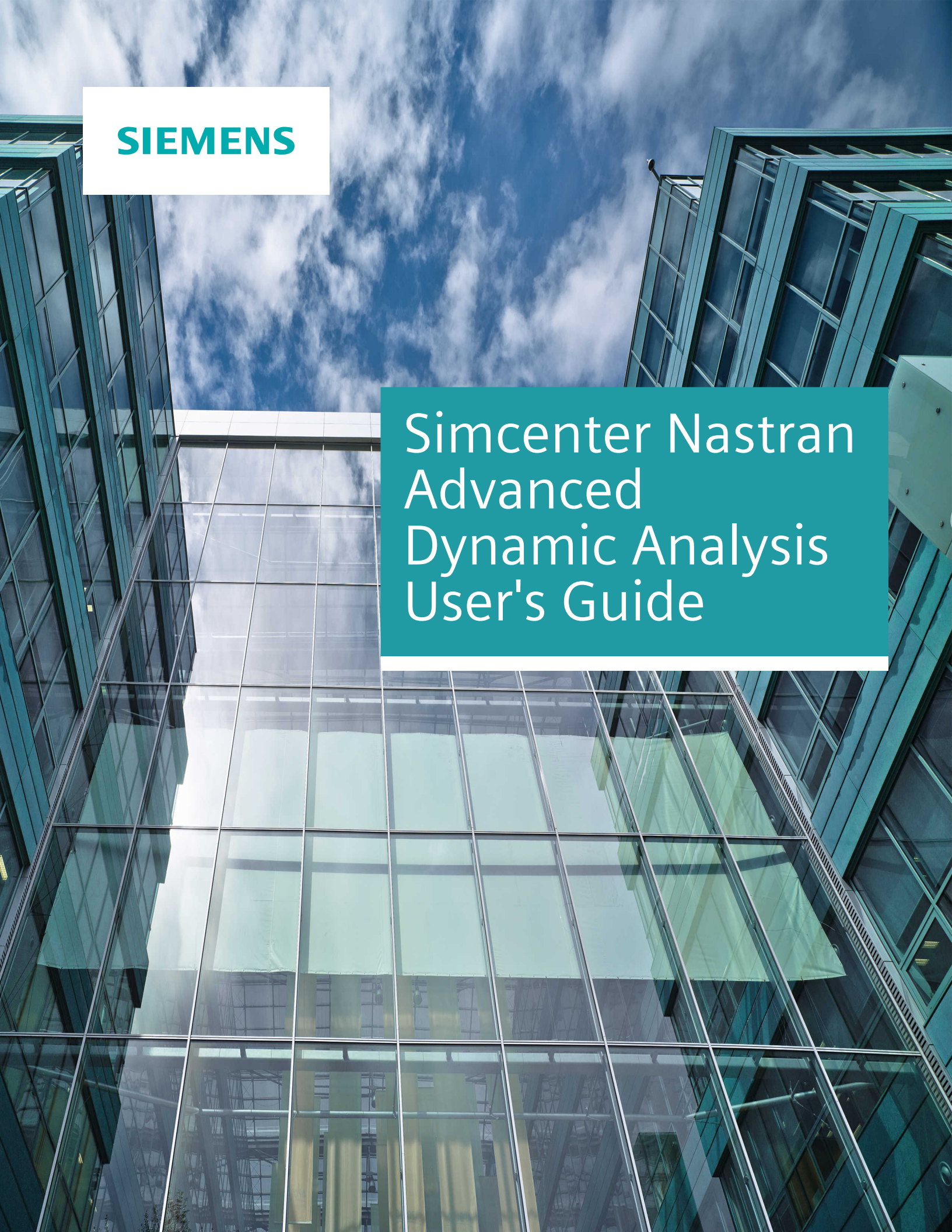



# Simcenter Nastran Advanced Dynamic Analysis User's Guide

# Contents

## Boundary Conditions and Coupling to Other Disciplines



## Dynamic Solution Techniques

# Proprietary & Restricted Rights Notice



**TAUCS Copyright and License**



Availability (TAUCS)

As of version 2.1, we distribute the code in 4 formats: zip and tarred-gzipped (tgz), with or without binaries for external libraries. The bundled external libraries should allow you to build the test programs on Linux, Windows, and MacOS X without installing additional software. We recommend that you download the full distributions, and then perhaps replace the bundled libraries by higher performance

ones (e.g., with a BLAS library that is specifically optimized for your machine). If you want to conserve bandwidth and you want to install the required libraries yourself, download the lean distributions. The zip and tgz files are identical, except that on Linux, Unix, and MacOS, unpacking the tgz file ensures that the configure script is marked as executable (unpack with tar zxvpf), otherwise you will have to change its permissions manually.

**HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 2006-2016 by The HDF Group**

NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted for any purpose (including commercial purposes) provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or materials provided with the distribution.

3. In addition, redistributions of modified forms of the source or binary code must carry prominent notices stating that the original code was changed and the date of the change.

4. All publications or advertising materials mentioning features or use of this software are asked, but not required, to acknowledge that it was developed by The HDF Group and by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign and credit the contributors.

5. Neither the name of The HDF Group, the name of the University, nor the name of any Contributor may be used to endorse or promote products derived from this software without specific prior written permission from The HDF Group, the University, or the Contributor, respectively.

DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS "AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.

In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage

# 1. Overview

Simcenter Nastran provides a comprehensive set of capabilities for structural dynamics analysis. Simcenter Nastran provides dynamic analysis capabilities which are widely applicable to a wide range of problems and industries, including aerospace, earthquakes, automobiles, nuclear power plants, and many consumer products.

The guide provides recommended procedures for solving more complex problems with structural dynamics. The *Simcenter Nastran Basic Dynamic Analysis User's Guide* provides instructions on the typical dynamics problems and use of the basic input data. This guide covers many aspects that fall beyond the scope of the basic guide as well as new topics related to other Simcenter Nastran capabilities (such as aeroelasticity, optimization, or random analysis). The basic principles also apply to related applications such as cyclic symmetry.

# 2. Dynamic Modeling Options

## 2.1 Overview of Dynamic Modeling Options

In general, the practice of building Simcenter Nastran models for dynamic analysis is similar to building statics or normal modes models. However, in dynamics the higher costs of running detailed large order models usually results in a different approach. For instance, a frequency response analysis using SOL 108 may require solutions at more than 100 frequencies. Each frequency will require an unsymmetrical matrix solution using complex variables, which will cost four to ten times more than a single static solution. If memory is sufficient, the total result is a running time larger by a factor of 400 to 1000. If the required memory (four times more is needed) is insufficient, the solution will spill, and even longer run times will result. In addition, the database and scratch file space requirements will be much larger and the size of the output data becomes unwieldy. The result is that many users who simply take fine mesh static models directly into a dynamic analysis find this process is very costly and may give up completely.

### Strategies for Dynamic Analysis

The two major approaches to avoid the problems described above are to use the matrix reduction methods available in Simcenter Nastran or to simply build another dynamic model with fewer points and elements. The matrix reduction metho ds use sophisticated approximations to reduce the cost of the dynamic solutions. However, they add complexity to the system and may also cost processing time overhead. Instead of making decisions about the element mesh size, the user must consider the frequency band and loading characteristics of the actual structure.

Although building a separate dynamic model will require extra work, it has several advantages. The requirements for precise stress results for statics are usually not as important for dynamics, thereby allowing the use of larger elements. The requirements for accurate dynamic results are not as dependent on small elements since loads tend to be distributed over wider areas by the inertial and damping effects. The actual accuracy criteria for dynamic models is related to wavelength size relative to the mesh size. Also, in a smaller dynamic model the job of interpreting outputs with plotters and other methods becomes much easier.

It should also be noted that the process of changing finite element meshes is becoming easier with modern geometry-based modeling systems and automated mesh generation.

### Plan the Analysis

The following steps are recommended before starting a dynamic analysis:

1. Estimate the frequency range of the structure. The first mode frequency can usually be estimated by a simple equivalent beam calculation. A normal modes analysis is highly recommended for dynamics regardless of the approach.

2. Consider the type and frequency content of the loads. Localized high frequency sources such as brake squeal require different approaches from low frequency distributed loads such as wind forces on a bridge.

3. Use expected wavelengths to estimate required mesh size. A wavelength can be estimated from the frequency and sound speed. Six elements per wave for approximately 10% accuracy is a general rule-of-thumb. Also, this method may be used to evaluate the quality of the calculated eigenvectors.

4. Always use SETs for output requests as a general practice. The use of Case Control requests such as STRESS = ALL for a large transient analysis could possibly exceed the size limits of your postprocessor.

5. Estimate the computer resources (CPU time, database size, output requirements) for large problems before production runs are made. A good method is to run the model with only a few time steps or frequencies, and examine the major time and size messages in the F04 output file. The actual costs can then be extrapolated with reasonable accuracy.

The basic modeling, constraints, and loading functions for dynamics are described in the *Simcenter Nastran Basic Dynamic Analysis User'sGuide*. The following sections discuss some of the more difficult and obscure topics.

## 2.2 Mass Modeling

The mass matrix in Simcenter Nastran may contain much more information than simple structural mass and inertia terms. In fact, it may contain control system terms, fluid compressibility, or electromagnetic capacitance. One basic definition is that any term which contributes to the generalized kinetic energy must create a coefficient in the mass matrix. Another definition is that any generalized force, *F*, proportional to an acceleration term ū produces a mass term *M*, i.e.,

$$\{F\} = [M]\{\ddot{u}\}$$

**Equation 2-1.**

where each component of the acceleration vector {ū} represents a generalized degree-of-freedom.

The mass matrix is required for nearly all dynamic solution sequences. It is also used for generating gravity and centrifugal loads for static solutions. Inertia relief solutions require the mass matrix to balance the unbalanced forces on a free body. All solutions may calculate the total mass and center of gravity (CG) information for printout.

Note that mass is not required for heat transfer dynamics.

### Mass Data Input Options

Simcenter Nastran provides the following means to specify the mass properties of the finite element model:

1. The density (mass per unit volume) of the structural materials, which comprises the finite element (RHO on the MATi Bulk Data entries).

2. Nonstructural mass per unit length of line elements or mass per unit area of surface elements (NSM on the property Bulk Data entry). Examples of this feature are coatings and thermal insulating materials.

3. Concentrated mass and inertia terms at a grid point via a CONM2 Bulk Data entry. The provisions of the CONM2 entry are the mass, the offset of the center of mass from the grid point, and the moments and products of inertia about the center of mass. As an option, the center of mass may be measured from the origin of the basic coordinate system rather than as an offset from the grid point.

4. A full 6 × 6 symmetric matrix of mass coefficients at a grid point via the CONM1 Bulk Data entry.

5. Mass coupling between any two degrees-of-freedom via the CMASSi (i = 1,2,3,4) Bulk Data entries. The form of the relationship is

$$\begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = -\begin{bmatrix} M & -M \\ -M & M \end{bmatrix} \begin{Bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \end{Bmatrix}$$

**Equation 2-2.**

where $f_1$ and $f_2$ are the inertia forces acting at degrees-of-freedom 1 and 2, respectively, and $M$ is the mass coefficient, specified on the CMASSi entry (or on the PMASS entry if i = 2 or 4). In most applications, the second degree-of-freedom is not specified. In this case, the entry generates the inertia force $f_1 = -M\ddot{u}_1$, and $M$ is added to the mass matrix in the diagonal position corresponding to $u_1$. An important application of the CMASSi entry occurs in the recommended method for specifying enforced motion at grid points (see **"Enforced Motion with Loads"** ).

6. Transfer functions defined on the TF Bulk Data entry may contribute terms to the mass matrix.

7. Direct Matrix inputs may be added to the mass matrix via DMIG Bulk Data and M2GG = or M2PP = Case Control commands.

### Three Related Parameters

The bulk data input also includes three parameters which relate to the specification of mass properties. They are:

1. PARAM,WTMASS,V1 – Specifies a factor by which to multiply the mass of the structure to obtain dimensionally correct mass. For example, if the ft-lb-sec system is used, and the mass input unit is pounds, then V1 = 1/32.174 = .031081. This factor operates on all mass terms (except those defined on DMIG entries; PARAM,CM2,V1 may be used for these instead).

2. PARAM,COUPMASS,1 – Requests that the coupled mass option be used rather than the lumped mass option for all elements which support the coupled mass option. In the lumped mass option, the distributed mass of an element is simply divided up and the parts are located at the connected grid points. In the coupled mass option, the distributed mass is replaced by a nondiagonal mass matrix which couples the connected grid points. The latter method is sometimes more accurate, but it uses more computer time. See the *Simcenter Nastran User'sGuide* for descriptions of the elements.

