response calculations, so they should be in a rectangular coordinate system. An option in the DDAM program allows you to orient the specific coordinate axes to the fore/aft, athwartship and vertical directions. Though only translational directions are used for the shock response, all 6 DOF may appear on the SUPORT entry. If one direction has no unconstrained mass (this happens a lot in test cases with few DOF), that direction should be SPCed and not called out on the SUPORT entry. If a massless direction is referenced, you get a humorous "MR MATRIX has NULL column" error.

The eigenvectors must be normalized to "MASS." This is the default for the Lanczos solver on the EIGRL entry, but not for other methods specified on the EIGR entry. "MAX" normalization will result in incorrect modal masses and participation factors, as the calculations use a shortcut for calculating the participation factors that relies on mass normalization.

Several parameters are available to control the analysis, and to direct data to its required places. There are two important and others somewhat less important parameters used in the bulk data file:

DDAM Control File

In order for MSC.Nastran to run the Fortran program correctly, it is necessary to create a small control file that tells the program some information about your model. The file is ASCII, and the format is detailed in "**Control File Format**" on page 187.

This is the file assigned to unit 21 described above.

Output

Output from SOL 187 consists of the following:

- A small ASCII file, as defined by the ASSIGN statement, for input to the Fortran program. This file contains a list of frequencies, participation factors, the total mass, and the available modal mass in each direction.
- Another small ASCII file, also defined in the ASSIGN statement, for input back into MSC.Nastran from the Fortran program.
- A verification file containing frequencies, modal masses, participation factors, and calculated accelerations.
- The .F06 file
- A .OP2 file containing mode shapes
- Three .op2 files containing the NRL-summed results

The MSC.Nastran .F06 output file will contain echoes of several matrices constructed in the DDAM process, including: (equation numbers refer to the numbers in "**Theoretical Background**" on page 172)

- OMEGX - the vector of natural frequencies in rad / sec

- PAB - The participation factor matrix [P] defined in equation 9
- MTOT - the six diagonal terms of MTOTC
- MEFF - the six diagonal terms of MEFFC
- MFRACT - the ratio of effective to total mass for each direction

OMEGX, PAB, MTOT and MFRACT are output to unit 11 by an OUTPUT4 module for use in Part 2. In addition, when the program calculates the rigid body vectors about the SUPORT point, values of Epsilon and Strain Energy are printed for each vector. These values can be used to determine if the model is over-constrained (contains constrained DOF not connected to the SUPORT point. In general, Epsilon terms should be less than 10^{-6}. Strain energy is a measure of elastic energy in the model resisting rigid body motion. Theoretically, it should be zero for DDAM, but in practice it usually has some small value. As a rule, <1 is acceptable for translations, and <100 is acceptable for rotations. If strain energies in the range of 10^4 or greater are encountered, this is an indication that a constrained degree of freedom not connected to the SUPORT entry probably exists in the model (such as an unintentionally grounded CELAS). Keep in mind, however, that these values are all model-dependent.

If the strain energies are non-zero for the first three SUPORT points (which correspond to the translational directions) it is an indication of non-singular degrees of freedom that are constrained. While MSC.Nastran allows this to take place, it violates the assumptions made in this DDAM processor, resulting in incorrect modal masses and participation factors in that direction. As a result, the final NRL results will be incorrect. To comply with the MSC.Nastran DDAM assumptions, it is necessary to connect all non-singular degrees of freedom that are to be constrained to the SUPORT point. What this means is that the only SPC entries in the model should be for singular rotational DOF. SPC entries should not be used for other reasons, such as symmetry constraints or non-moving foundation points.

Calculation of Shock Spectrum

“ddamish” (ddamish.exe on NT) is an interactive / batch program that performs shock spectrum calculation for the structure under consideration. Structural data consisting of natural frequencies and participation factors that were generated by MSC.Nastran are utilized here as input data. The program queries the analyst for spectrum inputs, loading characterization, and other details. In general, this program is run in batch mode by MSC.Nastran, with the answers supplied by the .ddd control file.

The following discusses the various prompts in the program if it is run interactively:

```
Do you have a shock spectrum or are you using
 coefficients ?
 ... <CR> = use default coefficients
 ... c    = use other coefficient file
 ... s    = user input shock spectrum
```

The program has a provision to include a set of shock coefficients compiled into the code. Hitting <CR> will use these coefficients. The default coefficients provide capability to cover the full range of Navy coefficients in DDS 072, including surface and submerged ships, deck, hull and shell mount, elastic, and plastic. The user has the option to input custom coefficients for any or all of the configurations selectively. Note that the choices are: <CR> (carriage return) for yes, “c” for coefficients stored in an external file (the most common option) and “s” to use an externally defined shock spectrum (not based on coefficients). These inputs are not case sensitive. The format for the external coefficient file is described in “**Format of Coefficient File**” on page 185, and the user spectrum format is described in “**User defined Shock Spectra**” on page 190. The external coefficient file also contains provision for a modal mass cutoff percentage and a minimum G value. If either “c” or “s” is chosen, another prompt will appear asking for the name of the external file.

```
 Are you using DDS-072 or NRL 1396 style
   equations ?
   ... <CR> = use DDS-072 equations
   ... n    = use NRL 1396 equations
```

DDS-072 and NRL 1396 differ slightly in the format of the equations and how many coefficients are required for different scenarios. This choice allows you to use either format.

```
ENTER THE DESIRED NASTRAN INPUT FILE NAME:
 Default: <CR> = ddam.f11
```

This is the .f11 file output by the MSC.Nastran run. It is necessary to type in the full file name. If the program is unable to find or open the file, a secondary prompt will inform you of that, and prompt for a new name. After successfully specifying the name, the program will echo the filename that it opened, and the unit number associated with it.

```
ENTER THE DESIRED VERIFICATION OUTPUT FILE NAME:
 Default: <CR> = d1.ver
```

This file is the intermediate result echo. This file contains the summary of modes, modal mass and mass percentages for each shock direction. The default name is the name of the .f11 file you specified, but with a .ver extension. This file is useful to verify which modes are contributing most to a model’s response.

```
ENTER THE DESIRED NASTRAN OUTPUT FILE NAME:
  Default: <CR> = d1.f13
```

This is the file that will be used as input for the MSC.Nastran restart in Part 3. The default name is constructed similarly to the verification default, but with .f13 appended to it.

```
 What type of ship do we have ?
... Enter 1 for SURFACE (default coefficients)
... Enter 2 for SUBMERGED (default coefficients)

 Where is the equipment mounted ?
... Enter 1 for DECK (default coefficients)
... Enter 2 for HULL (default coefficients)
... Enter 3 for SHELL (default coefficients)

 What type of factors do you want ?
... Enter 1 for ELASTIC (default coefficients)
... Enter 2 for EL-PL (default coefficients)
```

These three questions choose the appropriate shock coefficients. The notation “(default coefficients)” after the description indicates that the coefficients for that configuration were read from the list built into the program. If the user has specified some or all coefficients in an external file, the notation “(user coefficients)” will appear after the configuration for which coefficients were entered. The specific equations implemented in this program are documented in “**Format of Coefficient File**” on page 185.

```
ENTER F/A DIRECTION (X,Y, or Z):
ENTER VERTICAL DIRECTION (X,Y, or Z):
```

These two questions establish the coordinate directions for the application of the directional shock scaling factors. These are the AF and VF in the spectrum equations. This specification allows models built in non-standard MIL-spec coordinate directions (X forward, Y athwartship, Z up) to be processed by DDAM with the loads applied in the correct directions.

```
Is this a multiple or single mode analysis ?
   <CR> = normal, s = single mode
```

This option is provided if the user wishes to examine the contribution of a single mode to the overall summed response. Choosing the “s” option will generate a prompt asking for the mode number of interest. The program then generates a .f13 file containing factors of zero (the UHVi) for all modes except the one chosen. This will result in displacements, stresses, etc. for the model as if the entire response consisted of just a single mode. Note that this single mode will be NRL summed, so the signs will lost. For normal DDAM analysis, choose the normal option.

The PARTNVEC matrix and the BYMODE parameters provide similar capability. They are described more fully in section the following sections on special circumstances.

```
ENTER WEIGHT CUTOFF PERCENT OR <RET> FOR DEFAULT:
 Default = 80.0
```

This number is the percentage of total mass at which modal processing ceases. The DDAM document specifies that only a percentage of the total modal mass needs to be included in the NRL sum. The default value is specified in the program with the default coefficients, or in the alternate coefficient file. The value here should be entered as a percentage (i.e. 80. or 100.) not as a decimal. Specifying “100.” will process all the modes that were calculated by MSC.Nastran.

Special Circumstances – Selection of Specific Modes

If the user wishes to selectively choose which modes go into the NRL summation, a capability is provided using DMI (Direct Matrix Input) entries. On the DMI entries, the user describes a matrix called PARTNVEC, which is a partitioning vector. The vector is used to break up the eigenvector matrix and UHV matrix in a specific manner. PARTNVEC is a multi-row, 3-column matrix, where each row represents a mode number, and each column represents a shock direction. If the matrix has a “1” entry, that particular mode is retained for that particular shock direction. The following shows a PARTNVEC description employing several options that a user might be interested in for an analysis that generated 12 modes:

1	2	3	4	5	6	7	8	9
$ define the matrix as a 12 row by 3 column matrix								
DMI	PARTNVEC	0	2	1			12	3
$ define col 1 (keep all modes - 1-12)(f/a shock)								
DMI	PARTNVEC	1	1	1.	THRU	12		
$ define col 2 (keep modes 1, 2 and 3)(athw shock)								
DMI	PARTNVEC	2	1	1.	2	1.	3	1.
$ define col 3 (keep only mode 5)(vert shock)								
DMI	PARTNVEC	3	5	1.				

Be sure to ask for all the modal mass ("100.") in the DDAM program if you use this option, since it will perform its own partitioning on the output from the DDAM program.

Special Circumstances – Mode-by-Mode Output

It is occasionally desirable to look at DDAM output on a mode-by-mode basis. Such occasions can be model or methodology verification, or to gain a better understanding of which modes are contributing to particular results.

Mode-by-Mode output is controlled in MSC.Patran by the normal output requests, and by three parameters:

In SOL 187, the parameters are XBYMODE, YBYMODE, ZBYMODE. Because mode-by-mode output can generate a large volume of data, three parameters are included to handle just mode-by-mode data in specific directions. For example:

PARAM,XBYMODE,YES

Will generate data only for the X direction. In addition to the .F06 file printed output, each parameter generates a separate plot file with all the mode-by-mode results that have been requested. These three files are in units 41-43, and can be designated to specific files using an ASSIGN statement:

ASSIGN OUTPUT2='job_mbm_x.op2' UNIT=41, DELETE

Output is controlled by requests in the case control file, so if you have requested STRESS=ALL, and then request mode-by-mode data, you will get mode-by-mode stresses for all modes. Again, if this is necessary, be wary of the volume of data that can be easily produced.

Output in the .f06 file is labeled with section headers:

Each mode-by-mode section is preceded by a small header:

Important Note: Displacements, accelerations and velocities are all labeled as Eigenvectors – look at the magnitudes to make sure you have the right one.

Special Circumstances – Single shock direction calculations

If you only need output for a single shock direction, you can use= PARAM entries to skip particular directions:

PARAM XSHOCK NO
PARAM YSHOCK NO
PARAM ZSHOCK YES

If using DTI input to pick certain modes, you still have to define the 3 column matrix – this is just to cut down on post-processing output

This capability is especially useful to cut down output when using the mode-by-mode capability.

Special Circumstances – Running the Fortran part of the Analysis Separately

There are circumstances where it may be necessary to run the Fortran program separately from the MSC.Nastran job. Since the program runs automatically by default, it is necessary to add some parameters to disable the automatic running of the program. Some circumstances where you may need to run independently on MSC.Nastran:

- The Fortran program is on a classified computer, but MSC.Nastran is not
- You have a different version of the Fortran program
- You have a metric DDAM program

You can use a PARAM entry to skip the Fortran part of the run, run the Fortran elsewhere, then restart the MSC.Nastran run with the externally calculated .f13 file:

PARAM,MODEOUT,YES

This calculates modes, outputs the required DDAM data, then exits MSC.Nastran.

Use PARAM,UHVOLD,YES

This runs MSC.Nastran as usual, but does not run the Fortran program. Instead it will red a .f13 file that has been prepared. This parameter can be used on a restart from a MODEOUT run where the .f13 file was calculated externally.

It may also be desirable to replace the MSC Fortran program delivered with DDAM with a site-developed one. If so, it is necessary to replace the ddamish.exe program in the MSC.Nastran installation with your program. Because there are issues with arguments using some Fortran compilers, contact MSC for instructions on what is needed to use this approach.

8.2 Guidelines for Effective DDAM Analysis

In addition to the usual guidelines for effective structural dynamic modeling and analysis, several specific considerations are recommended for DDAM analysis. The present discussion addresses the structural modeling, foundation modeling, and data interpretation aspects of the procedure.

In the process of modeling the subject structure, a decision must be made at the outset to either employ sophisticated distributed mass modeling (the more common approach) or coarse lumped mass modeling. The latter choice is one that permits adherence to the DDAM “50% modes” criterion, and is the type of analysis for which DDAM was developed. However, fidelity of the structural model with the actual structure may be severely compromised by using this approach. This is especially true for plate and shell structures, which tend to have a significant number of shell-type modes, which are lost in a lumped mass approach. In more complicated shell structures the analysis is considered to be acceptable if enough modes are used such that at least 80% of the effective mass is accounted for. This frequently can be accomplished with significantly fewer than 50% of the modes, and usually less than 50 modes. This DDAM procedure allows the user to take either approach.

Redundant and geometrically distributed foundations require special consideration for meaningful DDAM analysis. Since the shock environment is specified at one location, distributed foundations must be referenced back to this single location. For foundations that are not “extremely” distributed, RBE2 constraints can be employed to effect the reference to the single point. On more distributed systems, however, the mass and flexibility of the foundation may be required in the structural model. In these cases, the foundation should be included in the analyzed model, and the DDAM “foundation” points will be below the actual foundation.

Resilient mounting is not normally analyzed with DDAM. In cases where equipment is resiliently mounted, the mounts are generally assumed to have bottomed out. In this case, the mount points can be considered the “foundation” points. DDAM analyses run on resiliently mounted systems are trivial exercises, since all of the modal mass is accounted for in a single rigid body mode. Flexible modes of the structure are not accounted for, and all of the loads on the actual structure are frequently zero or close to zero.

For the process of load evaluation on a structure, particular attention should be given to the mode-by-mode data provided in the verification file. For cases where forces or stresses are found to be excessive, the verification file can provide valuable guidance.

In particular, modes that contribute significantly to the overall loading can be identified. These modes can then be visualized via PATRAN, and the structure modified to change particularly detrimental modal behavior.

Another area where care must be taken are in models where significant mass is included that is not directly related to the structure of interest, and is of lower frequency. Reduction gear models that include significant runs of shafting are examples of this. In these cases, it is possible to get 80% of the modal mass entirely in shafting modes. In these cases, it may be necessary to increase the modal mass percentage to assure that you are getting 80% of the mass of the portion of model of interest.

A Note on Symmetry

While the NAVSEA 3010 document does not exclude running a symmetric model, it does not provide any guidelines to do so. There are some important considerations when using symmetry. Among them:

- The modal mass must be doubled to calculate the shock coefficients properly
- Both symmetric and anti-symmetric modes must be included in the final NRL sum
- Symmetry plane boundary points must be constrained with SPC entries (which violates the SUPORT assumption in this DDAM) or connected to the SUPORT entry with RBE elements (which incorrectly marks them as “foundation” input points)
- The MSC.Nastran NRL sum module (DDRMM) cannot combine modes from different NASTRAN runs without a modification to the DMAP alter.
- Cyclic symmetry involves calculating all of the proper harmonics, calculating the modal masses correctly and then summing all of the different harmonic modes in the final NRL sum.

Because of all these caveats, it is neither advisable nor easy to use this DDAM processor for symmetric models.

However, if it is to be used, there are several modifications that must be made to the procedure, including manually calculating modal masses and the resulting shock factors. The user should be very careful doing this. The theoretical outline in “**Theoretical Background**” on page 172 can serve as a guide. The methods should be tested on a small model to assure that all the appropriate masses, coefficients, etc are modified. Also note that the NRL sum must be either performed manually (external to MSC.Nastran) or the DMAP altered to accommodate combining results from the symmetric and anti-symmetric runs.

References:

1. R.O.Belsheim and G.J.O'Hara, "Shock Design of Shipboard Equipment, Part I, Dynamic Design Analysis Method," NAVSHIPS 250-423-30, May 1962
2. "Shock Design Criteria for Surface Ships," Naval Sea Systems Command, NAVSEA 0908-LP-000-3010 Revision 1, September 1995.
3. M.M.Hurwitz, "A revision of the Dynamic Design Analysis Method (DDAM) in NASTRAN," Naval Sea Systems Command, December 1982.
4. NAVSEA Design Data Sheet DDS-072. (Confidential)
5. Scavuzzo & Pusey, "Naval Shock Analysis and Design," Shock and Vibration Information and Analysis Center (SAVIAC), 2000

Reference 1 outlines the original concept of DDAM as applied to Naval Shipboard equipment. Reference 2 is the NAVSEA specification (latest version) that gives specifics for performing this type of analysis. Reference 3 describes the procedure for using DDAM within the framework of "modern" finite element codes. Reference 4 contains the classified NAVSEA coefficients for calculating the spectral quantities in the response equations. Reference 5 is a text that follows the shock class taught by Rudy Scavuzzo and Henry Pusey. It provides a lot of background on testing and requirements, as well as many theoretical and analytical aspects of underwater shock and DDAM.