3. PARAM,GRDPNT,V1 – Causes the Grid Point Weight Generator to be executed. The value of the parameter, V1, is an integer which identifies a grid point at which the rigid body mass properties of the structure will be computed. This capability can be used as a check on mass and geometric input data.

## Coupled Mass Matrix Terms

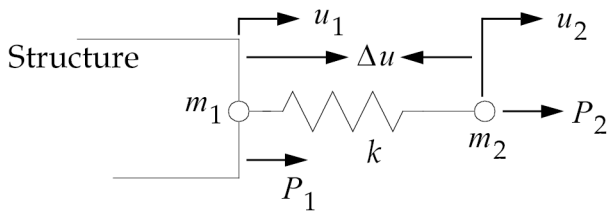
A common difficulty for finite element beginners is to comprehend the generation of off-diagonal or coupled terms in the mass matrix. These are caused by the following actions.

1. With PARAM,COUPMASS, the finite elements themselves generate coupled terms to represent the distributed element mass more accurately than simple lumped masses on the grid points. The total kinetic energy of the element is represented by shape functions, which in turn, are coupled functions of displacement and rotation.

2. The internal matrix operations in the solution sequences produce coupled mass terms. MPCs and RBE-type elements cause the mass to be redistributed from the dependent degrees-of-freedom to the remaining independent points. The ASET, OMIT, and Superelement reduction operations may automatically fill the entire mass matrix. These operations will maintain exact total mass and CG properties and provide a good approximation to the inertia effects of the eliminated points, but they can be costly.

3. Direct user inputs such as CMASSi scalar elements and CONMi mass elements may produce off-diagonal mass terms. Other inputs that cause coupling are direct matrix inputs (DMIG), and transfer functions (TF).

An example of a coupled scalar mass is illustrated in the example below:




A spring, $k$, and two masses, $m_1$ and $m_2$, are attached to a structural model. However, when modeling fluids or other special connections, the user may wish to use the difference in displacements, $\Delta u=u_1-u_2$, instead of $u_2$ as the primary degree-of-freedom.

We may find the equivalent coupled mass matrix by the use of energy methods. The potential energy, *V*, and the kinetic energy, *T*, for this part of the system are:

$$V = k\Delta u^2 / 2$$

$$T = m_1 \ddot{u}_1^2 + m_2(\ddot{u}_1 - \Delta\ddot{u})^2$$

**Equation 2-3.**

From basic energy principles, the resulting equilibrium equations are

$$\ldots + m_1\ddot{u}_1 + m_2(\ddot{u}_1 - \Delta\ddot{u}) = P_1 + P_2$$

and

$$k\Delta u - m_2(\ddot{u}_1 - \Delta\ddot{u}) = -P_2$$

**Equation 2-4.**

The mass matrix for the two degrees-of-freedom becomes

$$[m] = \begin{bmatrix} m_1 + m_2 & -m_2 \\ -m_2 & m_2 \end{bmatrix}$$

**Equation 2-5.**

where $[u] = [u_1, \Delta u]$ are the active degrees-of-freedom. Two CMASSi elements may be used; one will be coupled to two points. Note that the spring, *k*, is now connected to $\Delta u$ only.

An alternate method would be to use the MPC constraints to produce the same effect. Simply include all three DOFs in the model and use conventional lumped masses and a connected spring. Choose $u_2 - u_1 + \Delta u = 0$ as the MPC equation, and the reduced mass and stiffness matrices will be identical to the system above.

## Mass Effects in Moving Coordinate Systems

In most cases the motions of the grid points in Simcenter Nastran are measured in a set of fixed coordinate systems. Even large displacement nonlinear analysis uses displacements measured in fixed directions. However, there are several cases in which it is more convenient to use a moving coordinate system. Examples follow:

## Spinning Bodies

If the entire structure is spinning at a constant angular velocity, both static centrifugal forces and coupled dynamic inertial terms may be needed. An example is a high speed turbine wheel that exhibits gyroscopic stability problems.

For dynamic analysis of rotating bodies, a special solution will be provided in the DMAP alter library for the generation of extra matrix terms caused by the rotating coordinates. For a body rotation defined by the angular velocity vector, $\vec{\Omega}$, and for a location vector of a point, $\vec{r}$, the absolute velocity vector of the point, $\vec{V}$, in fixed coordinates is

$$\vec{V} = \vec{\Omega} \times (r + \vec{u}) + \dot{\vec{u}}$$

**Equation 2-6.**

The acceleration vector is

$$\vec{A} = \vec{\Omega} \times \left(\vec{\Omega} \times \vec{r}\right) + \vec{\Omega} \times \left(\vec{\Omega} \times \vec{u}\right) + 2\left(\vec{\Omega} \times \dot{\vec{u}}\right) + \ddot{\vec{u}}$$

**Equation 2-7.**

The first term on the right-hand side of **Eq. 2-7** is the static centrifugal force; the second term is the centripetal stiffness; the third term is the Coriolis force; the last term, $\ddot{\vec{u}}$, is the relative acceleration vector.

Another term that is calculated for spinning bodies is the so-called differential stiffness matrix. It is proportional to the steady centrifugal preloads in the elements. These are the terms that would stiffen a string if a weight on the string were swung in a circular motion. These terms are important because they are approximately the same magnitude as the Coriolis and centripetal stiffness terms defined in **Eq. 2-7**. (Centrifugal stiffness and differential stiffness terms are of the same magnitude.)

The basic matrix equation for the forces in the moving system is

$$[M]\{A\} + [B]\{V\} + [K]\{u\} = \{P\}$$

**Equation 2-8.**

The special Simcenter Nastran process assembles the terms in **Eq. 2-7** into matrices, that when substituted into **Eq. 2-8**, result in

$$[M]\{\ddot{u}\} + [B + B^c]\{\dot{u}\} + [K + K^c + K^d]\{u\} = \{P(t)\}$$

**Equation 2-9.**

where:

$[B^c]$ = generates the velocity-dependent Coriolis forces. Note that $B^c$ is not symmetric!

$[K^c]$ = the centripetal stiffness matrix

$[K^d]$ = the differential stiffness matrix

The system defined by **Eq. 2-9** may be solved with a transient analysis, a frequency response, or a complex eigenvalue calculation. The complex eigenvalues may be obtained for a series of spin rates to determine the critical angular speeds. At each angular velocity the complex roots are obtained in the form $p_n = \sigma_n \pm i\omega_n$. Unstable conditions usually occur when $\omega_n \sim P\Omega$, where $P$ is a positive integer. By definition, the system is unstable if $\sigma_n$ is a positive number.

## Inertia Relief

If a free body is accelerating due to constant unbalanced loads, the inertia relief solution provides the ability to obtain static deflections relative to a set of reference points attached to the moving coordinate system. An example is an airplane in a steady turn or accelerating dive. Although this capability is a static solution, it is obtained from the dynamics theory.

The basic matrix equation for the inertia relief method is

$$[K]\{u\} = \{P\} - [M]\{a_o\}$$

**Equation 2-10.**

where $\{u\}$ are displacements relative to the moving system and $\{a_0\}$ are the steady accelerations to be determined from the mass and loads. If $[D]$ is a matrix whose columns define the rigid body motions of the structure, then for a free body,

$$[D]^T[K]\{u\} = \{0\} = [D]^T\{P\} - [D]^T[M]\{a_o\}$$

**Equation 2-11.**

where $[D]$ is called the rigid body transformation matrix. However, since the full-sized vector, $\{a_o\}$, is a rigid body motion, it may be defined in terms of accelerations at a set of reference coordinates, $\{a_r\}$, by the equation

$$\{a_o\} = [D]\{a_r\}$$

**Equation 2-12.**

Combining **Eq. 2-12** into **Eq. 2-11** and eliminating $\{a_r\}$, we obtain

$$\{a_o\} = [D][m]^{-1}[D]^T\{P\}$$

**Equation 2-13.**

where the total mass matrix for the reference coordinates is

$$[m] = [D]^T[M]\{D\}$$

**Equation 2-14.**

The resulting set of equations defined in **Eq. 2-10** may now be arbitrarily constrained since the total load is balanced by the inertia forces.

## Base Excitations

One of several methods to solve problems with enforced motion is to constrain the point of motion and solve the problem in the accelerating system. This method is related to inertia relief but uses entirely different inputs. It is easy to use for earthquake analysis of buildings, in which the base is accelerating uniformly.

If a structure is attached to a semi-rigid base that causes a known stress-free motion, $\{u_0\}$, the total structural motion, $\{u_A\}$, is

$$\{u_A\} = \{u_0\} + \{u_g\}$$

**Equation 2-15.**

where $\{u_g\}$ are displacements relative to the base motion. If the structure is not constrained elsewhere, we may assume that the $\{u_0\}$ base displacements produce no force and

$$[M]\{\ddot{u}_g + \ddot{u}_0\} + [K]\{u_g\} = \{P_g\}$$

**Equation 2-16.**

If we move the known base motion $\{\ddot{u}_0\}$ to the right hand side it looks almost identical to a gravity load:

$$[M]\{\ddot{u}_g\} + [K]\{u_g\} = \{P_g\} - [M]\{\ddot{u}_0(t)\}$$

**Equation 2-17.**

**Acceleration Method**

If the acceleration is uniform over the structure, a time-dependent gravity load has exactly the same form as the last term of **Eq. 2-17**, and the GRAV input load data may be used.

An example data input for a dynamic acceleration load is shown below. For more details on these input formats, see the "**Simcenter Nastran Basic Dynamic Analysis User's Guide** ".

| **Case Control** | |
|---|---|
| LOADSET = 20 | $ Requests LSEQ Id. 20 Process |
| DLOAD = 200 | $ Requests Dynamic Load #200 |
| **Bulk Data** | |
| GRAV, 386, , 386.4, -1.0 | $ Defines Gravity Load in -x direction |
| LSEQ, 20, 201, 386 | $ Assembles GRAV load vector$ Added to DAREA Id. 201 |
| TLOAD1, 200, 201, etc. | $ Dynamic Load using DAREA Id. 201 |

The net result is that the time-dependent inertia loads are applied to all points on the structure in the –x direction in proportion to the time-dependent function specified on the TLOADi or RLOADi Bulk Data entry. The base points should be constrained and the displacements will be calculated relative to the moving base. However, note that the accelerations output from the solution will also be relative to the base motion and should be corrected before being compared with accelerometer data.

A simpler alternative to base motion is the large mass approach, described in **"Enforced Motion with Loads"** .

## 2.3 Modeling Damping Effects

The physical causes of damping in dynamic analysis are any processes which dissipate energy or reduce the structural response through internal friction. Furthermore, the internal velocities or displacements cause reactive damping forces which are irreversible and nonconservative. Examples are mechanical devices such as shock absorbers, the internal hysteresis that occurs in materials such as rubber, friction in joints, and other nonlinear effects such as plastic strains in metals.

The four types of damping in Simcenter Nastran are viscous, structural, modal, and nonlinear. The basic input formats and applications for the damping coefficients in Simcenter Nastran are explained in in the

*Simcenter Nastran Basic Dynamic Analysis User's Guide*. The following discussions are directed to advanced applications and the special problems of damping.

## Viscous Damping

The primary method for modeling viscous damping is through the CVISC and CDAMPi Bulk Data entries. These produce forces which are linearly proportional to the velocities of the connected grid points. Another source of viscous damping is structural damping that must be converted to viscous damping in the transient response solutions. The *Simcenter Nastran Basic Dynamic Analysis User's Guide* covers this topic. Some additional details are included below.

## Structural Damping

Structural damping is intended to simulate the effects of linear material energy loss proportional to the strains. In other words, this method approximates effects similar to hysteresis. It is specified in the material definition input, MATi, and on the parameter, *G*. Note that in the frequency response and complex eigenvalue solutions, the structural damping produces imaginary numbers in the complex stiffness matrix. In the transient solutions, the matrix terms are converted to equivalent viscous damping; i.e., the forces will be proportional to the derivative of strain with respect to time.

The physical basis for the phenomenon of imaginary stiffness is explained by **Figure 2-1**. Shown is the force due to a stiffness matrix term, *K*, plotted against the displacement of the point. As the displacements cycle sinusoidally, the imaginary stiffness, *G*, causes a phase lag in the force response resulting in an elliptical path. The area enclosed by the curve is equal to the dissipated energy.

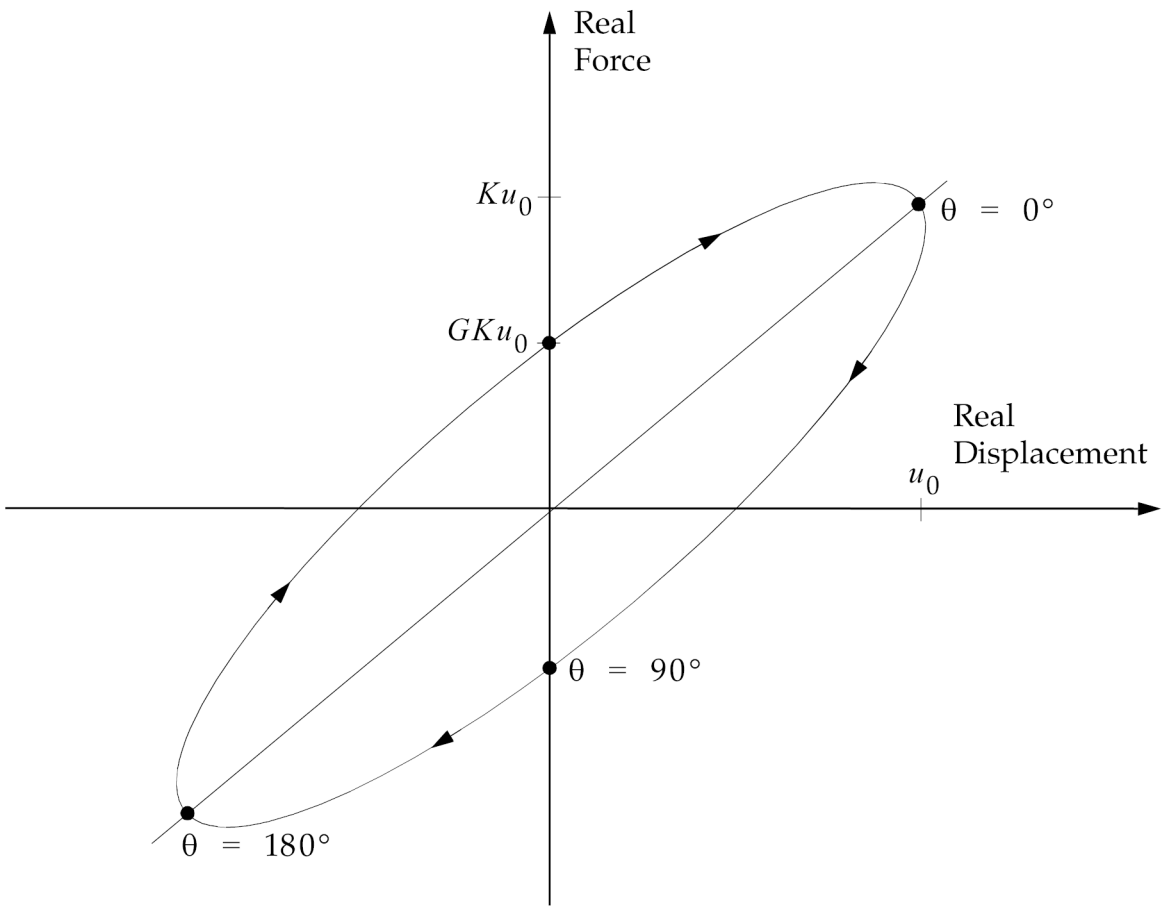



**Figure 2-1. Equivalent Hysteresis Path for Imaginary Stiffness Damping**

The basic definition for the steady-state actual displacements, in terms of the complex displacements is

$$u_r(t) = Re(u_0(\omega)e^{i\omega t})$$

**Equation 2-18.**

where $u_r$ is the actual real displacement, $u_0$ is normally a complex variable that we will set to a real number, $\omega$ is the steady state frequency, and $e^{ix} = \cos(x) + \sin(x)$. If a structural damping coefficient, $G$, is added to the stiffness matrix, the resulting complex forces are in the following form:

$$F(\omega) = (1 + iG)Ku_0e^{i\omega t}$$

**Equation 2-19.**

The real part of the forces are

$$F_r = Re(F) = Ku_0(\cos\omega t - G\sin\omega t)$$

**Equation 2-20.**

The incremental work done over a period of time is

$$dW = F_r du_r$$

**Equation 2-21.**

where, from **Eq. 2-18**

$$du_r = -(u_0\sin\omega t)\omega dt$$

**Equation 2-22.**

Combining **Eq. 2-20** through **Eq. 2-22** and integrating over a full cycle, results in the work

$$W = \int_0^{2\pi/\omega} -Ku_0^2\sin\omega t(\cos\omega t - G\sin\omega t)\omega dt$$

**Equation 2-23.**

Evaluating the integral, we obtain work loss per cycle

$$W = \pi GKu_0^2$$

**Equation 2-24.**

Note that the elastic energy terms average zero over the interval, but the energy dissipated by the structural damping exists. The area inside the curve in **Figure 2-1** is equivalent to the damping energy loss, *W*.

This damping method is a reasonable linear approximation to the classical hysteresis effect. It may also be used to approximate other similar cyclic energy losses such as the effects of loose joints and fasteners. (Hint: construct a similar trajectory curve and estimate the area.)

The main disadvantage of structural damping is that complex numbers must be converted to real numbers in transient analysis; i.e., structural damping is converted to linear viscous damping. This is good only when the response is dominated by a single known frequency.

## Modal Damping

For the modal solution formulations, a special damping input is provided in addition to the other damping terms described above. It is used primarily for efficiency and when test results or contract specifications provide damping factors. These terms are applied only to the uncoupled modal equations. When coupling effects are absent, the method avoids expensive matrix calculations.

A second reason for using modal damping is that modal testing may provide accurate damping inputs on a mode-by-mode basis. These damping factors may be converted to a frequency-dependent table, TABDMP. If the Simcenter Nastran normal mode frequencies are close and the damping factors are small. The third reason—when damping factors are specified by a third party—is frequently meant to keep the analyst more conservative. By restricting the solutions to artificially low damping, this will force the staff to solve their design problems for the worst case.

Modal damping is available only for the modal solutions method. The matrix terms generated for modal analysis, and added to any direct matrix inputs, are

$$m_i \ddot{\xi}_i + b_i \dot{\xi}_i + k_i \xi_i = P_i$$

**Equation 2-25.**

In this equation, $\xi_i$ is the generalized coordinate of the i-th mode.

In accordance with modal definitions, the modal viscous damping coefficient $b_i$ may be expressed as

$$b_i = g_i \omega_i m_i$$

**Equation 2-26.**

where $\omega_i$ is the undamped vibration frequency (equal to $\sqrt{k_i / m_i}$), expressed in radians per unit time. The dimensionless input coefficient, $g_i$, is equal to twice the critical damping ratio of the mode. It is evaluated by linear interpolation of a user-specified function of frequency, *g(f)*.

The physical meaning of modal damping is somewhat clouded. Its effect is viscous damping, but it is proportional to the stiffness matrix, and varies with modal frequency. The damping effects are distributed to the structure depending on the energy distribution in each mode shape.

Another aspect of modal damping to remember is that the damping coefficient, *b*, is constant for each mode. If the mode is forcibly excited at a different frequency the initial damping factor still applies. In

other words, the damping at any particular frequency is a function of several modal damping factors. This may cause unexpected results for frequencies where a mode is not dominant.

The user specifies pairs of values *f,g* on a TABDMP1 Bulk Data entry, which is selected by the SDAMPING Case Control command. There are also provisions for expressing damping as a fraction of critical damping ($C/C_c$) or amplification quality factor ($Q$). If accurate test results are available, the user can specify different damping coefficients, obtained from modal tests, for the different frequency ranges. The user can ensure that the desired damping has been obtained for a set of modes whose frequencies are known from a previous run by providing the desired *f,g* pairs on the TABDMP1 entry.

The complete damping matrix for modal transient analysis, $[B]$, is

$$[B] = [b_i] + [\phi]^T [B^v][\phi]$$

**Equation 2-27.**

where $[b_i]$ is a diagonal matrix whose elements are given by **Eq. 2-26**, $\varphi$ is the matrix of eigenvectors, and $[B^v]$ is the matrix of nonmodal damping terms. Note that the damping effects which enter $[B^v]$ may well be duplicated by the effects included in $[b_i]$ so that, in general, the user should be careful when using both forms simultaneously.

> Note:
>
> In frequency response and complex eigenvalue analysis, the complete damping matrix is similar, except that structural damping is treated as a complex stiffness matrix.

Also, the matrix $[\varphi]^T[B^v][\varphi]$ in **Eq. 2-27**, or the equivalent complex stiffness matrix, is generally coupled so that the efficient uncoupled methods of analysis cannot be used when $[B^v]$ is present.

It is the accepted practice in many industries to express viscous damping as a fraction of critical damping. Critical damping is defined as the value at which the homogenous solution of **Eq. 2-25** transitions from a damped sinusoid to a nonoscillating, decaying exponential. A solution for the modal equation shows the value of critical damping, $C_c$, to be

$$C_c = 2\sqrt{k_i m_i}$$

**Equation 2-28.**

The fraction of critical damping, $\zeta_i$, is calculated from the equation

$$\zeta_i \equiv \frac{b_i}{C_c} = \frac{b_i}{2\sqrt{k_i m_i}}$$

**Equation 2-29.**

Another form of damping specification is the amplification quality factor, $Q_i$, with the definition

$$Q_i = \frac{1}{2\zeta_i} = \frac{1}{g_i}$$

**Equation 2-30.**

All three forms of damping specification are available for modal damping.

An illustration of the comparison between modal damping and an equivalent function of structural damping is shown in **Figure 2-2**, showing the different options for modal damping tables. With unit modal masses, a constant modal damping table actually corresponds to a viscous damping that increases with modal frequency. This method falls between the two extremes (constant viscous damping and equivalent structural damping) and serves as a general purpose compromise.

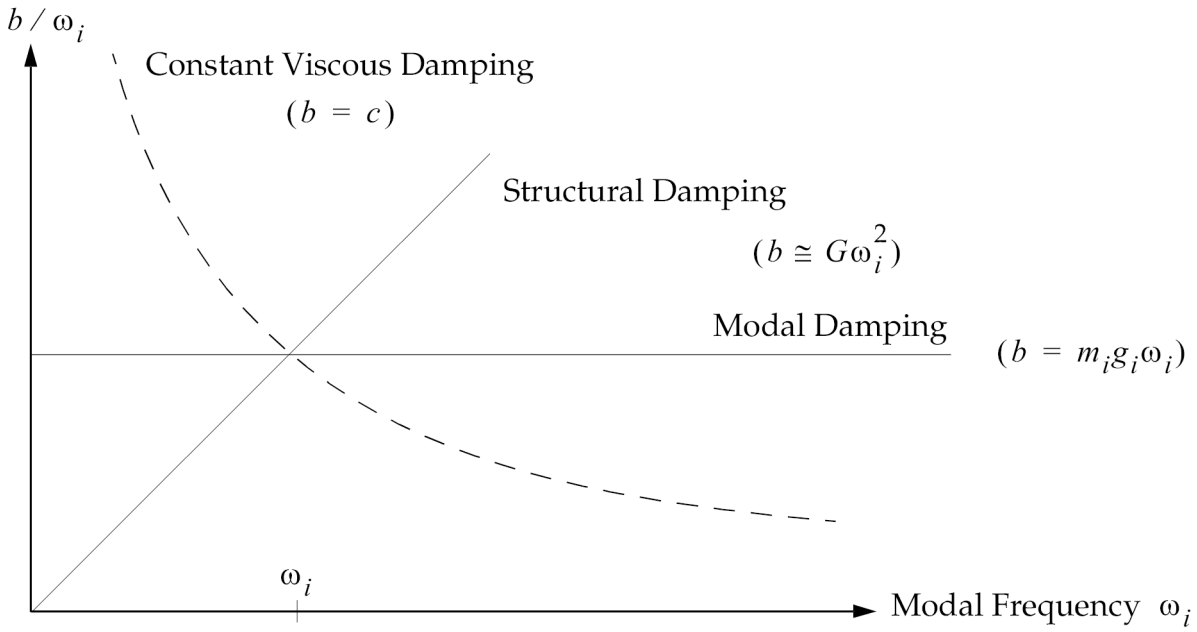



**Figure 2-2. Damping Coefficients Versus Modal Frequency**

### Nonlinear Damping

This discussion explains the limitations in damping caused by the nonlinear transient solution algorithms (SOL 129). Frequency response and complex eigenvalue solutions are not available in nonlinear analysis. For specialized nonlinear dynamics topics, see **"Nonlinear Transient Response Analysis"**. For basic information on the Simcenter Nastran nonlinear solutions, see the *Simcenter Nastran Handbook for Nonlinear Analysis* or the *Simcenter Nastran User's Guide*.