8.3 Theoretical Background

Consider a structural system (**Figure 8-1**) described by a set of $\{U_g\}$ displacements, a subset of which corresponds to a foundation interface $\{\bar{U}_m\}$. If the foundation interface is redundant, let it be assumed that the foundation undergoes rigid body displacements (that is, no foundation warping). Thus, the foundation interface is related to a six degree of freedom (DOF) reference point displacement subset $\{U_r\}$ through a multi-point constraint relationship

$$\{\bar{U}_m\} = (\bar{G}_m)\{U_r\} \qquad \text{Eq. 8-1}$$

The remaining displacements in the $\{U_g\}$ set may be interrelated in a variety of ways, depending on the particular structure's configuration and approximating assumptions, such as Guyan Reduction or Generalized Dynamic Reduction. When all constraints and reductions are applied, the structural displacement state is described in terms of a set, denoted here as the $\{U_x\}$ set which is partitioned as follows:

$$\{U_x\} = \left\{\frac{U_l}{U_r}\right\} \qquad \text{Eq. 8-2}$$

The subset $\{U_l\}$ is comprised of discrete grid point displacements and generalized displacements remaining after reduction, depending upon the choice of approximating assumptions.

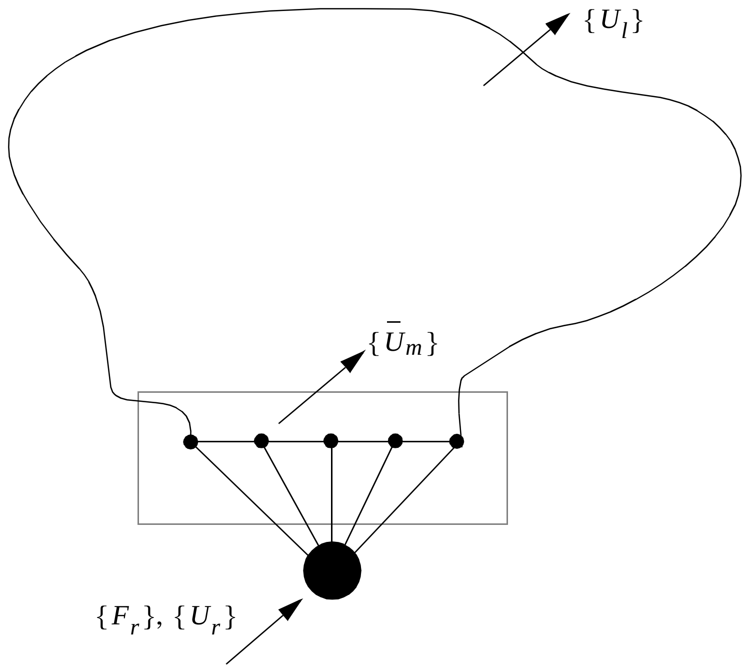


Figure 8-1

In terms of the $\{U_x\}$ displacement set, the dynamics of an undamped linear static structure subjected to foundation excitation is described by:

$$\begin{bmatrix} M_{ll} & M_{lr} \\ M_{rl} & M_{rr} \end{bmatrix} \begin{Bmatrix} \ddot{U}_l \\ \ddot{U}_r \end{Bmatrix} + \begin{bmatrix} K_{ll} & K_{lr} \\ K_{rl} & K_{rr} \end{bmatrix} \begin{Bmatrix} U_l \\ U_r \end{Bmatrix} = \begin{Bmatrix} 0 \\ F_r \end{Bmatrix} \qquad \text{Eq. 8-3}$$

Where M_{ll}, M_{lr}, M_{rl}, and M_{rr} are mass matrix partitions, K_{ll}, K_{lr}, K_{rl}, and K_{rr} are stiffness matrix partitions, and F_r is the foundation interface reaction set. Since the foundation $\{U_r\}$ is determinate due to **Eq. 8-1**, the transformation of the $\{U_x\}$ set into base fixed displacement patterns and rigid body motions, respectively, is introduced. Rigid body motions are readily defined on the basis of the stiffness matrix by imposing the requirement that the $\{U_l\}$ set produces no static reactions due to applied foundation motions $\{U_r\}$, i.e.

$$[K_{ll}]\{U_l\}_{RB} + [K_{lr}]\{U_r\} = 0 \qquad \text{Eq. 8-4}$$

or

$$\{U_l\}_{RB} = -[K_{ll}]^{-1}[K_{lr}]\{U_r\} \qquad \text{Eq. 8-5}$$

A convenient set of base fixed displacement patterns consists of a truncated set of base fixed unit mass normalized modes that are the solutions of

$$[K_{ll}]\{\phi_{li}\} = [M_{ll}]\{\phi_{li}\}\omega_i^2 \qquad \text{Eq. 8-6}$$

Assembling the truncated set of modes into the matrix $[\phi_{ll}]$, the desired variable transformation is defined as

$$\begin{Bmatrix} U_l \\ U_r \end{Bmatrix} = \begin{bmatrix} \phi_{ll} \\ 0_{rl} \end{bmatrix}\{q_1\} + \begin{bmatrix} K_{ll}^{-1} & K_{rl} \\ 1_{lr} & 1_{rr} \end{bmatrix}\{u_r\} = \begin{bmatrix} \phi_{ll} & -K_{ll}^{-1}K_{lr} \\ 0_{rl} & 1_{rr} \end{bmatrix}\begin{Bmatrix} q_1 \\ U_r \end{Bmatrix} \qquad \text{Eq. 8-7}$$

Upon transformation of the dynamic equation set, **Eq. 8-3** with **Eq. 8-7** (including pre-multiplication by the transpose of the transformation matrix, the modal equation set is of the form

$$\begin{bmatrix} I_{ll}P_{lr} \\ P_{rl}\tilde{M}_{rr} \end{bmatrix}\begin{Bmatrix} \ddot{q}_l \\ \ddot{U}_r \end{Bmatrix} + \begin{bmatrix} \omega_i^2 O_{lr} \\ O_{rl}O_{rr} \end{bmatrix}\begin{Bmatrix} q_1 \\ U_r \end{Bmatrix} = \begin{Bmatrix} 0 \\ F \end{Bmatrix} \qquad \text{Eq. 8-8}$$

The matrix partitions are defined as follows:

$$\begin{aligned}
[I_{ll}] &= \phi_{ll}^T M_{ll}\phi_{ll} \\
[\omega_l^2] &= \phi_{ll}^T K_{ll}\phi_{ll} \\
[P_{lr}] &= \phi_{ll}^T M_{lr} - \phi_{ll}^T M_{ll}K_{ll}^{-1}K_{lr} \\
[O_{lr}] &= \phi_{ll}^T[K_{lr} - K_{ll}K_{ll}^{-1}K_{lr}] \\
[\tilde{M}_{rr}] &= M_{rr} - M_{rl}K_{ll}^{-1}K_{lr} - K_{rl}K_{ll}^{-1}M_{lr} + K_{rl}K_{ll}^{-1}M_{ll}K_{ll}^{-1}K_{lr} \\
[O_{rr}] &= K_{rr} - K_{rl}K_{ll}^{-1}K_{lr} \\
[P_{rl}] &= [P_{lr}]^T \\
[O_{rl}] &= [O_{lr}]^T
\end{aligned} \qquad \text{Eq. 8-9}$$

The physical significance of the above noted matrix partitions is that $[P_{lr}]$ is the matrix of participation factors, $[\tilde{M}_{rr}]$ is the total rigid body mass referenced to $\{U_r\}$, and $[O_{rr}]$, the constraint matrix, is null due to the determinate foundation $\{U_r\}$.

For the case of DDAM requirements, response of a structure to an imposed foundation motion

$$\{\ddot{U}_r\} = \{\Gamma\}\ddot{U}_s(t) \qquad \text{Eq. 8-10}$$

is sought. The array $\{\Gamma\}$ serves as a directional vector for the applied motion history, $\ddot{U}_s(t)$ for example, if it is directed in the U_{r1} sense (x direction) then $\Gamma_1 = 1$, $\Gamma_2 = 0$, ... $\Gamma_6 = 0$. From the upper partition of **Eq. 8-8**, the modal response equation is obtained as

$$[l_{ll}]\{\ddot{q}_1\} + [\omega_1^2]\{q_1\} = -[P_{lr}]\{\Gamma\}\ddot{U}_s(t) \qquad \text{Eq. 8-11}$$

The individual mode equations are

$$\ddot{q}_i(t) + \omega_i^2 q_i(t) = -[P_{lr}]\{\Gamma\}\ddot{U}_s(t) \qquad \text{Eq. 8-12}$$

for mode "I" excited by a shock in the "j" direction. The foundation reaction forces are determined by the lower partition of **Eq. 8-8** as

$$\{F_r\} = [P_{rl}]\{\ddot{q}_l\} + [\ddot{M}_{rr}]\{\Gamma\}\ddot{U}_s(t) \qquad \text{Eq. 8-13}$$

In DDAM analysis, the second term is **Eq. 8-13** is neglected, due to the assumption of an extremely short duration applied shock, so that the $\ddot{U}_s(t)$ is zero by the time each modal response reaches its peak value. Thus, the reaction force in the "j" direction is determined by the summation of individual modal reactions F_{ij} which are

$$F_{ij} = P_{ij}\ddot{q}_i(t) \qquad \text{Eq. 8-14}$$

Consider now the response to an impulsive shock occurring over the duration $0 \le t \le t_s$ s which is much shorter than the period of any mode under consideration. Integrating **Eq. 8-12** over the shock duration results in

$$\dot{q}_i(t_s) = -P_{ij}V_{ai} \qquad \text{Eq. 8-15}$$

where the term $\int_0^{t_s} \omega_i^w q_i(t)$ is negligibly small since the modal displacement hasn't had enough time to develop; and

$$V_{ij} = P_{ij}\ddot{q}_i(t) \qquad \text{Eq. 8-16}$$

is the spectral velocity of the shock. The free response of mode "I" for $t > t_s$ is therefore

$$
\begin{aligned}
q_i(t) &= \frac{-P_{ij}V_{ai}}{\omega_i}\sin\omega_i(t-t_s) \\
\dot{q}_i(t) &= -P_{ij}V_{ai}\cos\omega_i(t-t_s) \\
q_i(t) &= P_{ij}\omega_i V_{ai}\sin\omega_i(t-t_s)
\end{aligned}
\qquad \text{Eq. 8-17}
$$

The physical significance of the structure and foundation reactions are determined by the modal superposition following **Eq. 8-7** with $Ur = 0$ for $t > t_x$ and **Eq. 8-14** respectively. Moreover, internal loads and stresses on the structure are determined by stress recovery operations for the particular model described by the generic equation

$$\{\sigma\} = [K^{\sigma}]\{U\} \qquad \text{Eq. 8-18}$$

According to NAVSEA requirements (Reference (2)), a shock spectrum summation method is adopted in which time phasing of modal responses is not considered. In this approach the peak modal responses are utilized as

$$
\begin{aligned}
q_i peak &= |P_{ij}|V_{ai}/\omega_i \\
\dot{q}_i peak &= |P_{ij}|V_{ai} \\
\ddot{q}_i peak &= \omega_i|P_{ij}|V_{ai}
\end{aligned}
\qquad \text{Eq. 8-19}
$$

Individual modal peak physical responses are governed by the relationships of the type

$$
\begin{aligned}
(\ddot{U}_{ki})_{peak} &= |\phi_{ki}|\ddot{q}_i peak \\
(F_{ji})_{peak} &= |P_{ij}|\ddot{q}_i peak \\
(\sigma_{ki})_{peak} &= \left|\sum K^{\sigma}_{kj}\phi_{ji}\right| q_i peak
\end{aligned}
\qquad \text{Eq. 8-20}
$$

It is interesting to note that substitution of the modal peak acceleration into the modal reaction equation results in

$$(F_{ij})_{peak} = |P_{ij}|^2\omega_i V_{ai} \qquad \text{Eq. 8-21}$$

where $|P_{ij}|^2$ is the modal mass, i.e.

$$M_{ai} = |P_{ij}|^2 \qquad \text{Eq. 8-22}$$

The NAVSEA modal summation convention utilized in DDAM calculations follows the generic form

$$R_j = R_{jm} + \sqrt{\sum_{\substack{i=1 \\ (i \neq m)}}^{N} |R_{ji}|^2}$$

Eq. 8-23

Where R_j is a generic response quantity and r_{jm} is the largest modal response quantity in the set r_{ji}, I=1 to N (N = number of modes). It should be noted that the index "m" is not necessarily the same for all physical responses. The current version of the DDAM program performs the NRL summation following the latest specification in Reference (**2.**). In this spec, the modes are added up in decreasing order of modal mass, rather than in increasing order of frequency. The output now gives a summation order so that the user can follow the process. Note that some modes may be included in the sum for one shock direction but not included for another direction.

Up to this point, the value of the modal spectral velocities, V_{ai}, have not been discussed. Due to the flexibility and finite mass of the ship structure onto which the structural system is mounted, the value of V_{ai} is an empirical function of the modal effective mass, M_{ai}, and modal frequency, ω_I. The actual formulae for the spectral quantities are presented in Reference (**4.**), and the form of the equations are presented in "**Format of Coefficient File**" on page 185.

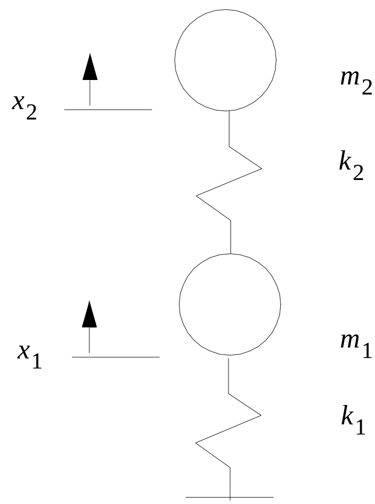
An interesting property of modal effective mass is that the sum of the individual terms, M_{ai}, will approach the total actual mass of the system as the number of modes, N, approaches the complete set for the mathematical model. The summation will exactly approach the total system mass if no mass is allocated to the foundation degrees of freedom $\{\tilde{U}_m\}$ or $\{U_r\}$. If mass is allocated to the foundation, the summed effective mass will approach the total mass minus the foundation mass. For DDAM analyses, NAVSEA allows utilization of a truncated mode set, which provides a minimum of 50% of the total number of modes of the system. Because this requirement is excessive for modern large-scale finite element models described by thousands of degrees of freedom, NAVSEA accepts analyses that contain at least 80% of the model's effective modal mass. The number of modes required to achieve this amount is highly model dependent, but frequently involves less than one hundred modes, compared to thousands required by the 50% criterion.

8.4 Worked Two Mass Problem

Problem:

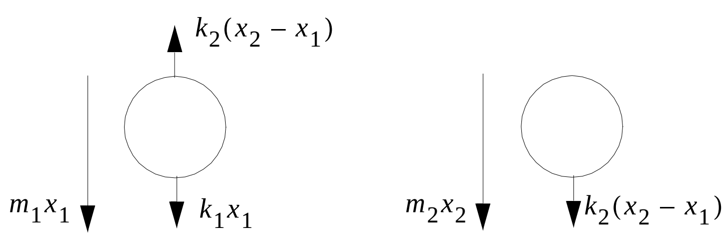
Consider the following arrangement:

$$\begin{aligned} m_1 &= 8000\,\text{lb} = 20.70\ \text{lb-sec}^2/\text{in} \\ m_2 &= 6000\ \text{lb} = 15.53\ \text{lb-sec}^2/\text{in} \\ k_1 &= 500,000\ \text{lb/in} \\ k_2 &= 200,000\ \text{lb/in} \end{aligned}$$



Step 1 - Calculate Natural Frequencies

Equation of motion from free body diagrams:



Summation of Forces:

$$\sum F_1 = 0. = m_1\ddot{x}_1 + k_1x_1 - k_2(x_2 - x_1)$$
$$= m_1x_1 + (k_1 + k_2)x_1 - k_2x_2$$

$$\sum F_2 = 0. = m_2\ddot{x}_2 + k_2(x_2 - x_1)$$
$$= m_2\ddot{x}_2 + k_2x_2 - k_2x_1$$

or, in the more familiar matrix form:

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{Bmatrix} \ddot{x}_2 \\ \ddot{x}_2 \end{Bmatrix} = \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

Step 2 - Solve the Equations

This can be done manually or using a computer, but the end result is the same.

Solution of these yields two modes with natural frequencies:

$$\omega_1 = 89.72 \text{ rad/sec} = 14.280 \text{ Hz.}$$
$$\omega_2 = 196.6 \text{ rad/sec} = 31.285 \text{ Hz.}$$

and corresponding eigenvectors. MSC / DDAM requires these to be "mass" normalized, which can be accomplished by dividing each "max" normalized value by the generalized mass $[M_g] = \{\phi\}^T[m]\{\phi\}$. Note that the generalized mass for a mass normalized vector set will be 1 for all modes. The mass normalized eigenvectors will be:

$$X_1 = \begin{bmatrix} .0873 \\ .2329 \end{bmatrix} \qquad \bar{X}_2 = \begin{bmatrix} .2017 \\ -.1008 \end{bmatrix}$$

Step 3 - Calculate the Participation Factors

They are determined from the following equations. Note that the participation factors are normalization dependent. Also note that MSC.Nastran may have reversed the signs on the second eigenvector. This simply changes some signs and the magnitudes of the participation factors in the intermediate steps. The final solution is independent of the normalization.

$$P_a = \frac{\sum \bar{X}_{ia} m_i}{\sum (\bar{X}_{ia})^2 m_i} \quad \bar{X}_{is} = \text{eigenvectors, } m_i = \text{individual masses}$$

$$P_i = \frac{\bar{X}_{11} m_1 + \bar{X}_{21} m_2}{(\bar{X}_{11})^2 m_1 + (\bar{X}_{21})^2 m_2} = \frac{(.0873)(20.70) + (.2329)(15.53)}{(.0873)(20.70) + (.2329)^2(15.53)} = 5.423$$

$$P_{21} = \frac{X_{12} m_1 + X_{221} m_2}{(\bar{X}_{12})^2 m_1 + (\bar{X}_{22})^2 m_2} = \frac{(.2017)(20.70) + (-.1008)(15.53)}{(.2017)^2(20.70) + (-.1008)^2(15.53)} = 2.610$$

Step 4 - Calculate the Modal Masses

Unlike the participation factors, the modal masses are not normalization dependent. Note that the modal “masses” are really weights.

$$M_a = \frac{(\sum \bar{X}_{ia} m_i)^2}{\sum (\bar{X}_{ia})^2 m_i}$$

$$M_1 = \frac{(\bar{X}_{11} m_1 + \bar{X}_{21} m_2)^2}{(\bar{X}_{11})^2 m_1 + (\bar{X}_{21})^2 m_2} = \frac{[(.0873)(20.70) + (.2329)(15.53)]^2}{(.0873)^2(20.70) + (.2329)^2(15.53)} = 11{,}368.\text{lb}$$

$$M_2 = \frac{(\bar{X}_{12} m_1 + \bar{X}_{221} m_2)^2}{(\bar{X}_{12})^2 m_1 + (\bar{X}_{22})^2 m_2} = \frac{[(.2017)(20.70) + (-.1008)(15.53)]^2}{(.2017)^2(20.70) + (-.1008)^2(15.53)} = 2{,}631.\text{lb}$$

Step 5 - Choose the Shock Coefficients

We will choose “surface ship,” “deck” inputs. These correspond to the coef.dat file in the sample directory. They do not represent any real spectrum, and only serve to demonstrate the DDAM solution methodology.

AA = 50.	VA = 120.
AB = 40.	VB = 50.
AC = 10.	VC = 10.

F/A factor = 1.0
Athwartship factor = 1.0
Vertical factor = 1.0

Step 6 - Calculate the Spectrum Inputs

The equations found here are of the same form as some formal specifications. The values, however, are simply for demonstration purposes. Note that the W in the equations represents the weight (not mass) in 1000s of pounds (kips). Also note that the equations deliver the accelerations in Gs and the velocity in in/sec. These conventions are hard coded into the ddam.f program.

Mode 1:

$$A_0 = \frac{AA(AB + M_1)}{M_1 + AC} = \frac{50(40 + 11.368)}{11.368 + 10} = 120\,\text{g}$$

$$V_0 = \frac{VA(VB + M_1)}{M_1 + VC} = \frac{120(50 + 11.368)}{11.368 + 10} = 344.6\ \text{in/sec}$$

$$A_v = \frac{V_0\omega}{g} = \frac{344.6(89.72)}{386.4} = 80.0\ \text{g}$$

Since A_v is less than A_0, use A_v for the calculation.

Mode 2:

$$A_0 = \frac{AA(AB + M_1)}{M_1 + AC} = \frac{50(40 + 11.368)}{11.368 + 10} = 120\,\text{g}$$

$$V_0 = \frac{VA(VB + M_1)}{M_1 + VC} = \frac{120(50 + 11.368)}{11.368 + 10} = 344.6\ \text{in/sec}$$

$$A_v = \frac{V_0\omega}{g} = \frac{344.6(89.72)}{386.4} = 80.0\ \text{g}$$

This time, A_v is larger than A_0, so we use A_0 for the calculations here.

Step 7 - Use the Accelerations, Eigenvectors, Individual Weights, and Participation Factors to Find the Dynamic Forces on the Masses.

The weights are the individual weights, not masses or modal masses.

$$F_{ia} = W_i\bar{X}_{ia}P_aA(\omega_a)$$

Mode 1:

$$F_{11} = W_1 \bar{X}_{11} P_1 A(\omega_1) = (8000)(.0873)(5.423)(80.0) = 302{,}994.\text{ lb}$$

$$F_{21} = W_2 \bar{X}_{21} P_1 A(\omega_1) = (6000)(.2329)(5.423)(80.0) = 606{,}248.\text{ lb}$$

Mode 2:

$$F_{12} = W_1 \bar{X}_{12} P_2 A(\omega_2) = (8000)(.2017)(2.610)(169.) = 711{,}\ 743.\text{ lb}$$

$$F_{22} = W_2 \bar{X}_{22} P_2 A(\omega_2) = (6000)(-.1008)(2.610)(169.) = -266{,}771.\text{ lb}$$

Step 8 - Use the Mass Forces to Get the Forces in the Springs:

Mode 1:

Spring 1 = 303.0 + 606.2 = 909,200 lb

Spring 2 = 606.2 = 606,200 lb

Mode 2:

Spring 1 = 711.7 - 266.8 = 444,900 lb

Spring 2 = -266.8 = 266,800 lb

Step 9 - Perform the NRL Sum of the Spring Forces.

Note that the NRL sum reduces to a trivial summation for this 2-mass case.

Spring 1 = 909,200 + 444,900 = 1,354,100 lb

Spring 2 = 606,200 + 266,800 = 873,000 lb

Step 10 - Calculate the Nodal Displacements and Velocities.

Since the deflection of each mode is an orthogonal mode shape, the displacements, velocities and accelerations are related by:

$$A = V\omega = x\omega^2$$

The accelerations can be simply calculated from:

$$A_{ia} = \frac{F_{ia}}{W_i}$$

Mass 1, Mode 1:

$$A_{11} = \frac{F_{11}}{W_1} = \frac{302994.}{8000} = 37.87 \text{ g}$$

$$V_{11} = \frac{A_{11}}{\omega_1} = \frac{37.87(386.4)}{89.72} = 163.1\frac{in}{\text{sec}}$$

$$x_{11} = \frac{A_{11}}{\omega_1^2} = \frac{37.87(386.4)}{(89.72)^2} = 1.818 \text{ in}$$

Mass 2, Mode 1:

$$A_{21} = \frac{F_{21}}{W_2} = \frac{606248.}{6000} = 101.04 \text{ g}$$

$$V_{21} = \frac{A_{21}}{\omega_1} = \frac{101.04(386.4)}{89.72} = 435.2\frac{in}{\text{sec}}$$

$$x_{21} = \frac{A_{21}}{\omega_1^2} = \frac{101.04(386.4)}{(89.72)^2} = 4.850 in$$

Mass 1, Mode 2:

$$A_{12} = \frac{F_{12}}{W_2} = \frac{711743.}{8000} = 88.97 g$$

$$V_{12} = \frac{A_{12}}{\omega_2} = \frac{88.97(386.4)}{196.6} = 174.9\frac{in}{\text{sec}}$$

$$x_{12} = \frac{A_{12}}{\omega_2} = \frac{88.97(386.4)}{(196.6)^2} = .8894 \text{ in}$$

Mass 2, Mode 2:

$$A_{22} = \frac{F_{22}}{W_2} = \frac{-266771}{6000} = 44.46 g$$

$$V_{22} = \frac{A_{22}}{\omega_2} = \frac{44.46(386.4)}{196.6} = 87.38\frac{in}{\text{sec}}$$

$$x_{22} = \frac{A_{22}}{\omega_2^2} = \frac{44.46(386.4)}{(196.6)^2} = .4445 \text{ in}$$

Step 11 - NRL Sum the Velocities, Displacements and Accelerations

Like the forces, this is a trivial summation for this sample problem.

Mass 1:

$x = 1.818 + .8894 = 2.707$ in

$V = 163.1 + 174.9 = 338.0$ in / sec

$A = 37.87 + 88.97 = 126.84$ g

Mass 2:

$x_2 = 4.850 + .4445 = 5.295$ in

$V_2 = 435.2 + 87.38 = 522.6$ in / sec

$A_2 = 101.04 + 44.46 = 145.50$ g

This sample problem is provided as the d1 model in the sample files. In the output note that directions 2 and 3 are all 0, as those degrees of freedom were constrained out. Only the X direction results and X directed shock (F / A) have any meaning.

8.5 Format of Coefficient File

The DDAM coefficient file contains the weighting factors used for the response calculations, the directional scaling factors, as well as the modal mass cutoff value. The file is structured as shown below. The default equations to which these apply are: (M is the modal weight in kips for that mode)

$$A_0 = AF\frac{AA(AB+M)}{AC+M} \qquad V_0 = VF\frac{VA(VB+M)}{VC+M}$$

For surface ship, hull and shell mount, the equation is:

$$A_0 = AF\frac{AA(AB+M)(AC+M)}{(AD+M)}$$

There is a complete set of AA, AB, AC, and (when needed) AD weighting factors for each of the possible analysis configurations (surface ship and submerged ship, deck mount, shell mount and hull mount). In addition, there is a trio of AF (acceleration factors) and VF (velocity factors) for each of these sets, one for each shock direction. There are additional factors for Elastic-Plastic design.

The file is formatted sort of like an MSC.Nastran file. Each set of coefficients and factors are entered on a COEF entry that describes the applicability of that set of factors. The COEF entry is formatted like a NASTRAN statement - i.e. ten eight character fields. The entry looks like:

1	2	3	4	5	6	7	8	9	10
COEF	nsurf	nstruc	nplast						
	VF(1)	VF(2)	VF(3)	AF(1)	AF(2)	AF(3)			
	VA	VB	VC	AA	AB	AC	AD		

nsurf — ship type. Allowable values are SUB (submerged) and SURF (surface ship)

nstruc — mounting location. Allowable values are DECK, HULL, and SHELL

nplast — elastic or elastic-plastic factors. Allowable values are ELASTIC and ELPL.

The (i) in the VF and AF refer to the directions: (1)=fore/aft, (2)=athwartship, and (3)=vertical

A blank entry or a * entry in any field will use the default value (from the program source) for that value.

In addition to the COEF entry defining coefficients, there is a CUTOFF entry that defines the modal mass cutoff percentage. That entry looks like:

```
CUTOFF  pref
```

pref is the cutoff weight percentage for the modal mass calculation. Enter as a percentage, not a decimal fraction (i.e. 85. instead of .85). Note that you still have the option of overriding this value when you run the program.

pref 100. Use all available modes.

nn. Make spectral values zero for all modes beyond the one that first exceeds nn percent of the total mass.

A sample coef.dat file to analyze different surface ship equipment using the elastic-plastic factors might look like:

```
# special elastic-plastic surface ship factors
# - deck coefficients
COEF    SURF    DECK    ELPL
        .25     .50     1.0      .25     .50     1.0
        10.     20.     50.      10.     37.5    6.
# - hull coefficients
COEF    SURF    HULL    ELPL
        .30     .60     1.0      .25     .50     1.0
        5.      10.     40.      10.     45.5    6.5    15.
# - shell coefficients
COEF    SURF    SHELL   ELPL
        .25     .50     1.0      .25     .50     1.0
        *       *       *        10.     45.5    6.5    15.
```

It is important that all fields be 8 characters (or spaces) long, as there is a bug in the read routine (that should have been fixed for MSC.Nastran 2005) that requires this. This is easily achieved by padding the ends of lines with blanks to achieve the full 8-character length.

8.6 Control File Format

The control file is simply a list of responses to the questions that the DDAM Fortran program asks. The format can be any one of three, depending on which user options are being requested.

No special user options:

```
F F T
nsurf nstruc nplast
pref
amin
f/a_axis vert_axis
.f11 filename
.f13 filename
.ver filename
```

User coefficient option:

```
T F T
coef.dat filename
nsurf nstruc nplast
pref
amin
f/a_axis vert_axis
.f11 filename
.f13 filename
.ver filename
```

User spectrum Option:

```
F T T
spec.dat filename
pref
amin
f/a_axis vert_axis
.f11 filename
.f13 filename
.ver filename
```

Specific file formats as follows:

- First Line – spectrum control – format a1,1x,a1,1x,a1
    - First item – DDAM or general spectrum run flag

      T = General non-DDAM spectrum run

      F = DDAM
    - Second item – Coefficients from File or form compiled source

      T = coefficients from external file

F = use built-in coefficients

Ignored if first item is T

- Third item – Equation format

  T = DDS-072 style equations

  F = NRL 1396 style equations

  Ignored if first item is T

- Second Line – file name (if needed) – format a80
  - If 1st item on line 1 is T

    Name of spectrum file
  - If 2nd item on line 1 is T

    Name of coefficient file
  - If neither are T, line is not needed
- Third Line – location flags – format i1,1x,i1,1x,i1
  - First item – Surface or Submarine

    1 = Surface

    2 = Submarine
  - Second item – equipment location

    1 = Deck

    2 = Hull

    3 = Shell
  - Third item – coefficient class

    1 = Elastic

    2 = Elastic / Plastic
- 4th Line – Weight cutoff percentage – format F8.3
  - Cutoff percentage (0. To 100.)
- 5th Line – Minimum G cutoff – format F8.3
  - Minimum G level to use (in Gs)
- 6th Line – Axis Orientation – format a1,1x,a1
  - First item – F / A axis

X, Y, or Z

- Second item – Vertical Axis

  X, Y, or Z

- 7th Line – Input file – format a80
  - Name of file (full path if needed)
- 8th Line – Output file – format a80
  - Name of file (full path if needed)
- 9th Line – verification file – format a80
  - Name of file (full path if needed)
  - These names will be ignored if they are passed as arguments when the DDAM program is run.

Note that the spacing of the first line and the axis definition line are important, as are the capitalization. The first line must be in the FORTRAN format (a1, 1x, a1, 1x, a1), with the T or F capitalized. The axis line is the same format, but with X, Y, or Z for each term. A sample file for a conventional analysis might look like:

```
F F T
1 1 1
100.
1.0
X Z
d1.f11
d2.f11
d1.ver
```

8.7 User defined Shock Spectra

This section describes the program package that allows for defining a user input shock spectrum. Not to be confused with the user input shock coefficients, this routine allows the user to completely define the spectrum as data pairs of frequency and some motion quantity (displacement, velocity or acceleration). The user can define the spectrum in selected units, as outlined in the following section. The frequency scale and / or the disp / vel / accel scale can be either logarithmic or linear, as frequency and value ranges often cover several orders of magnitude.

The data is entered into a user created file, which can have any arbitrary name. In general, the file is fairly simple:

```
# = comments, anywhere in file
DATATYP type    data1   dir     freq    interp
BEGIN DATA
f_1, data_1
f_2, data_2
...
f_n, data_n
[BEGIN DATA]
[f_1, data_1]
[f_2, data_2]
[...]
[f_n, data_n]
[BEGIN DATA]
[f_1, data_1]
[f_2, data_2]
[...]
[f_n, data_n]
END FILE
```

The data on the DATATYP entry are as follows:

type — describes what motion quantity is described in the following data. Type can be one of the following: DISP, VELO, or ACCE.

data — describes the units that the motion is described in. Data can be one of the following: G (acceleration data in Gs), F (Displacement, Velocity or Acceleration data in feet, ft / sec, or ft / sec^2), I (displacement, velocity or acceleration in inches, in / sec or in / sec^2), or M (displacement, velocity or acceleration in meters, m / sec or m / sec^2)

dir	describes how many spectra are in the file. dir can be 1 (a single spectrum will be used for all three shock directions), or 3 (there are three spectra in this file, one for each direction)
freq	describes the units for the frequency terms. Choices are RAD (radians) or HERTZ (frequency in hertz).
interp	describes the axis / plot type. Data can be any one of the following: LOGLOG (both axes are logarithmic), LINLIN (neither axis is logarithmic), LOGLIN (the frequency axis is logarithmic, the other is not), or LINLOG (the frequency range is linear, the other is logarithmic).

The BEGIN DATA entry precedes each frequency / motion data section. If dir=1, there will be only one BEGIN DATA entry, if dir=3, there will be three, one preceding each section.