Finite elements may be either linear or nonlinear in the nonlinear formulations, and the damping effects will be treated differently for each type. The basic rules for nonlinear transient analysis are as follows:

1. Plastic yield in the nonlinear materials automatically absorbs energy when the structure follows a loading and unloading cycle. This is an actual hysteresis effect that produces an accurate form of damping. However, note that strain rate effects are not calculated directly. Strain rate effects must be modeled with structural damping parameters, which are converted internally to viscous damping.

2. Viscous damping elements are always linear and will participate as constant matrix terms.

3. Structural damping, defined with parameters and material bulk data inputs (the GE field on the MATi entries), is recognized for both linear and nonlinear elements. The damping matrix terms are calculated for the current material stiffness moduli and geometry. Note, however, that the tangent matrices are only updated periodically. The actual damping on nonlinear elements is unpredictable and can change answers for different runs on the same problem—depending on the convergence rate and iteration strategy. It is recommended that the matrix update strategy forces an update on the tangent matrix at every time step.

4. A modal formulation (and therefore modal damping) is not available in a nonlinear solution.

5. Superelements may be used to reduce the size of a nonlinear problem by separating the linear elements into an upstream component. Component modal synthesis may then be used to maintain accuracy. Structural damping is allowed on superelements.

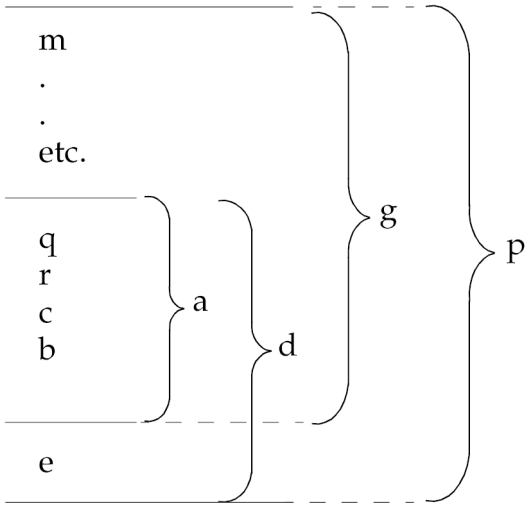
## 2.4 DMIGs, Extra Points, and Transfer Functions

In dynamics modeling, we frequently need to include special nonfinite element effects such as mechanical devices, servomechanisms, smart structures, and matrices from external structures. These effects can usually be included as extra terms in the system matrices along with extra degrees-of-freedom in the generalized displacement vectors.

The major difference between extra points (EPOINT data) and normal scalar points (SPOINT data) is that the extra points are added to the system after the finite element matrix assembly and real eigenvalue solution. Therefore, structural elements, constraints, and static loads may not be connected to EPOINTs. Also, as with scalar points, they are processed only in the residual superelement in SE formulations.

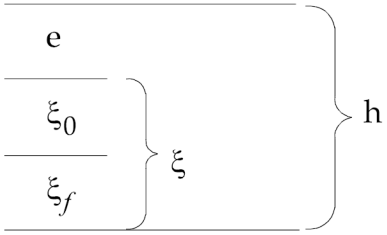
## The EPOINT Set

Using standard set notation (see **"Set Notations"** ), the extra point set merges with the structural degrees-of-freedom according to the following diagram:




In direct solutions, the system matrices are formed by merging structural points and e-points into the p-set. The actual solution matrices are assembled and added together at the reduced, d-set level.

The set logic for modal formulations is as follows:




In modal solutions the system matrices are transformed to the modal coordinates and merged with the extra point degrees-of-freedom. The modal coordinates include free body modes, $\xi_4$, and flexible modes, $\xi_f$. Note that the coupling between the structural motions and the extra points is only approximated by the truncated modal solution.

## Modal Transformations

When extra points are included in the modal formulation, a transformation is needed between the physical displacements and the modal coordinates. The displacements, $u_p$, are defined by the transformation

$$\{u_p\} = \begin{Bmatrix} u_g \\ u_e \end{Bmatrix} = [\phi_{ph}] \begin{Bmatrix} \xi \\ u_e \end{Bmatrix}$$

**Equation 2-31.**

where:

$$[\varphi_{ph}] = \begin{bmatrix} \phi_g & 0 \\ 0 & I \end{bmatrix}$$

and $\varphi_g$ are the real eigenvectors. The loads and direct input matrices are transformed consistently with the displacements. For the applied loads, *P*,

$$\{P_h\} = [\phi_{ph}]^T \{P_p\}$$

**Equation 2-32.**

Direct input matrices, K2PP, are transformed similarly in the form

$$[K^2_{hh}] = [\phi_{ph}]^T [K^2_{pp}][\phi_{ph}]$$

**Equation 2-33.**

Mass and damping matrices are also transformed similarly.

Note that the extra points remain in the modal formulation. These are useful for modeling transfer functions, initial conditions, and simple nonlinear functions.

Because these matrix terms couple the modal coordinates, this option is more expensive than solving the simple uncoupled modal equations of motion. The coupled solution requires the same type of matrix operations as the direct method.

## Direct Matrix Inputs

Since direct matrix input (DMIG) data is covered in the *Simcenter Nastran User's Guide* and the *Simcenter Nastran Basic Dynamic Analysis User's Guide*, this section will be brief.

Note:

DMIG matrices are selected by the K2PP =, B2PP =, and the M2PP = Case Control commands.

For dynamics modeling, the direct input matrices are defined for the p-set of degrees-of-freedom for mass, damping, and stiffness matrices. The actual values are defined by the user and may represent any type of linear solution, including fluids, electrical circuits, and external structures. However, some practical recommendations are as follows:

1. Use the double field bulk data format (DMIG*) if more than three significant digits of the input values are desired. Note that these solutions are sensitive to the precision of the matrix terms. Use DMAP modules, INPUTi, for full double-precision input.

2. Use the automatic selection of single- or double-precision data for the system. The program sets the size depending on the word size of the computer.

3. Use the symmetric matrix option if possible. The unsymmetric flag will force all operations into the unsymmetric mode, costing time and storage space.

An example problem that uses DMIG data for generating friction forces is given in **"Complex Eigensolutions"** .

## Transfer Functions

The Simcenter Nastran transfer functions (TF inputs) are actually a convenient method for generating special unsymmetric matrix input. When used with extra points they can define second order operators with one output and multiple inputs. The basic equation defining individual TF input is

$$(B_0 + B_1 p + B_2 p^2) u_d + \sum_i (A_{oi} + A_{1i} p + A_{2i} p^2) u_i = 0$$

**Equation 2-34.**

Here $u_d$ is the dependent degree-of-freedom, $u_i$ are the selected input degrees-of-freedom, and the coefficients *A* and *B* are user-specified.

Internally, these coefficients are simply added to a single row in the matrix equation:

$$[Mp^2 + Bp + K]\{u\} = \{P\}$$

**Equation 2-35.**

The terms are added to the matrices in the following positions:

1. All terms in a single function are added to the row of the matrices corresponding to $u_d$.

2. $B_0$ is added to the diagonal term of [$K$]. $B_1$ is added to the [$B$] matrix, and $B_2$ is added to the [$M$] matrix on the diagonals, respectively.

3. For each independent point , $u_i$, $A_{0i}$, $A_{1i}$ , and $A_{2i}$ are added to the column corresponding to $u_i$, and the row corresponding to $u_d$, of the stiffness, damping, and mass matrices, respectively.

### Limitations

If the TF terms are the only occupants of the $u_d$ row of the matrix equation, the dynamic solution will include the basic equation. However, no internal checks are made for additional terms in the TF row in the matrices or loads. To avoid conflicts, the following rules apply:

1. $u_d$ should be the dependent variable for only one TF.

2. $u_d$ should not be a component of a structural grid point unless the transfer function defines a load (see the discussion of transfer functions below).

3. External excitation signals, dynamic loads (DLOAD), and nonlinear functions (NOLINi) may be applied to the $u_d$ point.

4. Unlike the multipoint constraints, the TF equations do not conserve energy. The matrix terms are not symmetric, and no reciprocity forces are generated.

## Pickups and Transducers

Connections to the structure may be needed for both inputs and outputs from the transfer functions. To specify inputs, the structural displacements, velocities, or accelerations may be referenced on the $A_i$ and $u_i$ input fields. The $u_d$ may be an extra point defining the pickup voltage. This signal may then be processed by a series of transfer functions (including feedback loops) until a servo generates a load on the structure.

It is recommended that the transfer function results applied to the structure be represented by a force or moment. An enforced displacement, velocity, or acceleration is difficult to model within the limitations described above.

To generate a force, $F_j$ on a structural point, $u_j$, with a transfer function,

$$F_j = (A_0 + A_1 p + A_2 p^2) u_e$$

define a TF function of the form

$$(0)u_j - (A_0 + A_1 p + A_2 p^2)u_e = 0$$

**Equation 2-36.**

The $B_j$ coefficients must be zero since the structural point $u_j$ is already connected to the finite elements. This transfer function is equivalent to adding a positive force on the right-hand side of the matrix equation.

If the force or moment is acting on an actuator with a feedback loop, include the feedback as another transfer function.

## Higher-Order Transfer Functions

In many cases the control system includes a polynomial transfer function of order greater than two and cannot be modeled directly. Since the TF inputs are limited to second-order polynomials, the larger polynomials must be subdivided into several TF inputs with intermediate extra points. For instance, if a point, $u_a$, is defined by the polynomial

$$(A + Bp + Cp^2 + Dp^3 + Ep^4)u_b = G(p)u_a$$

**Equation 2-37.**

this may be subdivided into two equations suitable for the TF format:

$$(A + Bp + Cp^2)u_b + p^2 u_{e1} - G(p)u_a = 0$$

**Equation 2-38.**

and

$$u_{e1} - (Dp + Ep^2)u_b = 0$$

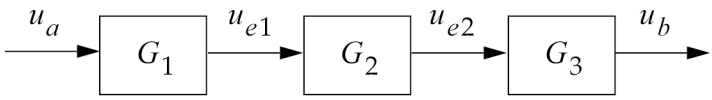
**Equation 2-39.**

Here, $u_{e1}$ is an intermediate extra point which is coupled to the $u_b$ degree-of-freedom in both equations.

Note that this method may produce a singular mass matrix that causes problems for certain complex eigenvalue options. This would occur in the example above if $E = 0$. A solution is discussed in **"Complex Eigensolutions"**.

### Alternate Method

If the polynomial is available in factored form, i.e., $G_1(p) \cdot G_2(p) \cdot G_3(p)...$, a series of TF transfer functions can be used as illustrated in the following sketch.




If $u_b$ is the dependent degree-of-freedom, the TF equations will be in the following form:

$$\begin{aligned} u_{e1} - G_1 u_a &= 0 \\ u_{e2} - G_2 u_{e1} &= 0 \\ u_b - G_3 u_{e2} &= 0 \end{aligned}$$

For polynomials in the denominator of a transfer function, use a feedback loop or coupled equations as shown in the previous example. See the example below for a realistic application.

### Example

As an example of the general approach, consider the control system shown in **Figure 2-3**, which might represent a simplified model of an optical device. It contains many of the components found in control systems including attitude and rate sensors, signal conditioners, and a nonlinear mechanical actuator with local feedback. The structure is represented in **Figure 2-3** by the displacement components, which are sensed by the control system, or to which loads are applied.

The transfer functions (TFs) of the control system are listed in **Table 2-4**. The first five transfer functions give the outputs of the devices labeled 1 to 5 in **Figure 2-3** in terms of their inputs. TF 6 defines the force on the structure, at point $u_{11}$, which is a function of the input to the mechanical actuator.

In the analysis, an extra point is assigned to each of the new variables, $u_4$, ..., $u_{10}$. The coefficients of the transfer functions, expressed by TFs 1 to 6, are listed in **Table 2-5**. Note the coefficient of 0.0 for the structural degree-of-freedom, $u_{11}$, on TF 6. The existing structure provides the stiffness and mass for this displacement.