The data section is ended with the END FILE entry. This is included to remove the machine-specific vagaries of reading to the end of a file.

FILES CONTENTS AND ENTRY NAMES ARE CASE SENSITIVE - USE CAPITAL LETTERS!

Sample user spectrum data file:

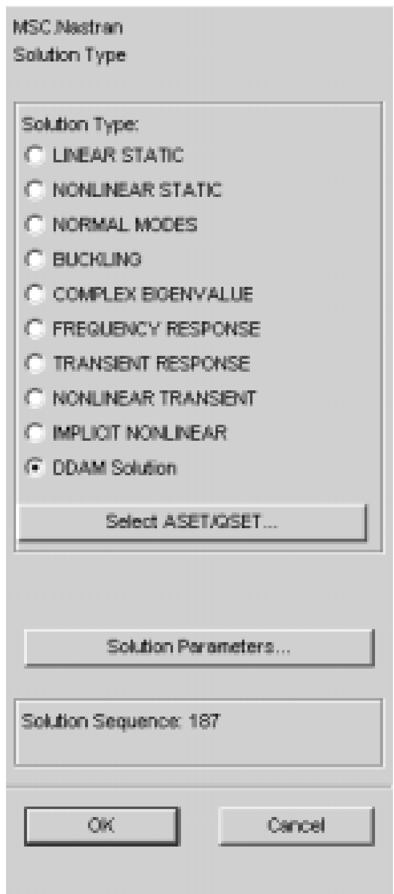
```
# sample spectrum file
#       2       3       4       5       6
DATATYP ACCE    I       1       HERTZ   LOGLIN
# acceleration vs frequency file - acceleration in in/sec**2
BEGIN DATA
1. 1.
10. 10.
100. 1000.
# point added to define range
500. 800.
1000. 500.
END FILE
```

8.8 MSC.Patran Interface

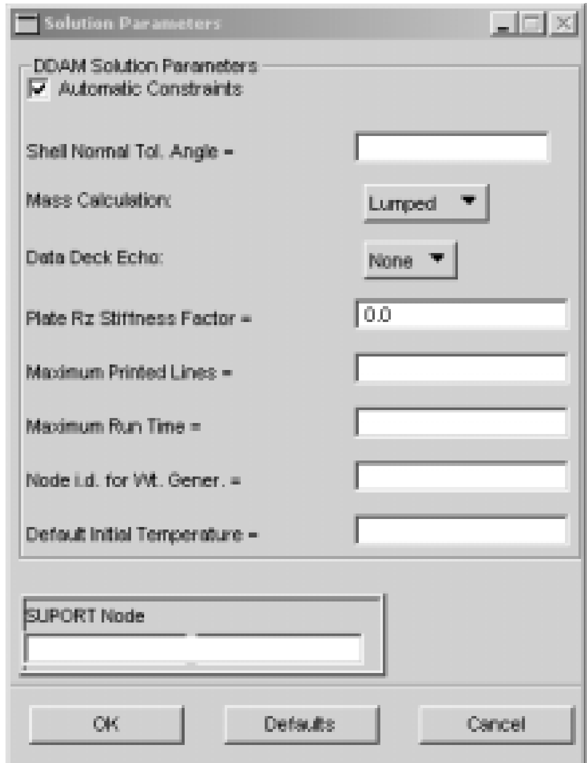
Starting in Version 2004, MSC.Patran has an analysis option that enables you to set up and run a DDAM analysis. The MSC.Patran interface will create the control file, write the file assignments, and allow entry of the SUPORT card. Any type of DDAM analysis can be run from within MSC.Patran, including the coefficient runs and user-input spectrum runs. The program performs several checks on the data that is entered to prevent the user from accidentally entering bad data and then trying to run the model.

Program Operation:

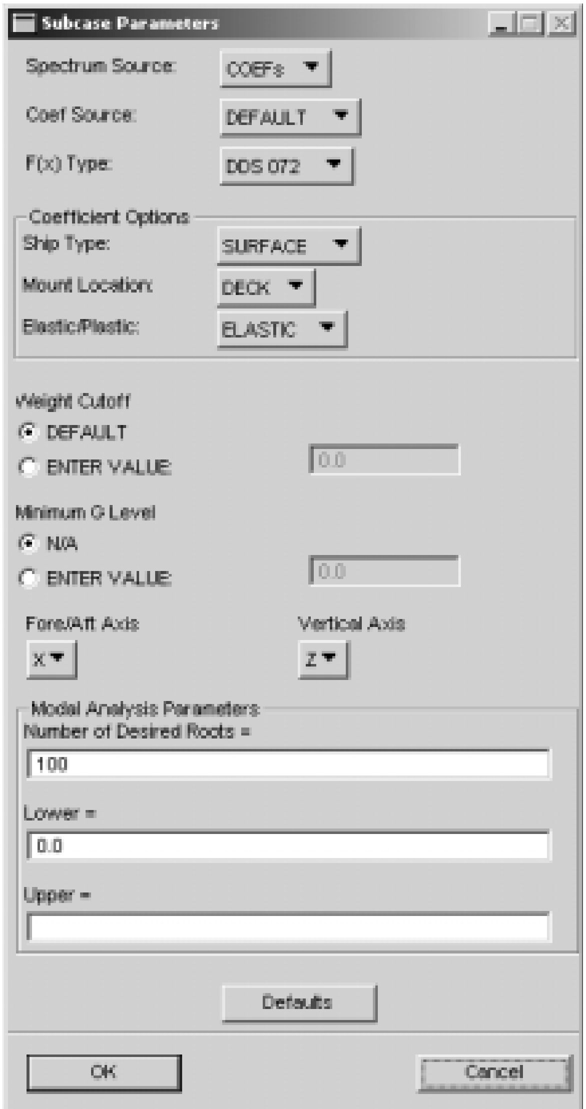
Choosing the “Solution Type” from the main analysis menu will bring up the following form:



The “Solution Parameters” button brings up the following form:



On the Subcase parameters form, most of the rest of the DDAM input is entered. That form looks like:



The “Spectrum Source” option menu has two choices, COEFs and FILE.

- “COEFs” will instruct DDAM to obtain the shock spectrum and spectral accelerations from a set of coefficients.
- “FILE” allows input of a general spectrum not determined by coefficients. If “FILE” is chosen, the file can be chosen using the “Spec Source…” button that will appear.

The “Coef Source” option menu dictates where the coefficients are to be found.

- "DEFAULT” will use the coeffiecients that are hard coded into the DDAM program.
- “FILE” will pull them from a file previously created by the user, who must then pick the file using the “Coef File…” button that will appear.

If the Spectrum Source is “FILE”, then the “Coefficient Options” items are enabled. Each menu has several choices that will allow the user to choose which set of coefficients is desired. If the coefficients are coming from a file, and the file does not contain coefficients for the particular configuration that has been specified, the default coefficients will be used. At the moment, no warning of this is given.

“Ship Type” can be “SURFACE” or “SUBMERGED”

“Mount Location” can be “DECK,” “HULL,” or “SHELL.”

“Elastic / Plastic” can be “ELASTIC” or “PLASTIC” reflecting the use of the elastic design coefficients, or the elastic / plastic design coefficients.

“Weight Cutoff” controls how many modes are used for the NRL sum. All modes up to the specified modal mass percentage specified will be included in the NRL sum for each direction. The “DEFAULT” switch will use the percentage that is hard coded into the DDAM program. If the “ENTER VALUE” switch is chosen, the cutoff should be entered as a percentage, not a decimal (e.g. 90. instead of .90).

“Minimum G Level” controls whether the calculated spectra values should be replaced with a minimum if they fall below a certain threshold. The “N / A” switch will use the value that is calculated, regardless of its magnitude. If the “ENTER VALUE” switch is chosen, the minimum cutoff should be entered as a G value (e.g 1.0. instead of 386.4).

The two axis toggles tell DDAM which direction of the model is oriented in the fore / aft and vertical directions. Each button has possible choices of “X,” “Y,” and “Z.” An error will be issued if both axes are set to be the same.

CHAPTER 9

Miscellaneous

9.1 MSC.Nastran ADAMS Integration

Overview

An additional capability has been added to allow the user to create a MSC.ADAMS modal neutral file (MNF) that does not contain any modal data. The purpose for this would be to aid model checkout, and to determine the location of attachment points. A RIGID option has been added to the mass invariant MINIVAR describer on the ADAMSMNF Case Control command:

ADAMSMNF FLEXBODY=YES, MINVAR=RIGID

Also, PARAM,AUTOQSET,YES can be used with this option that allows the required SPOINTs and QSETs to be supplied by the program automatically. This parameter should specified in the Case Control Section, above Subcase level.

Limitations

This option will work for a residual structure only model.

If PARAM,AUTOQSET,YES is specified to automatically generate SPOINT and QSET entries, then there should be no SPOINTs or QSETs present in the bulk data.

In order to determine the location of attachment points, there should be no SPOINTs or QSETs present in the bulk data, and PARAM,AUTOQSET,YES should not be present in the Case Control Section.

9.2 Alternative Solution Algorithms for Flutter Analysis

Introduction

Two new flutter solution algorithms are available in MSC.Nastran 2005. These two methods complement the existing PK,K,K-E and PKNL methods, and are referred to as PKS and PKNLS. The S signifies ‘sweep’ and is meant to indicate that these methods use a sweep technique to determine the flutter eigenvalues. By contrast, the PK and PKNL methods employ an iterative approach that relies on roots found at one estimated k (reduced frequency) value to estimate the roots at the next estimated k. This iterative process sometimes encounters a flutter analysis task that cannot be solved completely so that only a limited set of results are obtained. When this occurs, a message is printed:

```
**** (USER WARNING MESSAGE 4581 (FA1PKE)
     PK FLUTTER ANALYSIS FAILED TO CONVERGE FOR LOOP xx, ROOT yy
```

Figure 9-1 shows a comparison of extracted and estimated reduced frequency values and shows how the iterative scheme can break down. It is seen that most of the estimated roots line up in straight lines that are almost invariant with respect to the estimated frequency. However, one root starts at a kext of 3.0 and falls rapidly to kext = 0.0, crossing the 45 degree line near kext=1.0. It is this root that gives the **P-K** algorithm trouble since its order changes as kest increases, violating an assumption of the algorithm.

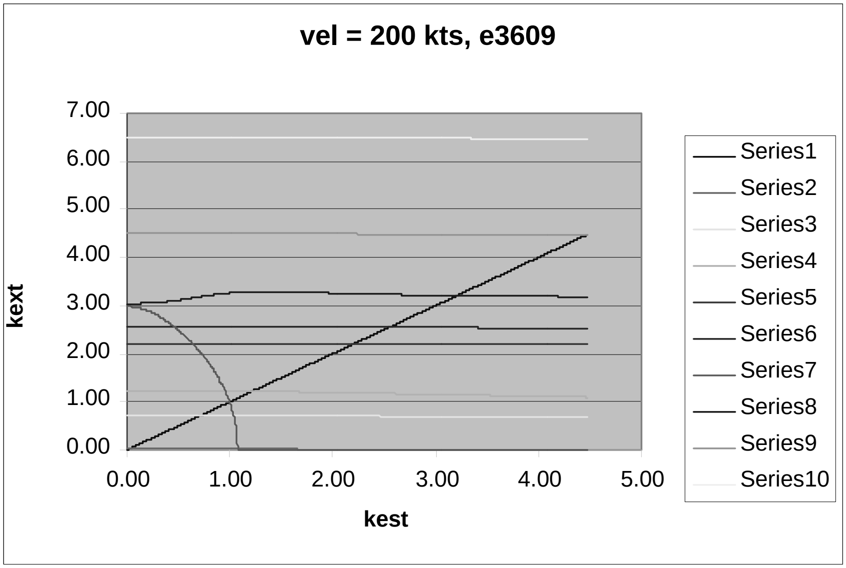


Figure 9-1 Reduced Frequency Sweep for A Test Deck Results in a Failure to Converge Message with the PK Method.

The PKS method simply sweeps across the k-range with a series of complex eigenanalyses at each of the estimated k-values. A determination is made as to when the 45 degree line is cross and the corresponding flutter root is stored.

Benefits

The FAILURE TO CONVERGE message is no longer issued, while the other benefits of the PK method (ability to deal with real roots, better estimate of complex roots, and use in optimization) are retained.

Input

The existing FLUTTER Bulk Data interface has been enhanced with additional input as given by the following table (with the modified inputs indicated in bold):

1	2	3	4	5	6	7	8	9	10
FLUTTER	SID	METHOD	DENS	MACH	VEL	IMETH	OMAX/N VALUE	**EPS**	
FLUTTER	10	PKNLS	1	2	3	L.			

Field	Contents
METHOD	PKS and PKNLS augment the existing K, KE, P-K and PKNL methods. See Remark 9.
OMAX	The maximum frequency for the frequency sweep in Hertz. (Real > 0.0), See Remark 10.
EPS	The inverse of the number of equal reduced frequency steps used in the frequency sweep. (Real > 0.0, Default = .01)

Remarks:

9. The **PKS** and **PKNLS** methods determine the flutter eigenvalues by performing a sweep of reduced frequencies ranging from kest = 0.0 through kest = π CREF OMAX/ Velocity.
10. OMAX specifies the maximum frequency in Hz. If this field in an integer, it corresponds to the current NVALUE parameter that provides the number of eigenvalues to be extracted. If the field is blank, the default is the number of modal degrees of freedom in the flutter analysis.

Outputs

The output from the PKS and PKNLS methods are quite similar to those of the PK and PKNL methods. The METHOD selected is printed in the flutter summary. DIAG 39 can be used display the eigenroots for each estimated k value, but this will produce a large amount of output.

In SOL 200, formatted prints of the density value are now the actual density value rather than the density ratio.

Guidelines and Limitations

- It is suggested that the PK and PKNL methods be the primary flutter algorithms with PKS and PKNLS reserved for cases where the former methods do not appear to be performing well.
- An attribute of the new methods is that they can produce more roots at a given velocity than there are normal modes. This is deemed a valid result but is counter to most current flutter methods, including those used in MSC.Nastran. This can produce a difficult sorting task when more (or fewer) roots are extracted at one velocity than were extracted at an earlier velocity. MSC.Nastran attempts to cope with this and puts the new roots at POINTs above those from of the previous velocities.

- The PKS and PKNLS methods can be applied in SOL 200
- DIAG 39 can be used in the PK and PKNL methods to provide insight into the iterative root extraction process. Additional output is available with PKS and PKNLS as well, but it can be voluminous for most problems.

Example (pkswep.dat)

This is the HA145A example of the *MSC.Nastran Aeroelastic User's Guide* with a single subcase applying the PKS method. The FLUTTER Bulk Data entry in this case is:

1	2	3	4	5	6	7	8	9	10
FLUTTER	3	PKS	1	2	3	L	5.	.01	

The PKS method is selected and a maximum frequency of 5.0 Hz. is used in the sweep and the sweep region is divided in 100 equally spaced frequencies ranging from 0.0 to 5.0 HZ.

The output flutter summary is headed by:

```
                                                   FLUTTER  SUMMARY
                         CONFIGURATION = AEROSG2D     XY-SYMMETRY = ASYMMETRIC     XZ-SYMMETRY =
ASYMMETRIC
       POINT =    1     MACH NUMBER =  0.0000     DENSITY RATIO =  1.0000E+00     METHOD = PKS
```

Where the METHOD=PKS indicates that the PKS method has been selected.

9.3 Little - Big Endian

Variable Endian OUTPUT2 and OUTPUT4 Files

Introduction

Major MSC.Nastran customers typically use the program in batch mode, on a remote mainframe computer, or cluster, requiring transfer of the model and results data between the remote machine and the local workstation. The amount of results data is often significant, so binary file formats are preferable with more efficient data storage and access. However, different types of machines have different formats, and so transferring data from one format to another involves a process of “transmitting” (from the remote machine) to a neutral format, copying the neutral format results data over to the local workstation, and “receiving” (to the local workstation) to rebuild a compatible binary file. Such a process is cumbersome and requires large amounts of disk space and can lead to reduced accuracy through loss of precision.

Benefits

MSC.Nastran has been enhanced to allow the user to specify the format in which binary OP2 and OP4 (OUTPUT2 and OUTPUT4) files are generated, regardless of the computer platform on which MSC.Nastran is running (note that this new capability will not be available on Cray UNICOS). Therefore, a user running MSC.Nastran on a platform such as Linux/i386 (the source machine) can request that a generated OP2/4 file be suitable for a platform such as IBM AIX or Hewlett-Packard HP-UX (the target machine). This allows the OP2/4 file to be used by post-processing programs running on the target machine directly, without having to go through a TRANS/RECEIVE data transfer process. This increases ease of use, reduces disk requirements, and overall processing time.

The target machine specification is entered through new options for the FORM qualifier on the OUTPUT2 and OUTPUT4 ASSIGN statements in the File Management Section of the input bulk data.

The benefits for OUTPUT4 files are even greater. MSC.Nastran on all platforms (except Cray UNICOS) will be able to read binary OUTPUT4 files from any platform (except Cray UNICOS) directly, and without the need for any intermediate translation. Also, for users who read OUTPUT4 files into their own programs, a translation program will be available that will allow binary OUTPUT4 files to be copied and transformed from one format to another.

Theory

The term "Endian" refers to the byte ordering for numeric data used by a particular computer architecture. "Big Endian" specifies that the most significant byte (MSB) of a data element is stored at the lowest byte address, while "Little Endian" specifies that the least significant byte (LSB) of a data element is stored at the lowest byte address. Most UNIX platforms (i.e., almost all except Compaq Alpha) are big-endian machines, while all Intel x86 and compatible platforms (e.g., Intel-Pentium and AMD Athlon, including those running both Windows and Linux) are little-endian machines. Some machines, like Intel Itanium, can be run in either big-endian mode (e.g., when running HP-UX), or in little-endian mode (e.g., when running Linux or Windows).

Inputs and Outputs

The OUTPUT2, OUTPUT4, and INPUTT4 modules have been modified to allow the user to specify the format of a binary file generated by these modules, i.e. whether the file is to be in "big-endian" format or "little-endian" format. The ASSIGN statement is used to assign physical files used by MSC.Nastran to FORTRAN units, and the desired output format is specified using the FORM= option.

For FORTRAN files, the format of the ASSIGN statement is:

```
ASSIGN logical-key[={filename|*}] [UNIT=u] [[STATUS=]{NEW|OLD|UNKNOWN}]
[[FORM=]{FORMATTED|UNFORMATTED|BIGENDIAN|LITTLEENDIAN|LTEND|<ostype>}] [DEFER]
[{TEMP|DELZERO}] [DELETE] [SYS='sys-spec']
```

In addition, for OUTPUT4 files a new utility program (OP4UTIL) has been developed that can test and convert OUTPUT4 files from one endian format to another.

The OP4UTIL utility may be used to validate, copy, or reformat binary files created using the MSC.Nastran OUTPUT4 module. The basic format of the "op4util" command is:

```
Msc2004 op4util <options> <file names>
```

These capabilities are available on all platforms except Cray Unicos.

Examples

To specify the endian format, the ASSIGN statement is used as follows:

Set the default OP2 file format to BIGENDIAN and assign two OP2 files, one to unit 12 with filename 'test_op2.12', and one to unit 35 with filename 'test_op2.35' in ASCII mode.

```
ASSIGN OP2 BIGENDIAN
...
ASSIGN OP2='test_op2.12' UNIT=12
```

```
ASSIGN OP2='test_op2.35' UNIT=35 FORM=FORMATTED
```

The OP4UTIL program is used as follows:

To generate a usage / help message:

```
msc2004 op4util
msc2004 op4util –h[elp]
msc2004 op4util -?
```

To convert a file from one big-endian to little-endian or vice-versa:

```
Msc2004 op4util [-x[change]] [-v[erbose]] [-m nnn] <from_fname> <to_fname>
```

To convert a file from one endian format to a specified endian format:

```
Msc2004 op4util <Endian-opt> [-v[erbose]] [-m nnn] <from_fname> <to_fname>
```

9.4 Reduced OP2 File SET Consistency Check

Introduction

A procedure to reduce the size of the op2 file produced for use by MSC.Patran in post-processing operations has been available as a DMAP alter since the release of MSC.Nastran 2001. With the release of MSC.Nastran 2004, this procedure has been incorporated into the MSC.Nastran DMAP and the alter is no longer required. To employ the procedure, MSC.Nastran Case Control SET commands and DMAP parameters are used. The procedure achieves its purpose by modifying the contents of two of the data blocks that are stored on the op2 file during POST output operations. The GEOM1 data block GRID point data record is modified so that it contains only a specified sub-set of all of the grids available in the model. Likewise, the element connection data records in the GEOM2 data block are modified so that they contain data for only a specified sub-set of the elements available in the model. The particular IDs to be retained in these data blocks are specified in SET list statements in the Case Control Section of the input data file. The user specifies the SET ID for the grid point list by assigning its value to the OGRDSET DMAP parameter. Similarly, the SET ID value assigned to the OELMSET DMAP parameter specifies the SET ID for the element list.

A new feature has been added to this procedure. This new feature ensures that all of the grid points connected to elements contained in the element list SET are also members of the grid point SET list. This test is called the SET consistency check. It is always performed when both the grid point set and element set are specified. Ensuring that the sets are consistent eliminates the problem sometimes encountered in post-processors when the op2 data is loaded. Some post-processors will refuse to load data for an element if the grid points connected to the element are not also present.

Benefits

The enhancement to the reduced op2 file size feature ensures that the grid point list used to “reduce” the grid geometry data contains all of the grid points that are connected to elements in the element set used to “reduce” the element connection data. Having a consistent set of data virtually eliminates the possibility of a post-processor rejecting element data due to element connection grid point data being missing. Until now, it has been up to the user to ensure that the grid point set was consistent with the element set. With this new release of MSC.Nastran, that burden has been removed from the user. MSC.Nastran will perform a consistency check of the grid point set and terminate the run if any grid points connecting the elements in the element set are missing from it. This improves user productivity in several ways.

Less time is spent re-running jobs to get the correct amount of reduced output for the op2 file. Less time is spent correcting the grid point set as all of the missing grid points are identified by MSC.Nastran, and a revised SET can be punched at user option.

Method and Theory

The theory behind this new feature is very simple. Each and every grid point connected to the elements in the element set should be present in the grid point set. The method is equally simple. A list containing all of the points present in the model is created. Next, each of the elements in the model is checked to see if its ID is in the element ID set. If it is, then each of the grid points that the element connects is flagged in the point ID list. Once all of the elements have been processed, each point in the point list that has been flagged as touched by an element is tested to see if it is present in the grid point ID set. If a point touched by an element is not present in the grid point set, then a FATAL error message is issued and the job will be terminated. The grid point set is also checked to see that every point in it was flagged during the element set processing that produced the element-related grid list. For each point in the grid point set that is not in the element-related grid point list or is not in the model, a WARNING message is issued. The element-related point ID list can be punched in Case Control SET format.

There are two other options available when using this new feature. Both options relate to how the content of the final point ID set list is created that controls the content of the GEOM1 grid point data record. One option simply uses the element-related list of point IDs as the grid point set. For this option, set consistency is guaranteed. The other option merges the input grid point set into the element-related list of points and uses the merged set of points as the grid point set. For this option, set consistency is not checked.

Inputs

The new set consistency check operation is automatically performed in MSC.Nastran when both the element set and the grid point set are specified. The reduced op2 file size capability already available in MSC.Nastran 2004 is controlled by the OELMSET and OGRDSET DMAP parameters and the specification of the associated Case Control SET lists. The OELMSET parameter value identifies the ID of the SET containing the IDs of the elements that are to be retained on the op2 file. The OGRDSET parameter value identifies the ID of the SET containing the IDs of the grid points that are to be retained on the op2 file. Several additional parameters have been introduced with the new set consistency check feature. All of the parameters associated with the reduced op2 file size and set consistency check feature are now summarized.

OELMSET – Integer – Default=0. Identification number of a Case Control command SET definition. The members of the specified SET represent the identification numbers of the finite elements that are to be retained in the “reduced” op2 file element connection data block.

OGRDSET – Integer – Default=0. Identification number of a Case Control command SET definition. The members of the specified SET represent the identification numbers of the grid points that are to be retained in the “reduced” op2 file grid geometry data block.

OPCHSET – Integer – Default=0. SET punch request flag. If OPCHSET=1, then the list of grid points used to reduce the grid point geometry data block will be punched in Case Control SET definition format.

OMSGLVL – Integer – Default=0. Set consistency check error message severity flag. The default causes FATAL messages to be generated if the grid set is not consistent with the element-related grid point set and the job is terminated. If OMSGLVL=1, the FATAL messages are reduced to WARNINGS and the job is allowed to continue.

OGRDOPT – Integer – Default=1. Selects the method used to create the set of grid points retained in the reduced grid point geometry data block. The default simply uses the set of grid point IDs listed in the OGRDSET Case Control SET. Set consistency is checked. OGRDOPT=2 uses the list of grid point IDs that are connected to elements in the OELMSET Case Control SET. OGRDOPT=3 merges the contents of the OGRDSET Case Control SET with the contents of the grid point list connected to the elements in the OELMSET Case Control SET. There is no consistency check for OGRDOPT=2 or OGRDOPT=3. OGRDOPT=0 turns the SET consistency check off altogether. For this case, the grid points retained are those specified in the OGRDSET SET and the elements retained are those specified in the OELMSET SET.

Outputs

The SET consistency check feature of the reduced op2 file size capability produces no printed output other than standard format MSC.Nastran FATAL and / or WARNING messages. Punched output can be produced at the user request. The punched output consists of a Case Control SET of the grid point IDs that are consistent with the OELMSET Case Control SET of elements.

Guidelines and Limitations

- Large THRU ranges in the grid point Case Control SET definition can generate large amounts of informational messages if the grid points in the THRU range are not present in the model.

- SPOINT IDs are treated as GRID point IDs. The SPOINT data record on the element connection data block (GEOM2) is not modified if it exists.
- Rigid element connections are ignored. Only finite element connectivity is examined for attached grid points.
- If duplicate element IDs across element types are encountered, no warning message is generated. All elements with same ID will be processed if the ID is in the OELMSET Case Control SET.
- This feature is active only when op2 postprocessing file generation is requested for the MSC.Patran program (v3 or higher) (param,post,-1) with geometry output.
- Case control grid point related output requests (e.g., DISP) must reference the OGRDSET Case Control SET. Case control element stress, strain and force requests (e.g., STRESS) must reference the OELMSET Case Control SET.
- Not available when superelements are present.

Demonstration Example

A simple example is presented that demonstrates the set consistency check feature. The model is composed of a series of disjoint elements and is not intended to be representative of any actual modeling situation. Two SETs are defined in the Case Control Section. The members of SET 100 are the IDs of elements that are to be retained on the reduced op2 element connection geometry data block. This SET ID is specified as the value of the OELMSET parameter and is entered in the Bulk Data Section using a PARAM Bulk Data entry. It is also referenced by the FORCE Case Control command. The members of SET 200 are the IDs of grid points that are to be retained on the reduced op2 grid point geometry data block. This SET ID is specified as the value of the OGRDSET parameter and is entered in the Bulk Data Section using a PARAM Bulk Data entry also. It, too, is also referenced by the DISP Case Control command.

Note: These PARAM entries could be placed in the Case Control Section instead of the Bulk Data Section.

Example files mass.dat and mass_bs.dat can be found in the TPL.

Example Input Data (TPL: rop2s*.dat)

```
NASTRAN SYSTEM(361)=1
$
```

```
$============================================================================
$
$  Particular features of this example:
$  1) the element ID set is SET 100.  It has a thru range that includes
$     non-existent elements.  There is no informational output for this
$     condition.  SET 100 is selected via param,oelmset,100 bulk data
$     entry.
$  2) the point ID set is SET 200.  It contains point ID 5 which does
$     not exist in the model.  It also contains point 1005 which exists,
$     but is not referenced by an of the elements in SET 100.  Both of
$     these conditions cause informational messages to be generated.
$     SET 200 is selected via param,ogrdset,200 bulk data entry.
$  3) the element ID set (SET 100) produces an element-related point ID
$     set to be generated that contains several points that are not in
$     the point ID set (SET 200).  This condition generates FATAL
$     messages and causes termination of the run unless the OMSGLVL
$     parameter is set to 1, which reduces the messages to WARNING level
$     only and the run continues.
$  4) the element-related point set is punched in case control SET
$     format due to the presence of the param,OPCHSET,1 bulk data entry.
$     This set could be used to replace the existing point ID set on
$     a subsequent run and as long as the element set SET remained the
$     same, the two would produce a consistent set of data.
$
$  Example Summary:
$  1) SETs supplied are not consistent.
$  2) Element-related point set is punched in case control SET format.
$
$============================================================================
$
SOL 101 $ STATIC ANALYSIS
CEND
$
  DISPL = 200
  FORCE = 100
  LOAD = 85
  MPC = 1
  set 100 = 27,35,25,41234,123,thru,134,9701,9901
  set 200 = 1,thru,5,1005
BEGIN BULK
param,post,-1
$
$ tested feature controls
$
param,oelmset,100  $  select reduced element SET ID from case control
param,ogrdset,200  $  select reduced point   SET ID from case control
param,opchset,1    $  punch element-related point set in SET format
$ param,omsglvl,1    $  WARNING message only
$
spoint,20001,21001
spoint,30001
celas3,9701,175,20001
celas3,9702,175,21001
celas3,9801,175,30001
celas3,9901,175,40001
$
GRID    777             10.     0.      0.
.
.
```

```
.
ENDDATA
```

Example Output

The example problem contains inconsistent sets. The MSC.Nastran run terminates with FATAL messages identifying the inconsistencies as shown in the following excerpt from the .f06 listing.

```
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 21 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 22 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 111 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 121 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 171 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 181 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 20001 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER FATAL MESSAGE 7759 (MTM36A)
    ELEMENT(S) IN SET 100 CONNECT(S) POINT ID 40001 NOT PRESENT IN GRID SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ALL POINTS CONNECTED TO ELEMENTS IN ELEMENT
SET.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER WARNING MESSAGE 7760 (MTM36A)
GRID SET 200 CONTAINS POINT ID 5 NOT PRESENT IN THE MODEL.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ONLY POINTS ACTUALLY IN THE MODEL.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
*** USER WARNING MESSAGE 7761 (MTM36A)
    GRID SET 200 CONTAINS POINT ID 1005 NOT CONNECTED TO ANY ELEMENTS IN SET 200.
     USER ACTION: MAKE SURE THAT THE GRID POINT SET CONTAINS ONLY POINTS ACTUALLY IN THE MODEL.
    PROGRAMMER INFORMATION: MATMOD OPTION 36, SUB-OPTION 1
```

The fatal messages inform the user that grid points 21, 22, 111, 121, 171, 181, 20001 and 40001 are connected to elements in set 100 (the OELMSET) and that they are not present in set 200 (the OGRDSET). The user is also warned that grid point ID 5 in set 200 is not a model grid point. In addition, a warning is issued for point 1005 in set 200. This point was not connected to any of the elements in set 100.

The user has requested that the element-related set of grid points be punched. This set of grid IDs is punched in Case Control SET format as shown.

```
SET 200 = 1,THRU,4,21,22,111,121,171,181,20001,40001
```

If the above SET 200 is used to replace the existing SET 200 definition, then the element set and grid point set would be consistent and there would be no fatal messages generated.

9.5 SPC and SPCD Entries in Machine Precision

In previous versions, enforced displacement values defined on the SPC and SPCD entries were always treated as single precision numbers. Under this enhancement the SPC and SPCD entries have been converted to machine precision format. All real values of these entries will be maintained in machine precision during all operations in static analysis leading to more accurate computation of loads.

Limitation

The entry, GMSPC used for p-element analysis still remains in real single precision in the bulk data interpretation.

9.6 Reading of PUNCHed Long Field Format Bulk Data

Large-field input format requires (at least) two lines for each Bulk Data entry. So, for an entry with fields 5-9 blank, a continuation entry is necessary. A fatal error results if a long-field entry does not contain at least one continuation entry. However, if requested, the PUNCHed bulk data from this run would still be created with the bad long-field format entry. This restriction has now been removed and a single line long-field format entry is now acceptable. Its PUNCHed bulk data would also be created consistent with the input bulk data entry.

9.7 New Complex Conjugate Option for Matrix Multiplication

The MPYAD and SMPYAD modules have been enhanced to compute conjugate matrix multiplication for complex matrices. The transpose flag DMAP parameter for both modules has been extended as follows:

Transpose Flag	Meaning
0	No Transpose
1	Transpose
2	Conjugate and No Transpose
3	Conjugate and Transpose

Examples:

1. Compute $[D] = [A^*][B]$ where $[A^*]$ is the complex conjugate transpose of $[A]$

   MPYAD A,B,/D/3 $

2. Compute $[X] = [A] \cdot [\bar{B}][C] - [F]$ where $[\bar{B}]$ is the complex conjugate of $[B]$.

   SMPYAD A,B,C,,,F/X/3/1/-1/0/0/2 $

3. Compute $[X] = [U^*][K][U]$

   SMPYAD U,K,U,,,/X/3////3 $

9.8 Acceleration Loads (ACCEL and ACCEL1 Bulk Data Entries)

Traditionally, MSC.Nastran users have used the GRAV Bulk Data entry to apply acceleration loads. The GRAV load is applied on the overall structural model as a uniform load.

Previously, MSC.Nastran was unable to apply an acceleration load that varied across the structure. New Bulk Data entries, ACCEL and ACCEL1, have removed this limitation. They allow the user to apply acceleration loads at individual grid points or in a specified region. Both ACCEL and ACCEL1 loads are used in the same way as other load entries (such as GRAV, FORCE, and MOMENT, etc.) through the MSC.Nastran Case Control commands.

Examples showing the use of ACCEL and ACCEL1 are available in the TPL - accelqr.dat and accel1qr.dat.