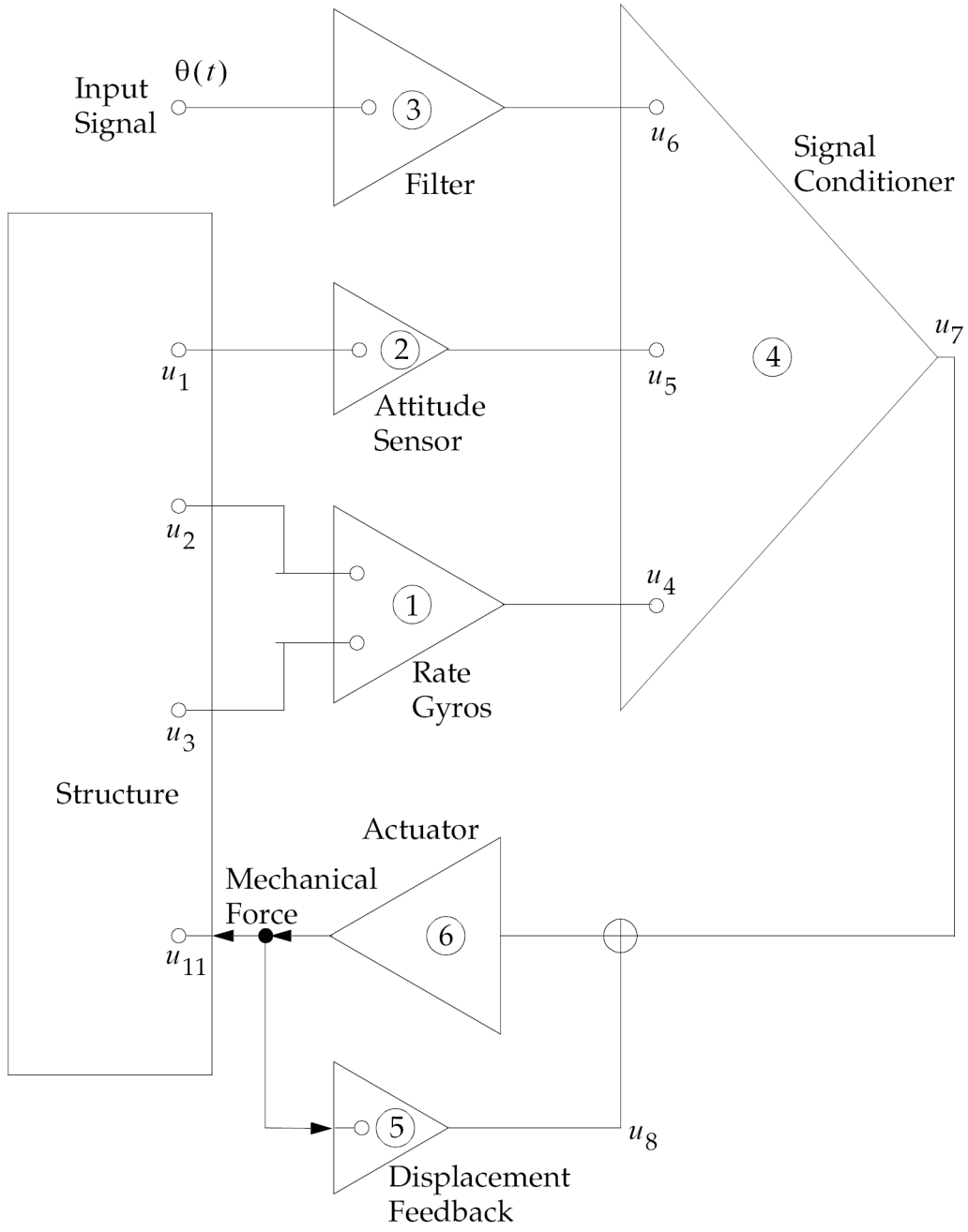



**Figure 2-3. Example Control System**

**Table 2-4. Equations of Example Control System**

| Equation | TF Number |
|---|---|
| $u_4 = \frac{1}{1+\tau_1 p}(Apu_2 + Bpu_3)$ | 1○ |
| $u_5 = \frac{1}{1+\tau_2 p}Ru_1$ | 2○ |
| $u_6 = \frac{1}{1+\tau_3 p}u_{10}$ | 3○ |
| $u_7 = \frac{A_0 + A_1 p}{B_0 + B_1 p + B_2 p^2}(Cu_4 + Du_5 + Eu_6)$ | 4○ |
| $u_8 = \frac{Gu_{11}}{1+\tau_4 p}$ | 5○ |
| $(0)u_{11} = -k(u_7 - u_8)$ | 6○ |

**Table 2-5. Transfer Function Table**

| Transfer Function No. | $u_d$ | $b_0$ | $b_1$ | $b_2$ | $u_i$ | $a_0$ | $a_1$ | $a_2$ |
|---|---|---|---|---|---|---|---|---|
| 1 | $u_4$ | 1 | $\tau_1$ | | $u_2u_3$ | | –A–B | |
| 2 | $u_5$ | 1 | $\tau_2$ | | $u_1$ | –R | | |
| 3 | $u_6$ | 1 | $\tau_3$ | | $u_{10}$ | –1 | | |
| 4 | $u_7$ | $B_0$ | $B_1$ | $B_2$ | $u_4$<br>$u_5$<br>$u_6$ | $-CA_0$<br>$-DA_0$<br>$-EA_0$ | $-CA_1$<br>$-DA_1$<br>$-EA_1$ | |
| 5 | $u_8$ | 1 | $\tau_4$ | | $u_{11}$ | –G | | |
| 6 | $u_{11}$ | 0.0 | | | $u_7u_8$ | –k<br>+k | | |

Input Data File

The actual structure is a beam-like optical pointing device pivoting on the center. The data file for a complex eigenvalue analysis is shown in **Listing 2-1**. The attitude sensor ($u_8 = \theta_z$), the first rate gyro ($u_2$

= $u_y$), and a lumped mass are located on the right end (GRID 1). The second rate gyro is attached to the next point (GRID 3). The actuator is connected to the left end (GRID 11) in the *y* direction. The units for the mechanical model are millimeters, kilograms, seconds, milliNewtons, and radians. The units for the control system variables (EPOINTs 4,5,6,7,8,10) are volts.

```
ID ADUG,TFANDE
DIAG 8
SOL 107
TIME 5
CEND
TITLE = CONTROL SYSTEM EXAMPLE FOR ADUG
SUBTITLE = TRANSFER FUNCTIONS AND EXTRA POINTS
LABEL = COMPLEX MODES
SPC = 10
TFL = 6
CMETHOD = 200
SDISP=ALL
BEGIN BULK
$
EIGC,200,INV,MAX,,,,,,+EIG1
+EIG1,0.0,-1.0,0.0,1200.,20.0,10
$
$ STRUCTURE IS A BOX BEAM PIVOTING AT THE CENTER
GRDSET,,,,,,,1345
GRID,1,,500.0
GRID,3,,250.0
GRID,14,,0.0
GRID,15,,-250.0
GRID,11,,-500.0
CBEAM,1,1,1,3,,,1.0
CBEAM,2,1,3,14,,,1.0
CBEAM,3,1,14,15,,,1.0
CBEAM,4,1,15,11,,,1.0
PBEAM,1,1,1000.,125+6,125+6,,250.+6
MAT1,1,69.0+6,,0.3,5.0-6
$ OPTICAL DEVICE ON THE END
CONM2,6,1,,10.0,,,,,+CNM2
+CNM2,8.0+5,,4.0+5,,,4.0+5
$                    PIVOT ON THE CENTER
SPC,10,14,12
$                     EXTRA POINTS ARE VOLTAGES
EPOINT,4,5,6,7,8,10
$            TRANSFER FUNCTIONS IN ORDER
$                   RATE GYROS PICK UP VELOCITIES
TF,6,4,,1.0,0.015,,,,+TF101
+TF101,1,2,,-0.25,,,,,+TF102
+TF102,3,2,,-0.5
$                    ATTITUDE SENSOR MEASURES ROTATION RZ
TF,6,5,,1.0,0.02,,,,+TF201
+TF201,1,6,-1.2
$                    INPUT SIGNAL, E10, IS FILTERED
TF,6,6,,1.0,0.001,,,,+TF301
+TF301,10,,-1.0
$                    SIGNAL CONDITIONER COMBINES THE VOLTAGES
TF,6,7,,39.48,8.885-2,1.0-4,,,+TF401
+TF401,4,,-20.0,-0.4,,,,,+TF402
+TF402,5,,-200.0,-4.0,,,,,+TF403
```

```
+TF403,6,,100.,-20.0
$                    DISPLACEMENT FEEDBACK TO THE ACTUATOR
TF,6,8,,1.0,0.005,,,,+TF501
+TF501,11,2,-1.0
$                 ACTUATOR FORCES ARE ADDED TO ROW OF STRUCTURAL MATRICES
TF,6,11,2,0.0,,,,,+TF601
+TF601,7,,-1.0+5,,,,,,+TF602
+TF602,8,,1.0+5
$                    TEMPORARILY GROUND POINT 10 WITH A DIAGONAL TERM
TF,6,10,,1.0
ENDDATA
```

**Listing 2-1. Data Listing for Optical Device Control System**

## 2.5 Nonlinear Load Functions (NOLINi)

The nonlinear load functions (NOLINi) are available in all of the transient solutions for the purpose of generating direct forces from simple displacement and velocity functions. This is a mature capability in Simcenter Nastran that preceded the development of the material and geometric nonlinear capabilities. It was intended to provide a direct method for modeling certain mechanisms and special effects at the local level where only a few degrees-of-freedom are coupled. It is not intended for general nonlinear analysis since the logic cannot detect geometric changes and has no means of using element stresses or forces.

The main applications for the NOLINi functions are for local devices such as contact problems, joints, and nonlinear dampers. For each nonlinear force component, the dependencies with displacement and velocity degrees-of-freedom are explicitly defined on one or more bulk data entries.

The limitations on the use of these functions are caused primarily by their simplicity. These limitations are:

1. They have no memory or other path dependencies. Specifically, they provide no direct means for including data from previous states.

2. Each input function only applies loads to a single degree-of-freedom, which requires a great deal of effort to describe complex models.

3. The system treats these functions as simple forces rather than finite elements. Therefore, the lack of a tangent matrix results in potential stability problems.

4. In the linear solutions, all degrees-of-freedom in the NOLIN functions must be in the dynamic solution set.

The input data is simply a case control request: NONLINEAR, with one or more NOLINi Bulk Data entries to define the set. The four basic options to define a scalar nonlinear force, $N_i$, in terms of the degrees-of-freedom, $u_j$, are summarized in the table below.

| Option | Function | Comments |
|---|---|---|
| NOLIN1 | $N_i = SF(u_j)$ | $F(u_j)$ is a TABLEDi input |
| NOLIN2 | $N_i = Su_ju_k$ | Product of two variables |
| NOLIN3 | $N_i = S(u_j)^A, u_j > 0$ | $A$ is an input exponent |
| NOLIN4 | $N_i = -S(u_j)^A, u_j < 0$ | Same except for negative $u$ |

The variables, $u$, may be displacement or velocity components of grid, scalar, or extra points in the solution set. In the DIRECT TRANSIENT solutions, the connected degrees-of-freedom, $u_i$ and $u_j$, etc., must remain in the solution set, $u_d$. In the modal transient solutions, only extra points are available for use by the NOLINi entries. In nonlinear transient solutions all degrees-of-freedom are available, but unfortunately, extra points are not supported.

Note that these functions mimic the basic nonlinear function generators used in passive analog computers. They may be added together and combined with other functions such as MPCs and scalar elements to handle a variety of problems.

## Theory for the Standard Transient Solutions

The basic linear transient solution integrates the matrix equation

$$[M]\{a\} + [B]\{v\} + [K]\{u\} = \{P\} + \{N\}$$

**Equation 2-40.**

where $\{N\}$ are the nonlinear forces which are dependent on variable displacements, $\{u\}$, and velocities, $\{v\}$, of the unknowns. Note that if these functions were used to replace a stiffness term, the proper definition of $N$ would be $N=-Ku$.

There are several options in Simcenter Nastran for transient integration. For the basic purposes of this introduction, we will discuss the simplest form, the three-point method which performs a step-by-step calculation. At time step $t_n$, the solution is $\{u_n\}$ and the step size is $h$. Dropping the brackets, the averaged values of displacement, $u$, velocity, $v$, acceleration, $a$, and load, $P$, in terms of the three discrete steps are as follows:

$$u = \beta u_{n+1} + (1 - 2\beta)u_n + \beta u_{n-1}$$

**Equation 2-41.**

$$v = \{u_{n+1} - u_{n-1}\} / 2h$$

**Equation 2-42.**

$$a = (u_{n+1} - 2u_n + u_{n+1}) / h^2$$

**Equation 2-43.**

$$P = \beta P_{n+1} + (1 - 2\beta)P_n + \beta P_n$$

**Equation 2-44.**

In the solution, the vectors at step *n+1* are obtained by substituting

$$\begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = -\begin{bmatrix} M & -M \\ -M & M \end{bmatrix} \begin{Bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \end{Bmatrix}$$

(for more information, see **Mass Data Input Options** through

$$[m] = \begin{bmatrix} m_1 + m_2 & -m_2 \\ -m_2 & m_2 \end{bmatrix}$$

(for more information, see **Coupled Mass Matrix Terms**) in **Eq. 2-40**. This method is actually a variation of the Newmark-Beta method and is guaranteed to be stable if $\beta > 0.25$. For consistency and stability, it would be desirable to have the *N* vector also in this form. Ideally, the nonlinear loads would be consistent with the linear displacements and loads, i.e.,

$$N = \beta N_{n+1} + (1 - 2\beta)N_n + \beta N_{n-1}$$

**Equation 2-45.**

However, $N_{n+1}=N(u_{n+1},v_{n+1})$ is not available and can only be approximated by extrapolating the equation:

$$N_{n+1} \sim 2N_n - N_{n-1}$$

**Equation 2-46.**

Substituting this approximation into **Eq. 2-45** we obtain the term used in Simcenter Nastran:

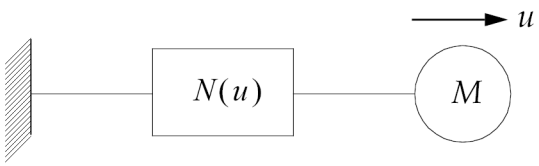
$$N \sim N_n$$

**Equation 2-47.**

In other words, calculating the nonlinear function at the center step is a valid approximation to the value averaged over three steps. Unfortunately, when *u* is rapidly changing, *N* will lag behind and create unstable oscillations. The results of an unstable integration usually grow with a period of two or three time steps per cycle. An alternate option for averaging nonlinear loads is available with DIAG 10, although it is *not* recommended.

## Example of a Nonlinear Spring

An example of an unstable system can be illustrated with a single variable example. Attach a nonlinear spring function to a mass as shown in the sketch below.




For small motions, if we define the function as an equivalent spring with $N=ku$, the integration equation for $u_{n+1}$ is

$$M(u_{n+1} - 2u_n + u_{n-1})/h^2 = N = -ku_n$$

**Equation 2-48.**

For stability analysis, we will assume a uniform growth rate, *r*, where $u_n=ru_{n-1}$ and $u_{n+1}=ru_n$. Substituting into **Eq. 2-48** we obtain a quadratic equation for *r*:

$$r^2 + 2\left(\frac{kh^2}{2M} - 1\right)r + 1 = 0$$

**Equation 2-49.**

If the magnitude of *r* is greater than 1, the solution will potentially diverge (*r* may be complex). It is easy to show that if $kh^2 > 5M$ the system will have a real root $r < -1$, which is an unstable oscillation that changes sign every step.

In other words, the stability limit for the nonlinear stiffness for this case is

$$-\frac{\partial N}{\partial u} = k \leq \frac{4M}{h^2}$$

**Equation 2-50.**

The cure for divergence is to either reduce the time step, *h*, add linear stiffness to replace some of the nonlinear portion, or add enough mass, *M*, in parallel to the nonlinear function to satisfy the criteria for every possible value of *k*. Scalar mass elements CMASSi may be added to couple two scalar points, in parallel with the nonlinear spring, without affecting the total mass of the structure.

## Velocity-Dependent Nonlinear Forces

When a NOLINi function references a velocity of a grid, scalar, or extra point component as an input, the nonlinear force must be calculated from the existing displacements. However, for consistency with the linear damping terms in

$$[M]\{a\} + [B]\{v\} + [K]\{u\} = \{P\} + \{N\}$$

(for more information, see **Theory for the Standard Transient Solutions**), the nonlinear force should be:

$$N_v = -B(u,v) \cdot (u_{n+1} - u_{n-1})/2h$$

**Equation 2-51.**

However, the current displacements $u_{n+1}$ have not been calculated at this stage, so we again use the assumption of

$$N_{n+1} \sim 2N_n - N_{n-1}$$

(for more information, see **Theory for the Standard Transient Solutions**) and obtain the approximation:

$$N_v \sim -B(u,v)(u_n - u_{n-1})/h \sim N[(u_n - u_{n-1})/h]$$

**Equation 2-52.**

The stability of velocity-dependent NOLINi systems is very marginal for most applications. Replacing the nonlinear spring (*k*) in the example problem above with a nonlinear damper, *b*, we obtain the following system equation:

$$M(u_{n+1} - 2u_n + u_{n-1})/h^2 \;=\; N \;=\; -b \cdot (u_n - u_{n-1})/h$$

**Equation 2-53.**

For stability analysis we may substitute $ru_n$ and $r^2u_n$ as in the stiffness example above, to obtain a quadratic equation:

$$r^2 + \left(\frac{bh}{M} - 2\right)r + \left(1 - \frac{bh}{M}\right) \;=\; 0$$

**Equation 2-54.**

Unstable roots ($|r|>1$) will occur when

$$b > 2M/h$$

**Equation 2-55.**

Again, the problem will show up as a spurious diverging oscillation. Divergence may be cured by decreasing the time step size, *h*, by changing part of the nonlinear function, *b*, into a linear damper, or by adding mass in parallel with the nonlinear function. SOL 129 is better suited than SOL 109 for unstable or badly conditioned problems.

## Nonlinear Transient Solution Sequences

The general nonlinear transient solutions described in **Nonlinear Transient Response Analysis** will also process the NOLINi functions. They provide an advantage in that they will give the same stability as a linear solution, and will control the diverging solutions. These methods make an attempt to calculate, for time step *n+1*, the displacements and velocities by iterating on

$$[M]\{a\} + [B]\{v\} + [K]\{u\} \;=\; \{P\} + \{N\}$$

(for more information, see **Theory for the Standard Transient Solutions**) through

$$N \;=\; \beta N_{n+1} + (1 - 2\beta)N_n + \beta N_{n-1}$$

(for more information, see **Theory for the Standard Transient Solutions**) in a loop. This will provide more consistency with the linear terms. When the advanced methods such as line search and quasi-Newton options are activated, the adverse effects of the nonlinear forces are corrected effectively, but at the cost of additional solution steps.

However, the nonlinear methods may also attempt to change the tangent stiffness matrix when errors become too large in the solution search. Unfortunately, the NOLINi functions only provide forces on the right side and do not contribute their changes to the matrices. Not only will results not be improved, cost will be increased as well. Furthermore, since new self-adaptive method modifies the time step size based on the current values of the critical nonlinear matrix terms, the NOLINi terms will be ignored and the process may still diverge.

In summary, the nonlinear transient solutions may be adapted for severe cases of NOLINi- caused instabilities but will need careful changes to the default control parameters on the TSTEPNL inputs.

## Recommendations for Using NOLIN1 Input

The following suggestions may be useful if problems occur because of nonlinear forces:

1. Use the nonlinear solution (SOL 129) for potentially better error controls and stability. This approach may use more time and disk storage space and require additional effort to select the proper control parameters. We recommend starting with the following approximate TSTEPNL Bulk Data parameters:

   ```
   Method = "TSTEP"
   KSTEP > NDT
   MAXITER = 2
   MAXLS = 6
   MAXDIV = 10
   LSTOL = 0.1
   ```

2. Avoid functions that will overwhelm the linear elements attached to the same points. See the discussions on stability above. Most nonlinear functions can be defined as the sum of linear elements and a nonlinear correction using NOLINi inputs. If the slope of the nonlinear correction function is smaller in magnitude than the linear coefficients, the problem will be stable. Also, added mass in parallel with the function always helps.

3. Never set up nonlinear functions in series, i.e., using the output of one NOLINi as an input to a second NOLINi function. Each stage in effect causes a time delay of one or more time steps for the resulting forces. If the results are a corrective force, they will act much like a control system with a bad feedback loop. At a critical frequency the corrective forces will be applied with a delay that causes them to have the wrong sign.

4. The forces, *N*, must have the same sign and direction as an externally applied load on the structure. Note that this will have the opposite sign as the forces applied to the GRID points by the finite elements. (The *N* force for a spring element will have a negative value for a positive displacement.) This frequently causes some difficulty in debugging the input data.

5. Use MPC or TF data to simplify the NOLINi inputs. A single nonlinear spring in three dimensions connecting two points could require up to 6 · 6 = 36 nonlinear functions. Alternatively, one MPC equation could be used to define the strain as an added scalar point, and one nonlinear function would define the generalized force (force times length) on the scalar point. Because the MPC constraints are consistent, the force will automatically be distributed to all six degrees-of-freedom.

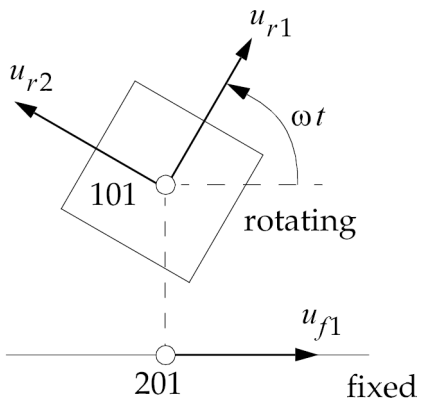
6. Plan ahead to retain the nonlinear points in the solution set. Remember that in SOLs 109 and 112 the NOLINi functions may not reference any dependent or eliminated degree-of-freedom. Note that in a modal analysis, this leaves only extra points.

7. Use the full nonlinear solution (SOL129) for small to medium-sized problems.

## Application Example – Coupling of Rotating Structures

An advanced use for the NOLINi functions is the connection between a rotating structure and a flexible supporting structure. The example below shows how to connect two grid points, with one rotating at a constant rate relative to the other. This method could be used to analyze sudden transients in rotating machinery. However, cost considerations would limit the analysis to a finite number of cycles.

As shown in the sketch, GRID 101 ($u_r$) is attached to the rotating structure. The rate of rotation is known, but the actual angle is a function of time. GRID 201 ($u_f$) represents the fixed bearing point coincident with point 101. It would typically be defined with an RBE3 as the average motion of the points on the bearing race.




The definition of the connection, as shown in the sketch is:

$$u_{r2} = -u_{f1}\sin(\omega t) + u_{f2}\cos(\omega t)$$

$$u_{r2} = -u_{f1}\sin(\omega t) + u_{f2}\cos(\omega t)$$

A simple MPC equation may be used to connect the axial coordinates $u_3$. The rotations may have similar definitions but are not shown. A similar equation defines the forces on the fixed point $N_f$ in terms of the rotating forces $Q_r$.