Example:

```
SOL 101
TIME 10
CEND
TITLE= UNIFORMLY VARYING ACCELERATION LOAD
SUBTITLE = ACCEL LOAD TEST DECK
AUTOSPC(NOPRINT)=YES
ECHO = SORT
SPC = 1000
DISP(PRINT)   = ALL
STRESS(PRINT) = ALL
OLOAD(PRINT ) = ALL
$
SUBCASE 1
   LABEL= GRAVITY LOAD VARIES IN THE X DIRECTION FOR A SQUARE PLATE
   LOAD = 1
SUBCASE 2
   LABEL= GRAVITY LOAD VARIES IN THE Y DIRECTION FOR A SQUARE PLATE
   LOAD = 2
$
BEGIN BULK
CQUADR  1       1       1       2       7       6
CQUADR  2       1       2       3       8       7
CQUADR  3       1       3       4       9       8
CQUADR  4       1       4       5       10      9
CQUADR  5       1       6       7       12      11
CQUADR  6       1       7       8       13      12
CQUADR  7       1       8       9       14      13
CQUADR  8       1       9       10      15      14
CQUADR  9       1       11      12      17      16
CQUADR  10      1       12      13      18      17
CQUADR  11      1       13      14      19      18
CQUADR  12      1       14      15      20      19
CQUADR  13      1       16      17      22      21
```

```
CQUADR  14      1       17      18      23      22
CQUADR  15      1       18      19      24      23
CQUADR  16      1       19      20      25      24
$-------2-------3-------4-------5-------6-------7-------8-------9-------0-------
ACCEL   1               .267261 .534522 .801784 X                       +
+       0.0     -32.2   4.0     -161.0
ACCEL   2       22      .267261 .534522 .801784 Y                       +
+       0.0     -32.2   -4.0    -161.0
CORD2R  22              0.0     0.0     0.0     0.0     0.0     1.0     +
+       0.0     1.0     0.0
$-------2-------3-------4-------5-------6-------7-------8-------9-------0-------
GRID    1               0.0
GRID    2               1.00000
GRID    3               2.000000
GRID    4               3.000000
GRID    5               4.000000
GRID    6               0.0     1.0
GRID    7               1.00000 1.0
GRID    8               2.0000001.0
GRID    9               3.0000001.0
GRID    10              4.0000001.0
GRID    11              0.0     2.0
GRID    12              1.00000 2.0
GRID    13              2.0000002.0
GRID    14              3.0000002.0
GRID    15              4.0000002.0
GRID    16              0.0     3.0
GRID    17              1.00000 3.0
GRID    18              2.0000003.0
GRID    19              3.0000003.0
GRID    20              4.0000003.0
GRID    21              0.0     4.0
GRID    22              1.00000 4.0
GRID    23              2.0000004.0
GRID    24              3.0000004.0
GRID    25              4.0000004.0
MAT1    1       1.0+5           0.3     1.0
PSHELL  1       1       0.1     1               1
+                                       0
SPC1    1000    123456  1       6       11      16      21
ENDDATA
```

9.9 A Caution Concerning MSC.Access Application Development

In anticipation of functional changes to the MSC.Access data base organization, changes to the Application Program Interface (API) are being introduced during the basic MSC.Nastran 2005 release. The changes occur in the user interfaces to the Open routines for the keyed objects. These interfaces are:

OPENC – Create a Keyed Object

OPENR – Read or Update a Keyed Object

OPENSQ – Read a Keyed Object using Sequential Methods

Parameters made obsolete during the MSC.Nastran Version 66 releases are being reused and redefined.

DBFLOC – Locate a Keyed Object within a Group of Logical Data Bases

An additional parameter has been added.

The user application should now provide a destination variable for the returned information in the arguments to the DBFLOC, OPENR and OPENSQ interfaces. Usage of a constant could result in premature application termination due an attempt to modify protected storage. The definition of the KEY variable in other interfaces has also changed, however until production release of the new functionality along with an update MSC.Access Users Manual, the current application interface will remain functional and provide a correct interfaces to any existing and current 2005 created MSC.Access data bases.

Updated pages for the interfaces from the MSC.Access Users Manual are now provided. The new access key, called BBB-Tree method, will be explained in the next release.

Subroutine Name: DBFLOC

1. Entry Point: DBFLOC
2. Purpose: Locate and open an object among the open database(s)
3. Calling Sequence: CALL DBFLOC (NAME, FILNUM, FLEN, FNUM, ***KEYLEN***,IRET)

NAME	Array-input	Dictionary entry of an object name
FILNUM	Integer-output	Logical file number assigned to the opened object

FLEN	Integer-output	The length of an instance for a keyed object or the total length in words for a sequential object
FNUM	Integer-output	The number of entries for keyed object or "1" for sequential objects
KEYLEN	**Integer-output**	**The key length in words for keyed objects**
IRET	Integer-output	Return code, conforming to OPENR/OPENS error codes, or the additional
		101 - object format code is neither RECORD or VECTOR
		102 - dictionary entry could not be located among open database(s)

4. Method: The object is first located, is possible, among the open databases by search from low to high logical data enumeration. Once the first is located, either OPENR or OPENS is used to depending upon its form. The OPENR allows for application updates, while OPENS for sequential objects opens for read-only. Statistics concerning the object size are also returned to the application.

Subroutine Name: OPENC

1. Entry Point: OPENC
2. Purpose: Create new keyed object and return a logical file reference.
3. Calling Sequence: CALL OPENC (DBNUM,NAME,WRDREC,FILNUM,***KEYLEN***,**CLSTER,**D3,D4,D5,D6,IRET)

DBNUM	Integer-input	Logical database number
NAME	Array-input	Dictionary entry and keyed object name to create
WRDREC	Integer-input	Number of words per logical record in object
FILNUM	Integer-output	Logical handle number assigned to the object

KEYLEN	**Integer-input**	**The number of words in the key**
		0-> Use Hierarchal Key Method
		+n-> Use BBB-Tree Method
CLSTER	**Integer-input**	**Clustering Method**
		0 -> Use standard Key clustering algorithm
		1-> Re-order keys for optimum entry storage
D3 D4 D5 D6	Integer-input	Currently unused. In prior releases, these arguments represented memory addresses for I/O buffer work areas.
IRET	Integer-output	Return code from the routine
		0 -> Normal data block creation
		1 -> Requested NAME already existed
		2 -> Too many logical files open

4. Method: NAME is checked to determine if it already exists.

   The control area is checked to make sure that a new object can be opened and made available for processing.

   If both conditions above are satisfied, the buffer management area is cleared and the DAT control area, as described in the DICENT routine description is created. The primary map blocks and the first data area are reserved in the dictionary and stored in the DAT array.

   The DAT array is copied to both the control area and the primary map block for file management.

   The logical file number assigned by the OPENC is returned to the calling application program.

Subroutine Name: OPENR

1. Entry Point: OPENR
2. Purpose: Open existing keyed objects for random access updating and return logical file reference.

3. Calling Sequence: CALL OPENR (DBNUM,NAME,WRDREC,FILNUM,***KEYLEN***,D2,D3,D4,D5,D6,IRET)

DBNUM	Integer-input	Logical database number
NAME	Array-input	Dictionary entry and object name to update
WRDREC	Integer-output	Number of words per record in object
FILNUM	Integer-output	Logical handle number assigned to the object
KEYLEN	**Integer-output**	**The number of words in the key**
		0-> Used Hierarchal Method
		+n-> Used B-Tree Method
D2 D3 D4 D5 D6	Integer-input	Currently unused. In prior releases, these arguments represented memory addresses for I/O buffer work areas.
IRET	Integer-output	Return code from the routine
		0 -> Normal data block open
		1 -> Requested NAME does not exist
		2 -> Too many logical files open
		3 -> Currently unused. In prior releases, it indicated too few buffers allocated.
		4 -> Object already open

4. Method: DICRDR is used to check the existence of the object NAME and to retrieve its DAT control area.

   When the object exists, it is checked for a conflict to another logical file.

   When no conflict exists, then a check for available processing space (i.e., less than thirty logical files currently open) is made.

   When space is available, the DAT control area is copied to the available control area. The remaining control fields are initialized for object management.

The logical handle number and words per record are returned to the calling application program.

Subroutine Name: OPENSQ

1. Entry Point: OPENSQ
2. Purpose: Open a keyed object for sequential processing and return logical file reference.
3. Calling Sequence: CALL OPENSQ (DBNUM,NAME,FILNUM,***KEYLEN***,IRET)

DBNUM	Integer-input	Logical database number
NAME	Array-input	Object dictionary entry and object to open
FILNUM	Integer-output	Logical handle number assigned to object
KEYLEN	**Integer-output**	**The number of words in the key**
		0-> Used Hierarchal Method
		+n-> Used B-Tree Method
IRET	Integer-output	Return code from the routine
		0 -> Normal data block open
		1 -> Requested object does not exist
		2 -> Too many logical files open
		3 -> Unused
		4 -> Object already open for update

4. Method: This routine can only be used to open keyed objects for read access.

   The existence of the object is determined by DICRDR, and its form (keyed) is verified.

   Control areas are created for logical file operations and initialized with file control data.

   FILNUM is returned to the calling routine.

9.10 Divergent Thermal Results Error Correction (Q1-0768221)

This is an error correction for radiation boundary conditions in nonlinear heat transfer. It can only occur when a RADBC entry is used in a nonlinear solution. For problems where radiation heat transfer dominates conduction, strange non-physical results have been observed. For most problems where radiation is modest, no bad results will be observed. In the January 2004 beta release this error was correct for the linear (QUAD4 and TRIA3) elements. This correction is now extended to the quadratic (QUAD8 and TRIA6) elements in this release.

9.11 Displacement Output Filters

The option to filter displacement output based on user-defined threshold values is now available. This option can be requested using the following new keywords in the DISPLACEMENT command.

Format:

$$DISPLACEMENT\left(....,\left[\frac{TM=f}{T1=f,T2=f,T3=f}\right],\left[\frac{RM=f}{R1=1,R2=f,R3=f}\right]\right)$$

Examples:

DISP(T1=1.0E-3, T3=1.0E-2) = ALL
DISP(TM=1.0E-3, PRINT, PLOT) = ALL
DISP(TM=1.0E-3, PRINT, PLOT, SORT2) = 20

Describers	Meaning
TM	Translational Magnitude Filter
T1, T2, T3	Translational Component Filters
RM	Rotational Magnitude Filters
R1, R2, R3	Rotational Component Filters
F	Filter value (Real > 0.0)

Remarks:

1. Displacement components may be selected to control filtering to reduce the amount of output produced. When magnitudes are selected, the component values are ignored. Only a single positive value for f can be supplied and comparisons are performed in the global reference frame. Comparisons are performed after the SET intersection is performed against the domain. Selection of this option does not effect the MAXMIN(GRID) operations. Scalar comparisons are performed using the minimum of all supplied values for the filters. Complex values filters are performed on the Magnitude when components are selected. Complex vector magnitudes follow a derivation using a deterministic interpretation for frequency response.

2. When using filters the compound usage of the verbs PRINT, PLOT is allowed. The entries in the printed output are the entries that exceed any threshold, while the remaining entries within the SET are marked as plot to allow for post-processing operations. When SORT2 is selected, then print, plot must be used to allow for table transpose operations to occur. When any entry in the SORT2 format is above the threshold, all values for time or frequency will be printed for the grid.

9.12 Write Results Recovery for Subcases into Separate F06 Files

Recovery results written to the F06 output file can now be redirected to separate output files for each subcase.

Inputs

Both the ASSIGN and POST commands are modified for assigning the physical filename and specification of the subcase specific filename suffix respectively.

The modified ASSIGN and POST command are shown below followed by a small example illustrating the use of the above commands. In this example the results re-directed for both subcases are redirected to use specified files.

POST Post-Processor Data Specifications

Controls selection of data to be output for post-processing functions via the OUTPUT2 module interface for selected commercial post-processor products. Another feature is to redirect F06 output file results for a subcase to a user defined file.

Format:

$$\text{POST}\left\{\begin{array}{c}\text{TOFILE}\\\text{TOCASE}\end{array}\right\}\left\{\begin{array}{c}\text{furn}\\\text{filename}\end{array}\right\}[\text{ppname}][\text{oplist}]$$

Examples:

POST TOFILE 51 PATRAN NOSTRESS
POST TOFILE SUBCASE8
POST TOCASE SUFNAME1

Describer	Meaning
TOFILE	Keyword to specify the destiny of output files. (No default if it appears above all subcases.)
TOCASE	Keyword to specify the destiny of subcase results to user-defined output files. (No default if it appears above all subcases.)
furn	Fortran file unit reference number where data will be written. (Integer>0)
filename	Suffix filename (see Remark **8.**). (Char8)
ppname	Name of the target post-processor program. (Default = PATRAN)
oplist	Names of output items to be processed.

Remarks:

1. The POST Case Control command controls the placement of output data on external fortran files for use by commercial post-processors. Use of the POST command generates the proper value for the POST DMAP parameter associated with the particular post-processor. All of the other parameter controls related to the POST DMAP parameter remain in effect and are described in “**Parameters**” on page 603. The products supported are identified in the following table. PATRAN is the default post-processor name used for ppname. DBC output (POST=0) cannot be controlled by the POST command.

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

2. The TOFILE describer is followed by the specification of either a FORTRAN unit reference number or a file name associated with the external file that receives the output data. If a FORTRAN unit number is used, the file must be associated with it via the ASSIGN File Management Statement. If POST appears above all subcases, TOFILE must be used to specify either a FORTRAN unit reference number or a file name. The default value of TOFILE, which appears under a subcase, will inherit from the value given in the POST above all subcases. If the unit reference number is associated with a form=formatted file, changes in unit numbers across subcases are not allowed.
3. The data that can be controlled for each post-processor product is limited and is identified under the description of the POST and related DMAP parameters in "**Parameters**" on page 603. The keywords that can be used for the oplist options are shown in the following table. If an output item supported by a particular post-processor is described in "**Parameters**" on page 603 but is not listed here, then the POST command cannot be used to control its output to the external file.

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

Output Item	oplist Keyword	Case Command
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	----------------

4. Output data items must have been generated via the appropriate case control command in order for the data to be available for post-processing options. For example, the specification of SPCF in the oplist of the POST command will not produce forces of single point constraint on the POST output file unless there is a SPCF Case Control command present. Refer to the tables under the POST parameter description in “**Parameters**” on page 603 for a list of the output items supported by each post-processor.
5. Any data generated by a case control output request is automatically included in the oplist of the POST command. If output data is not wanted for a particular case, then the characters “NO” should be the first two characters of the keyword in the oplist. For example, NODISP specifies that displacements are not to be posted to the output file even though they have been requested via the DISP Case Control command. Alternatively, the related POST parameters may be used. For example, to avoid outputting any displacements whatsoever to the .op2 file, use a “PARAM, OUG, NO” Bulk Data entry.
6. Certain data (e.g. geometry) is always generated and is not dependent upon the presence of a case control command in the input data. The POST command affects the placement of this data on the external file only insofar as the selection of the post-processor defines the value of the POST DMAP

parameter value. The actions described in “**Parameters**” on page 603 under the POST parameter description will prevail for the particular value of POST associated with the selected post-processor. The primary purpose of the POST command is to give the user more control over subcase-dependent output data being stored on the external OUTPUT2 file.

7. If a POST command is present within any subcase, a POST command must also be present above the subcase level. 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 generate any output to the OUTPUT2 file for displacements. This also implies that changing the OUTPUT2 file unit reference number with the TOFILE 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 oplist for the current POST command.
8. When the name of an output file is specified by keyword TOFILE, the ASSIGN statement in the File Management Section (FMS) can be used to specify the full path of its root name. the logical-key word for the root name is OUTPUT2F. The default root name is the MSC.Nastran job name. FORTRAN unit reference number 19 has been reserved by MSC.Nastran for OUTPUT2F, although the user can assign other FORTRAN unit number to it. The full file name is in the form of <root name>.<suffix filename>.
9. When the name of an output file is specified by keyword TOCASE, the ASSIGN statement in the File Management Section (FMS) can be used to specify the full path of its root name. the logical-key word for the root name is OPCASE. The default root name is the MSC.Nastran job name. FORTRAN unit reference number 22 has been reserved by MSC.Nastran for OPCASE. Although the user can assign other .FORTRAN unit numbers to it. The full file name is in the form of <root name>.<suffix filename>. Also ppname and oplist are not required. If ppname and oplist are specified, they will be ignored. Suffix filename must be specified with keyword TOCASE.

ASSIGN Assigns Physical File

Assigns physical file names or other properties to DBset members or special FORTRAN files that are used by other FMS statements or DMAP modules. Also, assigns physical name and / or other properties to Modal Neutral Files (MNF) for MSC.Nastran / ADAMS interface.