For the actual connection, we will use the Lagrange Multiplier technique by defining another GRID $Q_r$ which defines the constraint forces on point 101. The resulting matrix partition equation for the three points is

$$\begin{bmatrix} M_{ff} & 0 & 0 \\ 0 & m_{rr} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{u} \\ \ddot{u} \\ \ddot{Q}_r \end{Bmatrix} + \begin{bmatrix} K_{ff} & 0 & 0 \\ 0 & K_{rr} & I \\ 0 & I & 0 \end{bmatrix} \begin{Bmatrix} U_f \\ u_r \\ Q_r \end{Bmatrix} = \begin{Bmatrix} P_f + N_f \\ P_r \\ N_q \end{Bmatrix}$$

**Equation 2-56.**

where:

$$\{N_f\} = \begin{Bmatrix} P_{r1}\cos(\omega t) - P_{r2}\sin(\omega t) \\ P_{r1}\sin(\omega t) + P_{r2}\cos(\omega t) \end{Bmatrix}$$

$$\{N_Q\} = \begin{Bmatrix} u_{f1}\cos(\omega t) - u_{r2}\sin(\omega t) \\ -u_{f1}\sin(\omega t) + u_{r2}\cos(\omega t) \end{Bmatrix}$$

In order to generate the sine and cosine terms, we will add two scalar points and connect unit masses and springs $k=\omega^2$. Using initial conditions, these are the sine and cosine functions defined as scalar displacements. The following input data will solve the connection:

In Case Control

```
IC = 100 $ For sine and cosine
K2PP = RCOUP  $ For matrix terms
NONLINEAR=10 $ For NOLIN2
```

In Bulk Data

```
$ 50 radian/sec oscillators ( Note: Don't use PARAM,G)
SPOINT,2001,2002  $ sine and cosine
CMASS4,2001,1.0,2001
CMASS4,2002,1.0,2002
CELAS4,2003,2500.0,2001
CELAS4,2004,2500.0,2002
TIC,100,2001,,,50.0   $ sine
TIC,100,2002,,1.0       $ cosine
```

```
$ Lagrange Multiplier Grid and unit Matrix Terms.
GRID,3001, , , , , ,3456
DMIG,RCOUP,0,6,1,2
DMIG,RCOUP,101,1,,3001,1,1.0
DMIG,RCOUP,101,2,,3001,2,1.0
$ Note that other half is generated when using the SYM option

$ Nonlinear Functions = Products of Two Dof., Q_r
 and sine/cosine
NOLIN2,10,201,1,1.0,3001,1,2002
NOLIN2,10,201,1,-1.0,3001,2,2001
NOLIN2,10,201,2,1.0,3001,1,2001
NOLIN2,10,201,2,1.0,3001,2,2002
$
$ Products of U_1
 and sine/cosine
NOLIN2,10,3001,1,1.0,201,1,2002
NOLIN2,10,3001,1,1.0,201,2,2001
NOLIN2,10,3001,2,-1.0,201,1,2001
NOLIN2,10,3001,2,1.0,201,2,2002
$ end for one pair of points
```

Any type of structure may be connected to the axis points. However, for best stability place all of the mass at the axis on the active side, GRID 201, where the nonlinear force is applied. For inertia effects such as centripetal and Coriolis forces, the rotating body may require matrix inputs described in the section on moving coordinates. In order to keep these terms from affecting the fixed structure, superelement partitioning is recommended.

## 2.6 Debugging Dynamic Models

This section outlines a procedure for systematically checking and documenting a finite element model. Currently many models are generated by one organization and transmitted to other groups for subsequent analysis. Since these models have had various levels of analytical validation and documentation, checkout procedures are needed to ensure that the models will be consistent and mathematically well-conditioned. These procedures, however, are not substitutes for the independent verification phase of analysis. They are intended to remove modeling errors in the design process, rather than during the test updating phase, which occurs well after the hardware is built.

### Modeling Hints

There is no single checklist that will ensure a complete check of a comprehensive finite element model. Also, there is no substitute for actual test correlation with the model, nor is there a substitute for the analyst's engineering interpretation of the output and one's intuition. A results prediction, which is determining gross results before the analysis, can be used to good advantage. Simple load paths or natural frequencies of equivalent simple beam/mass systems, etc., can be used to remove redundancies and predict the results. In fact, this must be accomplished to some degree to size the model initially. This will also provide baseline data, and the effects of finer modeling will then be known.

Relative to preparation of the pilot model, the following are a few suggestions for eliminating or reducing modeling problems.

- Start construction of a simple pilot model in which the user should:
  - Use beams and plates instead of solids.
  - Use RBE2s and RBE3s where they will simplify.
  - Simplify modeling offsets and local modeling details.
  - Ignore minor discontinuities such as holes and fillets.

Further refinement after this initial modeling should yield acceptable changes in the results.

- Do not rely on bending capability of thin plates and long, thin axial members to render the model kinematically stable.
- Make an initial run with membrane-only properties and pinned ended bars, and check for irregularities.
- Avoid use of AUTOSPC in the final model.
- For shells, use the parameters K6ROT or SNORM.

The post-analysis assessment should include a check of the physical significance of the loads and of the load path. Offsets whose moments are not properly accounted for may overly weaken a very stiff load path. Also, large moments in relatively weak bending members or plates may indicate modeling problems.

Stress analysis should be performed at the detailed part level with the loads from the model. The use of element stresses directly from the output of the model requires detailed review in most cases. In fact, model properties may be intentionally different from the actual hardware to obtain correct load distributions, and to match test data or dynamic characteristics. Effective thicknesses or reduced bending properties may have been used to reflect panel cutouts or partial beam and fixity. In this event, the finite element model loads should be used with the actual drawing or as-built dimensions for detail stress analysis. This piece-part assessment ensures a check and balance of the finite element model and the stress distributions visualized and treated by the element selection. Also, the source of the components of stress are known, that is, whether the predominant stress component is due to bending or axial loads. Load transformation matrices are useful for isolating critical design conditions but are not necessarily a sufficient basis for computing the margin of safety.

One area in which an underestimation of load could occur is the local response of small masses during a dynamic analysis. These should be addressed in the detailed stress analysis with both the model predictions and an alternate loading such as a specified loading condition. For the model to give correct loads for the local response of a mass, one needs all of the following:

- Mass must be represented by enough points to characterize the energy of the critical local mode (a single-point mass may not be sufficient).
- Mass must be supported by proper elastic elements to represent the local mode (RBE2 or RBE3 may not be sufficient).
- Mass must be in the ASET.
- Model and all analysis (input spectra, etc.) must be carried beyond this local critical mode (as far as frequency is concerned).

## Testing Dynamic Models

Once the finite element model is completed and all documentation (such as model schematics (road maps) as well as material and geometric property calculations) are updated to the final model version, the following series of tests should be performed in order to validate the model. It is recommended that these tests be run on the model and subsystem models during the development stages as well.

### Geometry Plots

You can use a preprocessor graphics package should be used to obtain visual images of the finite element model from many views in such a way as to provide a clear representation of each element in at least one view and to verify overall geometry and placement of elements. **Figure 2-4** shows the Galileo spacecraft finite element model and **Figure 2-5** shows the Wide Field/Planetary Camera finite element model. A shrink option should be used if possible to make sure all elements are present (see **Figure 2-6**). This is particularly helpful when bars or beams are used to model stringers along the edges of plate elements. Discontinuities show up only when the shrink option is invoked.

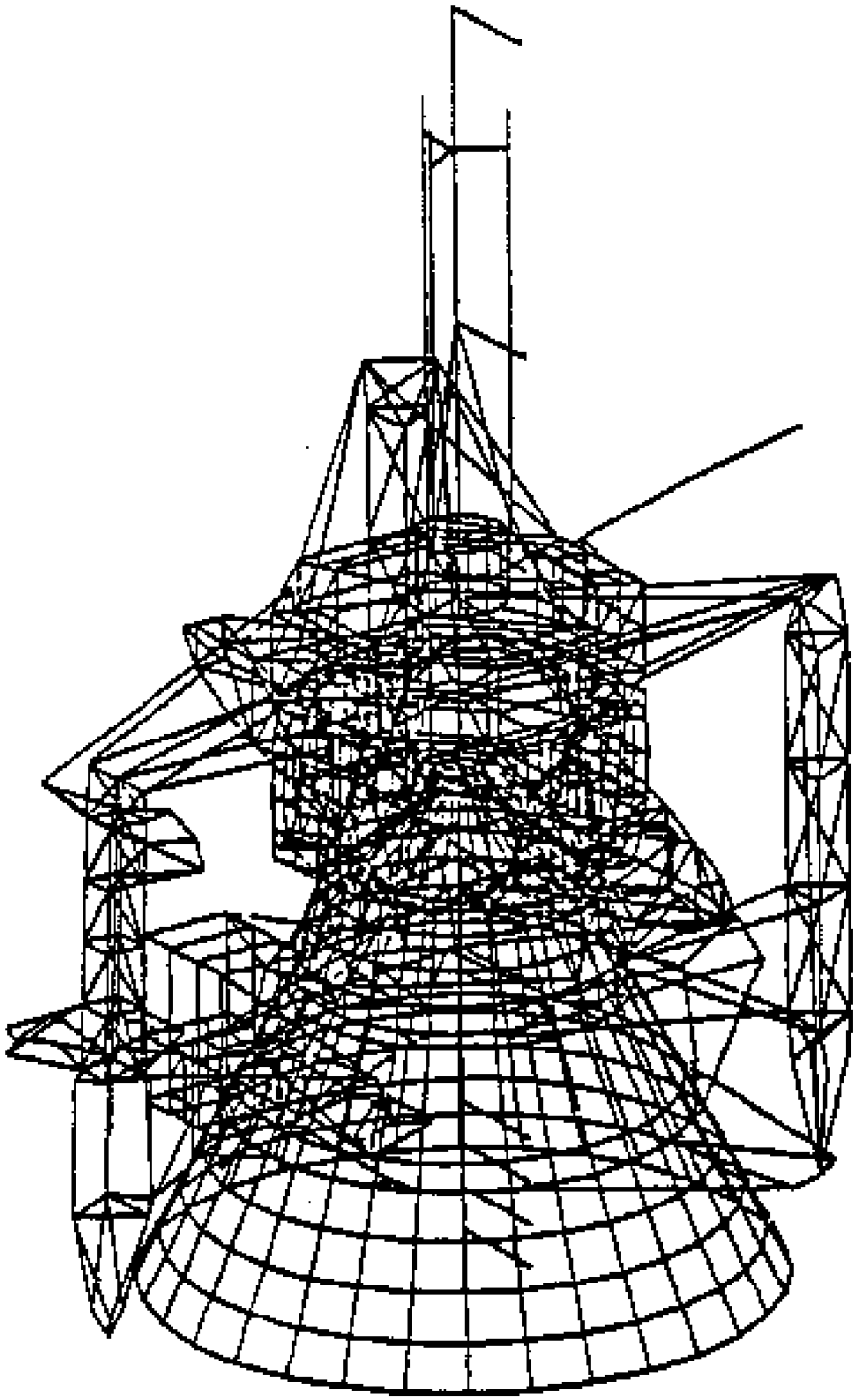


**Figure 2-4. Galileo Finite Element Model**

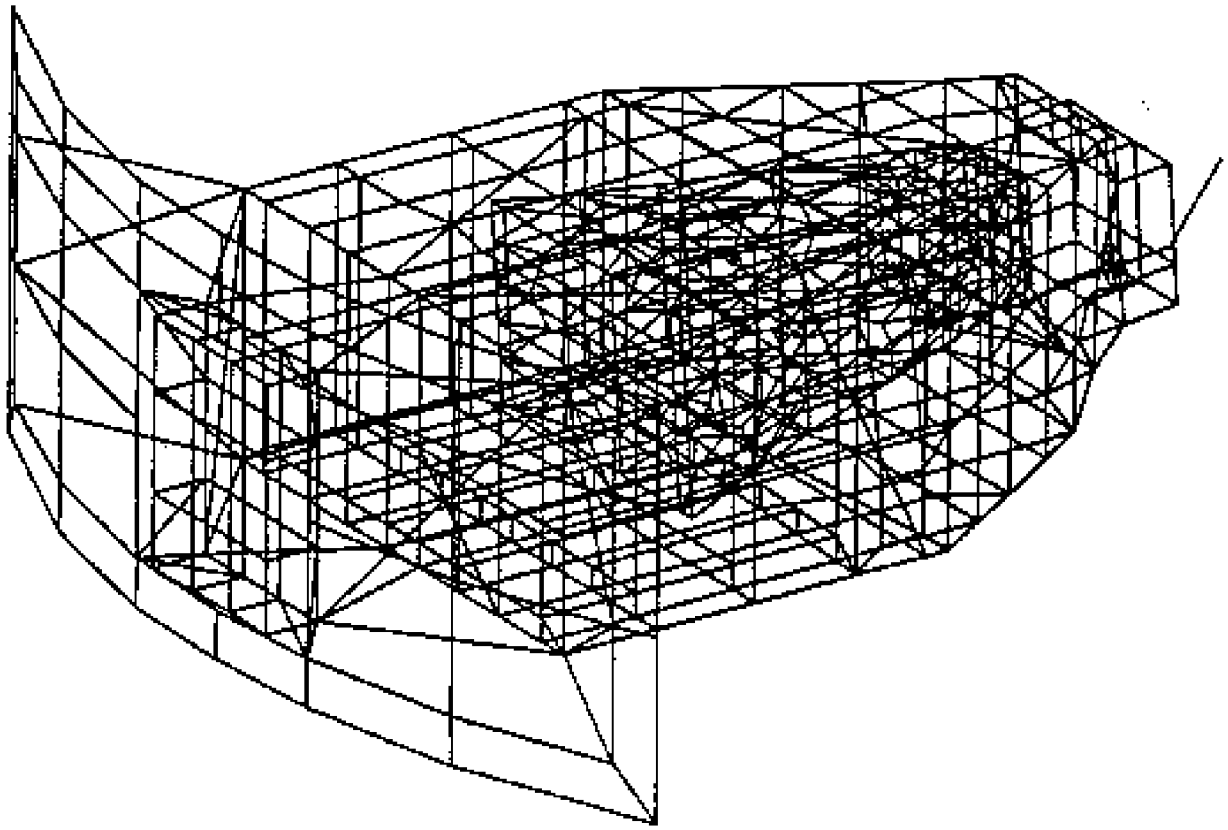
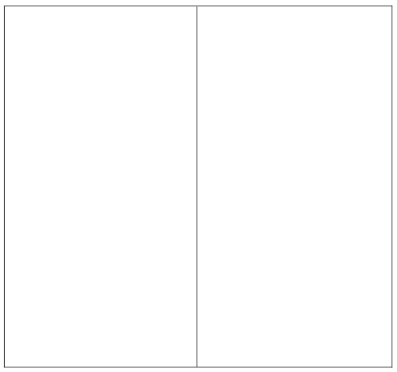
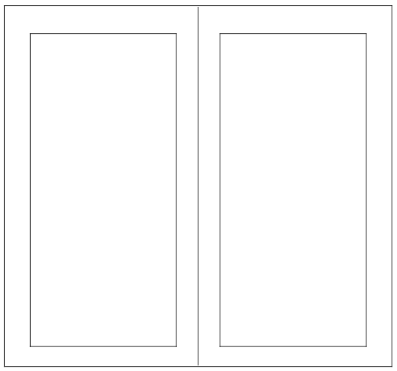