Format 1: Assign a DBset member name

```
ASSIGN   log-name= [ = 'filename1' ]   [TEMP]  [ DELETE]  [ SYS='sys-spec' ]
                   [ = *           ]
                   [ = '*'         ]
```

Format 2: Assign a FORTRAN file

```
ASSIGN   logical-key [ = 'filename2' ]    [UNIT = u]
                     [ = *           ]
                     [ = '*'         ]

        [ [STATUS =] { NEW     } ]
        [            { OLD     } ]
        [            { UNKNOWN } ]

        [ [FORM =] { FORMATTED    } ] [RECL = 1][SIZE = s]
        [          { UNFORMATTED  } ]
        [          { BIGENDIAN    } ]
        [          { LITTLEENDIAN } ]
        [          { LTLEND       } ]
        [          { <ostype>     } ]

        [DEFER ]  [ TEMP    ]  [ DELETE]  [SYS = 'sys-spec']
                  [ DELZERO ]
```

Examples:

1. Assign the DBALL DBset:

```
ASSIGN DB1='filename of member DB1'
INIT DBALL LOGI=(DB1)
```

2. Assign FORTRAN file 12 to the OUTPUT4 module using the ASCII option:

   ```
   ASSIGN OUTPUT4='filename of FORTRAN file'
   UNIT=12, FORM=FORMATTED
   ```

3. Assign FORTRAN file to the OPCASE using the ASCII option:

   ```
   ASSIGN OPCASE='Filename of FORTRAN file', STATUS=NEW
   ```

4. Define SYS parameters for the SCR300 DBset file using the default file name.

   ```
   ASSIGN SCR300 SYS='...'
   ```

5. Set the default OP2 file format to BIGENDIAN and assign two OP2 files, one to unit 12 with the file name "test_op2.12' and one to unit 35 with file name 'test_op2.35' in ASCII mode.

   ```
   ASSIGN OP2 BIGENDIAN
   ...
   ASSIGN OP2='test_op2.12' UNIT=12
   ASSIGN OP2='test_op2.35' UNIT=35 FORM=FORMATTED
   ```

Describer	Meaning
log-name	The name of a DBset member name. log-name may also be referenced on an INIT statement after the LOGICAL keyword.
filename1	The physical filename assigned to the DBset member. If the default filename (if there is one) is to be used, filename1 may be omitted or specified as * or '*'. See Remark **6.**
logical-key	Specifies defaults for STATUS, UNIT, and FORM of FORTRAN files for other FMS statements, DMAP modules, punching and plotting operations.
filename2	The physical file name assigned to the FORTRAN file. If the default filename is to be used, filename2 may be omitted or specified as * or '*". See Remark **7.**
UNlT=u	u is the FORTRAN unit number of the FORTRAN file. If this describer is omitted and if filename2 is omitted, this ASSIGN statement will update the defaults for subsequent ASSIGN statements for the same logical-key value. See Remark **7.**
TEMP	Requests that the file associated with log-name or logical-key / UNIT be deleted at the end of the run.
DELETE	Requests that the file associated with logical-key / UNIT, if it exists before the start of the run, be deleted.

Describer	Meaning
DELZERO	Requests that the file associated with logical-key / UNIT be deleted at the end of the run if it is zero-length, that is, if it does not contain any data.
STATUS	Specifies whether the FORTRAN file is being created (STATUS=NEW) or has been created prior to the run (STATUS=OLD). If its status is not known, then STATUS=UNKNOWN is specified.
FORM	Indicates whether the FORTRAN file is written in ASCII (FORM=FORMATTED) or binary (FORM=UNFORMATTED, BIGENDIAN, LITTLEENDIAN, LTLEND, <ostype>) format. See Remark **11.**
DEFER	Defers opening / creating the specified file. That is, the file will not be opened / created during MSC.Nastran initialization. The file must be explicitly opened by the module or DMAP accessing the file, using, for example, FORTIO, before it can be used.
sys-spec	System specific or machine-dependent controls. For DBset files, these control I / O performance. For FORTRAN files, these are controls for IBM / MVS-type computers only. See Remark **14.**
RECL = l	The size of a block of input / output information specified in words. See Remark **15.**
SIZE = s	The number of blocks allocated to the DBC database. See Remark **16.**

Remarks:

1. The ASSIGN statement and its applications are discussed further in the "***Database Concepts***" on page 513 of the *MSC.Nastran Reference Guide*.
2. The log-name or logical-key describer must be the first describer on the ASSIGN statement. All other describers may appear in any order. With the exception of log-name, logical-key, filename1, filename2, and sys-spec, describers and values longer than four characters may be abbreviated to four characters.
3. For FORTRAN files, the logical-key names and their default attributes are listed in **Table 2-1**. If a logical-key name is identified as "Assignable YES", then the defaults may be overridden on the ASSIGN statement.

4. Certain reserved names may not be used for log-names or logical-key names. These names are the logical names listed in **Table 2-1** that are identified as “Assignable NO”. This list includes: SEMTRN, LNKSWH, MESHFL, LOGFL, INPUT, PRINT, INCLD1, and CNTFL. If they are used, then a fatal message is issued. Also unit numbers 1 through 10, 14, 16, 18, 19 and 21 should not be assigned. PUNCH and PLOT may be used but are not recommended.
5. If one of the logical-key names indicated in the Remarks **3.** and **4.** is not specified on this statement, then it is assumed to be a DBset member name log-name as shown in Format 1.
6. If the same log-name is used on more than one DBset ASSIGN statement, the following rules apply:
    a. If there is no current entry for the specified log-name, a new entry in the DBset tables will be created. If there is an existing entry for the specified log-name, the ASSIGN parameters will modify that entry instead of creating a new one.
    b. If filename1 is omitted or is specified as * or ‘*’, the default file name or, if this is a second or subsequent ASSIGN statement for the same log-name, the previously specified file name (or default name if none was previously specified) will be used.
7. If the same logical-key is used on more than one FORTRAN file ASSIGN statement, the following rules apply:
    a. If filename2 is omitted (or specified as * or ‘*’) and if the UNIT describer is omitted, the ASSIGN parameters will modify the system default entry for the logical-key, establishing the new defaults for any subsequent ASSIGN entry for the logical-key. Note, however, that any entries previously created with the same logical-key will not be modified by the new parameters specified on this ASSIGN statement.
    b. If the value specified by the UNIT describer matches the value for an entry created by a previous ASSIGN statement with a UNIT describer, then:
        - if the logical-key values are different, a UFM will be generated,
        - if the logical-key values are the same, the previous entry will be updated instead of having a new entry created.
    c. If the value specified by the UNIT describer does not match the value for an entry created by a previous ASSIGN statement with a UNIT describer, then a new entry will be created in the FORTRAN unit tables.

d. If the file name is omitted or specified as * or '*', the default file name or, if this is a second or subsequent ASSIGN statement for the same logical-key / UNIT combination, on previously specified file name (or default name if none was previously specified) will be used.

8. If it is necessary to execute the INPUTT4 and OUTPUT4 modules on the same unit, then specify ASSIGN OUTPUT4 only. The same is recommended for the INPUTT2 and OUTPUT2 modules.

9. STATUS, UNIT, and FORM are ignored if assigning a log-name (DBset member name).

10. FORM=FORMATTED must be specified for a unit when:

    - ASCII output is desired from the OUTPUT4 DMAP modules that processes the unit and, for Cray UNICOS, when ASCII input is supplied to the INPUTT4 DMAP module that processes the unit. See the ***MSC.Nastran 2005 DMAP Programmer's Guide***.
    - FORMAT=NEUTRAL is selected on the DBUNLOAD and DBLOAD FMS statements that process the unit. See the "***Database Concepts***" on page 513 of the *MSC.Nastran Reference Guide*.
    - The neutral file format is desired for the OUTPUT2 module and, for Cray UNICOS, when ASCII input is supplied to the INPUTT2 module.

11. For the DBUNLOAD, OUTPUT2 and OUTPUT4 modules, binary format may be requested using FORM=UNFORMATTED and, for all platforms except Cray UNICOS, using FORM=BIGENDIAN, FORM=LITTLEENDIAN, FORM=LTLEND or FORM=<ostype>. The FORM=BIGENDIAN, FORM=LITTLEENDIAN, FORM=LTLEND and FORM=<ostype> specifications are used when the generated output file is to be processed on a platform other than current platform. The format appropriate for the platform on which the file is to be processed (the target platform) must be specified. FORM=LTLEND is equivalent to FORM=LITTLEENDIAN. The FORM=<ostype> specification can by used as a convenience, allowing the desired output format to be specified using the target platform OS name or vendor (if there can be no ambiguity) instead of its actual binary file format. <ostype> can be one of the following:

    - AIX, FUJITSU, HPUX, IRIX, PRIMEPOWER, SOLARIS, SUPERUX or UXPV. These are equivalent to BIGENDIAN.
    - ALPHA, LINUX or WINDOWS. These are equivalent to LITTLEENDIAN.

See the *MSC.Nastran 2004 r3 Installation and Operations Guide* for further information on binary file formats.

12. For all platforms except Cray UNICOS, the FORM= describer is ignored for the DBLOAD, INPUTT2 and INPUTT4 modules. MSC.Nastran determines the actual file format when it accesses the specified file. If the FORM= describer is specified on an ASSIGN statement for these logical-keys, the syntax of the describer will be validated but will otherwise be ignored. Note, however, that the DBLOAD and INPUTT2 modules cannot process input files in other than the native binary format. That is, a binary file in BIGENDIAN format cannot be processed on a LITTLEENDIAN platform and vice versa. For MSC.Nastran on Cray UNICOS, the FORM= describer is required for the DBLOAD, INPUTT2 and INPUTT4 modules if the file does not have the default format.
13. For the DBUNLOAD and OUTPUT2 modules, if FORM is other than UNFORMATTED (or equivalent, e.g., BIGENDIAN on an AIX or HPUX platform and LITTLEENDIAN on a Linux or Windows platform), then only data blocks with an NDDL description are processed. (See the ***MSC.Nastran 2005 DMAP Programmer's Guide*** under the DATABLK statement.) An NDDL description is required for TYPE=TABLE and none is required for TYPE=MATRIX. The data block must be processed with FORM=UNFORMATTED if TYPE=UNSTRUCTURED, KDICT or KELM.
14. See the *MSC.Nastran 2004 r3 Installation and Operations Guide* for further information on sys-spec controls and on machine-dependent aspects of the ASSIGN statement. Also, if there are SYS specifications on more than one ASSIGN statement specifying the same log-name or logical-key / UNIT combination, the second and subsequent specifications will appended to the current SYS specification with a comma separator.
15. Currently the RECL keyword is used by the DBC module and has a default minimum of 1024 words. The maximum allowed is 65536 words and is used to increase the database capacity.
16. The SIZE keyword is used by the DBC module and has a default of 16777215. The maximum allowed is 2147483647 and is used to increase the database capacity. MSC.Patran releases before 2001 should use the defaults for RECL and SIZE or database verification failures will occur.
17. logical-key name MNF does not utilize UNIT or FORM.

Table 2-1 FORTRAN Files and Their Default Attributes

Logical Key Name	Physical Name	Unit No.	Form	Status	Assignable	Open	Access	Description/ Application
SEMTRN	sdir / data.f01	1	FORMATTED	NEW	NO	YES	SEQ.	Input Data Copy Unit
LNKSWH	sdir / data.f02	2	UNFORMATTED	NEW	NO	YES	SEQ.	Link Switch Unit
MESHFL	*sdir/data*.f03	3	FORMATTED	NEW	NO	YES	SEQ.	Input Data Copy Unit
SEMTRN	sdir / data.f01	1	FORMATTED	NEW	NO	YES	SEQ.	Input Data Copy Unit
LNKSWH	sdir / data.f02	2	UNFORMATTED	NEW	NO	YES	SEQ.	Link Switch Unit
MESHFL	*sdir/data*.f03	3	FORMATTED	NEW	NO	YES	SEQ.	Input Data Copy Unit
LOGFl	*out*.f04	4	FORMATTED	NEW	NO	YES	SEQ.	Execution Summary Unit
INPUT	*data*.dat	5	FORMATTED	OLD	NO	YES	SEQ.	Input File Unit
PRINT	*out*.f06	6	FORMATTED	NEW	NO	YES	SEQ.	Main Print Output Unit
PUNCH	*out*.pch	7	FORMATTED	NEW	YES	YES	SEQ.	Default Punch Output Uniit
	authorize.dat	8	FORMATTED	OLD	NO	YES	SEQ.	Authorization File
INCLD1					NO			Available for Use
CNTFL					NO			Available for Use
INPUTT2	REQ	REQ	++	OLD	YES	NO	SEQ.	INPUTT2 Unit
OUTPUT2+	*out*.op2	12	UNFORMATTED*	NEW	YES	YES	SEQ.	OUTPUT2 Unit
INPUTT4	REQ	REQ	++	OLD	YES	NO	SEQ.	INPUTT4 Unit
OUTPUT4	REQ	REQ	UNFORMATTED*	NEW	YES	NO	SEQ.	OUTPUT4 Unit
PLOT	*out*.plt	14	UNFORMATTED	NEW	YES	YES	SEQ.	Plotter Output Unit
BULKECHO	*out*.becho	18	FORMATTED	NEW	YES	YES	SEQ.	Plotter Output Unit
OUTPUT2F	out	19	UNFORMATTED	NEW	YES		SEQ.	Named OUTPUT2 Pattern
OPCASE	REQ	22	FORMATED	NEW	YES		SEQ.	Available for Use
TOPDES	*out*.des	21	FORMATTED	NEW	YES	YES	SEQ.	Topology Optimization
DBC	*out*.xdb	40	UNFORMATTED	NEW	YES	YES	DIRECT	Database Converter Unit
DBUNLOAD	REQ	50	UNFORMATTED*	NEW	YES	NO	SEQ.	DBUNLOAD FMS statement
DBLOAD	REQ	51	++	OLD	YES	NO	SEQ.	DBLOAD FMS statement

Table 2-1 FORTRAN Files and Their Default Attributes (continued)

Logical Key Name	Physical Name	Unit No.	Form	Status	Assignable	Open	Access	Description/ Application
MNF	*out*.mnf	none	none	NEW	YES	NO	SEQ.	Interface for ADAMS/Flex
A502LU								Available for Use
DBMIG								Available for Use
USER FILE	REQ	REQ	REQ	REQ	YES	NO	SEQ.	Any User-Defined File

where:

Logical Key Name	specifies the logical-key NAME used on the ASSIGN statement.
Physical Name	specifies the default name used to open the file, i.e., the default filename2 name. “REQ” means that this parameter is required in the ASSIGN statement from the user.
Unit No.	specifies the default FORTRAN unit number used by MSC.Nastran. “REQ” means that this parameter is required in the ASSIGN statement from the user.
Form	specifies the default FORM used when the file is opened.
Status	specifies the default STATUS used when the file is opened. “REQ” means that this parameter is required in the ASSIGN statement from the user.
Assignable	If “YES”, the user may assign a physical file to this logical name. If “NO”, the unit (if any) and logical name are reserved by MSC.Nastran.
Open	If “YES”, the file is opened by default. If “NO”, the file must be explicitly opened.
Access	If “SEQ”, the file is opened for sequential access. If “DIRECT”, the file is opened for direct access.
sdir	The scratch directory specified using the “sdirectory” keyword.

data	The name of the input data file with all directory and extensions removed.
out	The directory and file prefix specified using the “out” keyword or taken by default.

Notes:

\+ The actual logical-key name for this is “OP2”. If you use “OUTPUT2” (even though this is still the logical-key name put out by MSC.Patran) you will get a user fatal message from MSC.Nastran.

* FORMATTED is required for neutral-format OUTPUT2 files and ASCII-format OUTPUT4 files.

++ For Cray Unicos, the default Form is UNFORMATTED. For all other platforms, the Form is ignored. See Remark **12.**

Example

The following example demonstrates the form of the modified ASSIGN and POST entries. The expected result would be that files ‘rsltsubc.subc1’ and ‘rsltsubc.sub2’ would contain the results for subcase 1 and 2 respectively.

```
assign opcase='rsltsubc'
ID MSC, RSLTSUBC
TIME    5   $ MINUTES
SOL     101 $
CEND
TITLE = WRITE RESULTS TO SEPARATE SUBCASE FILES
  ECHO  = UNSORT
  DISPL = ALL
  STRES = ALL
  GPSTR = ALL
post tocase allcase
SUBCASE 1
  TITLE = GPSTR1, QUADR ELEMENTS, MEMBRANE
  subtitle = **** subcase 1 ****
  LOAD  = 1
post tocase subc1
SUBCASE 2
  TITLE = GPSTR1, QUADR ELEMENTS, BENDING
  subtitle = **** subcase 2 ****
  LOAD  = 2
post tocase subc2
OUTPUT(POST)
  SET 1 = ALL
  SURFACE 1, SET 1, FIBRE Z1, SYSTEM BASIC, NORMAL Z, TOPOLOGICAL
BEGIN BULK
GRID    1               .04     .02
```

```
GRID    2               .18     .03
GRID    3               .16     .08
GRID    4               .08     .08
GRID    5               .0      .0                      123456
GRID    6               .24     .0                      123456
GRID    7               .24     .12                     123456
GRID    8               .0      .12                     123456
CQUADR  1       100     1       2       3       4
CQUADR  2       100     5       6       2       1
CQUADR  3       100     6       7       3       2
CQUADR  4       100     7       8       4       3
CQUADR  5       100     8       5       1       4
PSHELL  100     100     .001    100             100
MAT1    100     1.+6            .25
FORCE   1       5               0.      1.
SPCD    1       5       1       0.      5       2       0.
SPCD    1       6       1       2.4-4   6       2       1.2-4
SPCD    1       7       1       3.-4    7       2       2.4-4
SPCD    1       8       1       0.6-4   8       2       1.2-4
FORCE   2       5               0.                      1.
SPCD    2       5       3       0.
SPCD    2       5       4       0.      5       5       0.

SPCD    2       6       3       2.88-5

SPCD    2       6       4       1.2-4   6       5       -2.4-4
SPCD    2       7       3       5.04-5
SPCD    2       7       4       2.4-4   7       5       -3.-4
SPCD    2       8       3       7.2-6
SPCD    2       8       4       1.2-4   8       5       -6.-5
PARAM   NEWSEQ  -1

ENDDATA
```

CHAPTER
10 Upward Compatibility

- DMAP Modules in MSC.Nastran 2005
- Summary of Data Block Changes from MSC.Nastran 2004 to MSC.Nastran 2005
- More Stringent Case Control Check

10.1 DMAP Modules in MSC.Nastran 2005

This section summarizes DMAP module changes from MSC.Nastran 2004 to MSC.Nastran 2005 which could affect user DMAP alters and solution sequences. This information is intended to help convert MSC.Nastran 2004 DMAP alters and solution sequences to run in MSC.Nastran 2005. The format of the following modules has been modified in MSC.Nastran 2005 such that the MSC.Nastran 2004 format is not upwardly compatible with MSC.Nastran 2005 and / or their behavior is not upwardly compatible. The changes are described in the next section.

BDRYINFO	DOPR3	DPD	ELTPRT	FA1	FRLG	GUST
MODQSET	MPP	NLCOMB	NLTRLG	SDRHT	TRLG	WEIGHT

The following is a list of existing modules with new features or fixes which require format changes in MSC.Nastran 2005 but are not documented here because their MSC.Nastran 2004 formats are considered upwardly compatible in MSC.Nastran 2005. They are fully documented in the ***MSC.Nastran 2005 DMAP Programmer's Guide***.

BCDR	DISOPT	DOM11	DOM12	DOM6	DOM9	DOPFS
DOPR1	DSAD	DSAL	DSAW	DSPRM	DSTAP2	FA2
GKAM	GP1	GP4	GPFDR	GPSP	IFP9	INPUTT2
MAKAEFS	MAKMON	MATMOD	MKRBVEC	MODGM2	MPYAD	NLITER
NLSOLV	OUTPRT	SDRCOMP	SEP1X	SEQP	SMPYAD	SSG1
TA1						

The following is a list of new modules in MSC.Nastran 2005. They are not documented here but are documented in the ***MSC.Nastran 2005 DMAP Programmer's Guide***.

DSGRDM	GI2C	GUSTLDW	ILMP1	ILMP2	ILMPGPF	MASSCOMB
MDENZO	MODCASE	NDINTERP	SLITX			

DMAP Module Changes

This section shows the changes for DMAP module instructions which were changed from MSC.Nastran 2004 to 2005. The module change descriptions are presented as differences with respect to the ***MSC.Nastran 2005 DMAP Programmer's Guide*** which is available on the "MSC.Software Combined Documentation 2005" CD-ROM. The change descriptions below includes the MSC.Nastran 2005 format of the module with changes in bold text. Any new or changed data blocks and parameters are also described below the format.

BDRYINFO

The first parameter, ASMUNIT, is obsolete and has been removed.

DOPR3

The UNUSED data block has been removed and DIT and DYNAMIC are moved into the 18th and 19th positions.

Format:

```
DOPR3          CASE,EDOM,DTB,ECT,EPT,DESTAB,EDT,OL,DEQIND,DEQATN,
               BGPDT,DVPTAB*,VIEWTB,OINT,PELSET,XINIT,FOL,DIT,DYNAMIC/
               OBJTAB,CONTAB,R1TAB,RESP12,RSP1CT,FRQRSP,CASEDS,
               OINTDS,PELSETDS,DESELM,RESP3,ADRDUG,ADRDUTB,CASADJ,
               MODRSP,CASEDM,RQATAB,RESP12X,RESP3X,CONTABX,OBJTABX,
               ARVEC/
               DMRESD/S,N,DESGLB/S,N,DESOBJ/S,N,R1CNT/S,N,R2CNT/
               S,N,CNCNT/SOLAPP/SEID/S,N,EIGNFREQ/PROTYP/DSNOKD/
               SHAPES/S,N,R3CNT/RGSENS/INREL/S,N,ADJFLG/S,N,TADJCOL/
               AUTOADJ/SOLADJC/S,N,NORMEV $
```

DPD

The amplitude data in DLT has been moved into five new output matrices to support machine precision enforced motion.

Format:

```
DPD          DYNAMIC,GPL,SIL,USET,UNUSED5,PG,PKYG,PBYG,PMYG,YG/
             GPLD,SILD,USETD,TFPOOL,DLT,PSDL,RCROSSL,NLFT,TRL,
             EED,EQDYN,APPLOD,ENFLODK,ENFLODB,ENFLODM,ENFMOTN/
             LUSET/S,N,LUSETD/S,N,NOTFL/S,N,NODLT/S,N,NOPSDL/
             DATAREC/S,N,NONLFT/S,N,NOTRL/S,N,NOEED/SORTNLFT/
             S,N,NOUE/UNUSED12/SEID $
```

Output Data Blocks:

APPLOD	Matrix of applied load amplitudes
ENFLODK	Matrix of equivalent enforced motion load amplitudes due to stiffness effects
ENFLODB	Matrix of equivalent enforced motion load amplitudes due to viscous damping effects
ENFLODM	Matrix of equivalent enforced motion load amplitudes due to mass effects
ENFMOTN	Matrix of enforced motion amplitudes

ELTPRT

ECT is always input in the 12st position and no longer input in the 4th position, which is now occupied by the new data block NSMEST.

Format:

```
ELTPRT      ECT,GPECT,BGPDT,NSMEST,EST,CSTM,MPT,DIT,
            CASSECC,EPT,UNUSED/
            VELEM/
            PROUT/S,N,ERROR/WTMASS $
```

Input Data Blocks:

NSMEST	NSM Bulk Data entries in EST format
UNUSED	Unused and may be purged.

FA1

VREF parameter is now required input.

Format:

```
FA1         KHH,BHH,MHH,QHHL,CASECC,EDT/
            FSAVE,KHH1,BHH1,MHH1,FLUTABP/
            S,N,FLOOP/S,N,TSTART/S,N,NOCEAD/LPRINT/XYUNIT/VREF $
```

Output Data Block:

FLUTABP

Flutter summary table for all methods except K and KE

Parameters:

XYUNIT	Input-integer-default=0. FORTRAN unit number to which extracted khh1 values are written at each sweep point for the PKS and PKNLS methods.
VREF	Input-real-no default. Flutter velocity divisor to obtain flutter indices.

FRLG

Format:

```
FRLG        CASECC,USETD,DLT,FRL,GMD,GOD,DIT,PHDH,
            APPLOD,ENFLODK,ENFLODB,ENFLODM,ENFMOTN/
            PPF,PSF,PDF,FOL,PHF,YPF/
            SOLTYP/S,N,FOURIER/S,N,APP $
```

Input Data Blocks:

APPLOD	Matrix of applied load amplitudes
ENFLODK	Matrix of equivalent enforced motion load amplitudes due to stiffness effects
ENFLODB	Matrix of equivalent enforced motion load amplitudes due to viscous damping effects
ENFLODM	Matrix of equivalent enforced motion load amplitudes due to mass effects
ENFMOTN	Matrix of enforced motion amplitudes

GUST

Format:

```
GUST         CASECC,DLT,FRL,DIT,QHJL,UNUSED6,UNUSED7,ACPT,
             CSTMA,PHF,APPLOD,ENFLODK,ENFLODB,ENFLODM,ENFMOTN/
             PHF1,WJ,QHJK,PFP/
             S,N,NOGUST/BOV/MACH/Q $
```

Input Data Blocks:

APPLOD	Matrix of applied load amplitudes
ENFLODK	Matrix of equivalent enforced motion load amplitudes due to stiffness effects
ENFLODB	Matrix of equivalent enforced motion load amplitudes due to viscous damping effects
ENFLODM	Matrix of equivalent enforced motion load amplitudes due to mass effects
ENFMOTN	Matrix of enforced motion amplitudes

MODQSET

Format:

```
MODQSET      GEOM1,GEOM2,GEOM4/
             GEOM1W,GEOM2W,GEOM4W/
             NOQSETT/QSETREC/QSETID $
```

Parameters:

QSETREC — Input-integer-default=0. Records to use in defining the q-set degrees-of-freedom:

>=0	No records are written
=-1	SENQSET record to GEOM1W
=-2	SPOINT record to GEOM2W and QSET1 record to GEOM4W

QSETID — Input-integer-default=0. Starting q-set identification number for QSETREC=-2.

MPP

MP2S inserted after MPSRP.

Format:

```
MPP          AECTRL,UXDAT, {MONITOR } , {MPSR} , {MPSER} ,MPEU,
                           {AEMONPT }   {MPAR}   {MPAER}

             MPSIR,MPSRP,MP2S,MPSERP,UXV,INDX//
             MACH/Q/AECONFIG/SYMXY/SYMXZ/MESH $
```

Input Data Block:

MP2S	Table of MONPNT2 responses at trim

NLCOMB

Format:

```
NLCOMB       CASECC,ESTNL,KDICTNL,BKDICT,ETT,PTELEM0,PTELEM,UNUSED8,
             MPT,EQEXIN,SLT,DLT,BGPDT,APPLOD,DYNAMIC/

             ELDATA,{SLT1}  /
                    {DLT1}

             NSKIP/LSTEP/LINC/STATIC/LGDISP/OSTEP $
```

Input Data Blocks:

APPLOD	Matrix of applied load amplitudes
DYNAMIC	Table of Bulk Data entry images related to dynamics.

NLTRLG

Format:

```
NLTRLG       CASECC,USETD,DLT,SLT,BGPDT,SIL,CSTM,TRL,DIT,GMD,GOD,
             PHDH,EST,MPT,APPLOD,ENFLODK,ENFLODB,ENFLODM,ENFMOTN/
             PDT,TMLD,DLT1/TABS $
```

Input Data Blocks:

APPLOD	Matrix of applied load amplitudes
ENFLODK	Matrix of equivalent enforced motion load amplitudes due to stiffness effects
ENFLODB	Matrix of equivalent enforced motion load amplitudes due to viscous damping effects
ENFLODM	Matrix of equivalent enforced motion load amplitudes due to mass effects
ENFMOTN	Matrix of enforced motion amplitudes

SDRHT

Format:

```
SDRHT          UG,OEF1,SLT,EST,DIT,RDEST,RECM,DLT,OEFNL1,MPT,BGPDT,
               CSTM,SIL,USET,CASECC,OESNLH,APPLOAD/
               HOEF1,HOES1/
               TABS/SIGMA/NORADMAT $
```

Input Data Blocks:

APPLOD	Matrix of applied load amplitudes
OESNLH	Table of element heat flow in SORT1 format for nonlinear elements

Output Data Block:

HOES1	Table of element heat flow in SORT1 format combined for linear and nonlinear elements.

TRD1

Format:

```
TRD1           CASECC,TRL,NLFT,DIT,KXX,BXX,MXX,PXT,SILD,USETD,
               PARTVEC,PXT0,ROTORT,BGDD*,KCVDD*,RDG,PXTDV
               UXT,PNL/
               SOLTYP/NOUE/NONCUP/S,N,NCOL/FAC3/SETNAME/
               NSOLT,NOTRLDFM/WTMASS $
```

Input Data Block:

PXTDV Transient response load matrix in h-set (modal) or d-set combined from two executions of TRLG: one with DVFLAG=0 and the other of DVFLAG=1.

TRLG

Format:

TRLG CASECC,USETD,DLT,SLT,BGPDT,SIL,CSTM,TRL,DIT,

$$\left\{\begin{matrix}\texttt{GMD}\\ \ \end{matrix}\right\},\left\{\begin{matrix}\texttt{GOD}\\ \ \end{matrix}\right\},\left\{\begin{matrix}\texttt{PHDH}\\ \texttt{RPX}\end{matrix}\right\}\texttt{,EST,MPT,MGG,V01P,}$$

APPLOD,**ENFLODK**,**ENFLODB**,**ENFLODM**,**ENFMOTN**/

$$\left\{\begin{matrix}\texttt{PPT}\\ \texttt{PPT}\end{matrix}\right\},\left\{\begin{matrix}\texttt{PST}\\ \ \end{matrix}\right\},\left\{\begin{matrix}\texttt{PDT}\\ \ \end{matrix}\right\},\left\{\begin{matrix}\texttt{PDT1}\\ \ \end{matrix}\right\},\left\{\begin{matrix}\texttt{PHT}\\ \texttt{PXT}\end{matrix}\right\}\texttt{,TOL,DLTH,YPT,YPO,TOLR/}$$

S,N,NOSET/S,N,PDEPDO/IMETHOD/STIME/BETA/
S,N,FAC1/S,N,FAC2/S,N,FAC3/TOUT/TABS/
STIME/S,N,NCOLT/S,N,NSOLT/**DVFLAG** $

Input Data Blocks:

APPLOD Matrix of applied load amplitudes

ENFLODK Matrix of equivalent enforced motion load amplitudes due to stiffness effects

ENFLODB Matrix of equivalent enforced motion load amplitudes due to viscous damping effects

ENFLODM Matrix of equivalent enforced motion load amplitudes due to mass effects

ENFMOTN Matrix of enforced motion amplitudes

Parameter:

DVFLAG Input-integer-default=0. Enforced motion processing flag for both the large mass and direct methods of specification.

= 0 Process only applied loads and excitations due to enforced accelerations (default)

> 0 Process only excitations due to enforced displacements and velocities.

WEIGHT

The SEID parameter has been moved to the 4th position.

Format:

```
WEIGHT         VELEM,EST,MPT,DIT,OPTPRM,OGPWG,DESTAB,XINIT/
               WMID,WGTM/
               WGTVOL/S,N,VOLS/S,N,FRMS/SEID/DOFRMASS $
```

Input Data Blocks:

DESTAB	Table of design variable attributes.
XINIT	Matrix of initial values of the design variables.

Parameters:

FRMASS	Output-real-default=0.0. Fractional mass of designed structure.
SEID	Input-integer-default=0. Superelement identification number.
DOFRMASS	Input-integer-default=0. Fractional mass flag.

10.2 Summary of Data Block Changes from MSC.Nastran 2004 to MSC.Nastran 2005

The following material describes changes in table data blocks for only those records that existed in MSC.Nastran 2005.

DYNAMIC - Table of Bulk Data entry images related to dynamics

Added the following real-valued words for initial displacement and velocity factors after the TID item on the TLOAD1 record and after the B item on the TLOAD2 record after.

Item	Type	Description
U0	Real	Initial displacement factor for enforced motion
V0	Real	Initial velocity factor for enforced motion

GEOM2 - Table of Bulk Data entries related to element connectivity

Changed names and header words for the following records:

MSC.Nastran 2004		MSC.Nastran 2005	
Name	Header Words	Name	Header Words
BEAMAERO	(2601,26,0)	BEAMAERO	(1701,17,0)
CHEXAL	(7708,77,369)	CHEXAL	(7908,79,369)
Q4AERO	(3002,46,0)	AEROQ4	(2002,20,0)
T3AERO	(2701,27,0)	AEROT3	(1801,18,0)

GEOM4 - Table of Bulk Data entry images related to constraints

Changed the type of the enforced displacement value, D on the SPCD and SPC records from single-precision to machine-precision. Also added a null item in front preceding D.

RESP12 - Table of second level (synthetic) responses

Inserted four more items after word 5 (item REG):

Item	Type	Description
METH	Integer	Method flag for BETA / MATCH responses
C1	Real	Constant to scale beta responses
C2	Real	Constant to scale distance responses
C3	Real	Constant to shift lower bound

10.3 More Stringent Case Control Check

Previous versions of MSC.Nastran check the first 4 characters of the Case Control command. For example, to request displacement output, any one of the following commands is acceptable prior to MSC.Nastran 2005.

```
Disp = all
Displ = all
Displa = all
Displacement = all
Displcement = all
```

Starting in MSC.Nastran 2005, the full spelling is checked. Correctly spelled short forms are still acceptable (e.g., disp, etc.) For the above example, the first four commands are acceptable. The last one, due to the misspelled command, will cause the job to terminate with the following fatal message:

```
*** USER FATAL MESSAGE 601 (IFP1D)
     THE KEYWORD ON THE ABOVE CARD TYPE IS ILLEGAL OR MISSPELLED.
```

This change is necessary to alleviate problems caused by the lack of uniqueness for checking only 4 characters for certain commands. In versions prior to MSC.Nastran 2005, the following 2 commands

```
Elstress = all
Elstrain = all
```

are both treated as element stresses (elst). Starting in MSC.Nastran 2005, the command, ELSTRAIN, will terminate the job with UFM 601 as elstrain is a non-existing command.

INDEX

T

U

V

W