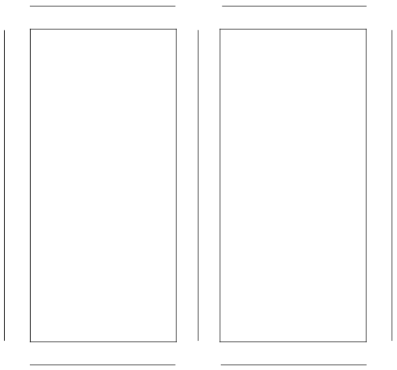
**Figure 2-5. Galileo Wide Field Planetary Camera Finite Element Model**



a) Shears and Bars, No Shrink



b) Bars Regular Size, Shears Shrunk



c) Bars and Shears Both Shrunk

**Figure 2-6. Shrink Option**

## Mass Distribution

PARAM,GRDPNT uses the Grid Point Weight Generator (GPWG), which gives the mass, the CG, the moments of inertia, and the principal moments of inertia and their direction cosines (see Table 2-8). Full use should be made of this diagnostic tool to correlate the model with existing hardware or mass properties calculations. The GPWG uses only the weight properties and geometry to calculate mass properties. The resultant mass properties are also called the rigid body mass properties. Note that PARAM,WTMASS does not affect the GPWG output—it is factored out of the printed results.

**Table 2-8. PARAM,GRDPNT Weight Inertia Matrix**

OUTPUT FROM GRID POINT WEIGHT GENERATOR

REFERENCE POINT = 1000 — GRID POINT OR ORIGIN OF BASIC COORDINATE SYSTEM

MO

| | | | | | |
|---|---|---|---|---|---|
| 1.562092E+00 | -1.689640E-17 | -1.040834E-17 | -4.440892E-16 | 5.894034E+01 | -1.359722E+01 |
| -1.767218E+17 | 1.562092E+00 | -1.416518E-16 | -5.894034E+01 | -4.973032E-16 | 3.635694E-01 |
| -1.040834E-17 | -1.419229E-16 | 1.562092E+00 | 1.359722E+01 | -3.635694E-01 | -4.440892E-16 |
| 4.440892E-16 | -5.894034E+01 | 1.359722E+01 | 3.191095E+03 | 3.648479E+00 | -2.795030E+00 |
| 5.894034E+01 | 2.972035E-16 | -3.635694E-01 | 3.648479E+00 | 3.283023E+03 | -5.082753E+02 |
| -1.359722E+01 | 3.635694E-01 | 2.220446E-16 | -2.795030E+00 | -5.082753E+02 | 4.852746E+02 |

S

| | | |
|---|---|---|
| 1.000000E+00 | 0. | 0. |
| 0. | 1.000000E+00 | 0. |
| 0. | 0. | 1.000000E+00 |

TRANSFORMATION MATRIX FROM THE BASIC SYSTEM TO THE PRINCIPAL MASS AXES

| DIRECTION MASS AXIS SYSTEM (S) | MASS | X-C.G. | Y-C.G. | Z-C.G. |
|---|---|---|---|---|
| X | 1.562092E+00 | 0.0 | 5.704492E+00 | 3.773167E+01 |
| Y | 1.562092E+00 | 2.327452E-01 | 0.0 | 3.773167E+01 |
| Z | 1.562092E+00 | 2.327452E-01 | 5.704492E+00 | 0.0 |

PRINCIPLE MASSES AND ASSOCIATED CENTERS OF GRAVITY RELATIVE TO THE REFERENCE POINT

I(S)

| | | |
|---|---|---|
| 8.488208E+02 | -6.813166E+00 | -1.092305E+01 |
| -6.813166E+00 | 1.059020E+03 | -4.770467E+00 |
| -1.092305E+01 | -4.770467E+00 | 3.668331E+02 |

MOMENTS OF INERTIA IN THE BASIC COORDINATE SYSTEM

I(Q)

| | | |
|---|---|---|
| 1.059279E+03 | | |
| | 8.488408E+02 | |
| | | 3.665549E+02 |

PRINCIPLE MOMENTS OF INERTIA

Q

| | | |
|---|---|---|
| -3.274377E-02 | -9.992094E-01 | 2.254901E-02 |
| -9.994364E-01 | 3.290163E-02 | 6.665341E-03 |
| -7.401970E-03 | -2.231605E-02 | -9.997235E-01 |

DIRECTION COSINES FROM BASIC COORDINATES TO PRINCIPAL INERTIA AXES

RIGID BODY MASS MATRIX RELATIVE TO THE REFERENCE POINT IN THE BASIC COORDINATE SYSTEM

## Static Loading

Static loading, such as a simple gravity load, can be helpful in checking out various properties of finite element models. Displacements, element forces, and support reactions (SPC forces) derived from 1G loading conditions provide a first check on mass, stiffness, and determinacy of supports. Weight and CG can be calculated from SPC forces, which should also be compared to any applied loads or weight. Load paths can also be assessed using the element forces. Epsilon, Max Ratio, and SPC forces (at grids other than legitimate boundary conditions) describe the overall health of the stiffness matrix. See Table 2-9 and Table 2-10 for example printout data.

## Table 2-9. Epsilon, Strain Energy and Max Ratio From Statics Run

*** USER INFORMATION MESSAGE 4158 – STATISTICS FOR SYMMETRIC DECOMPOSITION OF DATA BLOCK KOO FOLLOW
MAXIMUM RATIO OF MATRIX DIAGONAL TO FACTOR DIAGONAL = 3.9E+03 AT ROW NUMBER 594

*** USER INFORMATION MESSAGE 3035 FOR DATA BLOCK KLL

| LOAD SEQ. NO. | EPSILON | STRAIN ENERGY | EPSILONS LARGER THAN 0.001 ARE FLAGGED WITH ASTERISKS |
|---|---|---|---|
| 1 | -3.0539089E-14 | 8.2555099E+01 | |
| 2 | 1.0082989E-14 | 8.7870163E+01 | |
| 3 | 9.0509631E-16 | 2.3730776E+01 | |

*** USER INFORMATION MESSAGE 3035 FOR DATA BLOCK KOO

| LOAD SEQ. NO. | EPSILON | STRAIN ENERGY | EPSILONS LARGER THAN 0.001 ARE FLAGGED WITH ASTERISKS |
|---|---|---|---|
| 1 | 6.5035202E-15 | 2.2547545E-01 | |
| 2 | 7.3623740E-15 | 2.2061828E-01 | |
| 3 | 2.0116192E-15 | 1.3978869E-01 | |

## Table 2-10. SPC Forces From Statics Run

FORCES OF SINGLE-POINT CONSTRAINT

1.0G (+X) SUBCASE 1

| POINT ID. | TYPE | T1 | T2 | T3 | R1 | R2 | R3 |
|---|---|---|---|---|---|---|---|
| 201 | G | 0.0 | 0.0 | 0.0 | 0.0 | 2.797872E-13 | 0.0 |
| 2811 | G | 0.0 | 1.778028E-06 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2711 | G | 0.0 | -3.070124E-06 | 0.0 | 0.0 | 0.0 | 0.0 |
| 3197 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.731404E-09 |
| 3198 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.327725E-09 |
| 3297 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.494900E-09 |
| 3298 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -1.839192E-09 |
| 3397 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -4.187774E-10 |
| 3398 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -4.716660E-09 |
| 3497 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.664817E-09 |
| 3498 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 6.814997E-11 |
| 5602 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 6.451412E-05 |
| 5603 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -8.358259E-05 |
| 5604 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.047899E-05 |
| 5605 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -3.166949E-05 |
| 5606 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.126513E-05 |
| 5607 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.643013E-05 |
| 5608 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -4.914991E-05 |
| 5609 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 4.414012E-05 |
| 5610 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -3.978080E-05 |
| 5611 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.061714E-05 |
| 9999 | G | -5.640888E+03 | 5.932931E-06 | -1.097669E-07 | 1.582507E-04 | 4.669583E+05 | -4.979101E+02 |

MODEL WEIGHT RECOVERED AT THE BOUNDARY

MOMENT CONSISTENT WITH MODEL WEIGHT AND C.G. DATA

SPC FORCES AND MOMENTS IN THE ACCEPTABLE RANGE

Allowable values for these quantities are as follows:

| | |
|---|---|
| Epsilon* | ≤ $1.0 \times 10^{-6}$ (large model) |
| | ≤ $1.0 \times 10^{-9}$ (small model) |
| Max Ratio | ≤ $1.0 \times 10^{+5}$ |
| SPC Forces (at internal points) | ≤ $1.0 \times 10^{-5}$ (model weight) |
| SPC Moments (at internal points) | ≤ $1.0 \times 10^{minus;3}$ (model weight) × (unit length) |

> Note:
>
> Epsilon is machine dependent. The above data is for CDC 64-bitword. Other machines should give smaller numbers ($1.0 \times 10^{-8}$ and $1.0 \times 10^{-11}$). Mechanisms or symmetry conditions may require reevaluation of SPC force limits.

The 1G cases also provide a rough approximation of the frequency of the first mode. This approximation can be accomplished by using the displacement (*D*) at the CG in the equation $FN \sim \frac{1}{2\pi}\sqrt{G/D}$.

A 1G static load case can easily be obtained through the GRAV entry. This is preferred over the inertia relief type method.

Static forces and moments can be applied to generate displacements, element, and SPC forces. The magnitude and point of application of the forces should be representative of typical structural loading, thereby allowing the analyst a good feel for the size of displacements and forces as in the 1G cases. It is also helpful to use element strain energy and grid point force balance with these runs. For information on the details of requesting these capabilities, see Case Control Commands, “ESE” and “GPFORCE” in the *Simcenter Nastran Quick Reference Guide*.

## Free Body Equilibrium Check

Using Solution 101 and the set of CHECKA alters in the SSSALTER Library, an equilibrium check can be run to further validate the stiffness matrix. This check calculates the strain energy resulting from unit translations and rotations.

All SUPORTs and SPCs should be removed. Include the following PARAMs in the Bulk Data: AUTOSPC,YES; GPL; SEQOUT,1. The output from the AUTOSPC processor should be reviewed carefully (see **Table 2-12** for an example of AUTOSPC). This table will include all degrees-of-freedom that have no stiffness. Each degree-of-freedom should be checked and, if singular, it should be SPC’d; if it is not intended to be singular, then the model must be corrected.

**Table 2-12. Grid Point Singularity Table**

| POINT ID | TYPE | FAILED DIRECTION | STIFFNESS RATIO | OLD USET | NEW USET |
|---|---|---|---|---|---|
| 1000 | G | 4 | 0.00E+00 | O | S |
| 1000 | G | 5 | 0.00E+00 | O | S |
| 1000 | G | 6 | 0.00E+00 | O | S |

The KRBF matrix (see Table 2-13) is printed out and is a measure of the force required for the rigid body displacements. All elements should be small, e.g.,

- Diagonal translations < $1.0 \times 10^{-2}$
- Diagonal rotation < $2.0 \times 10^{2}$
- Off-diagonal terms < $2.0 \times 10^{2}$

**Table 2-13. KRBF Matrix**

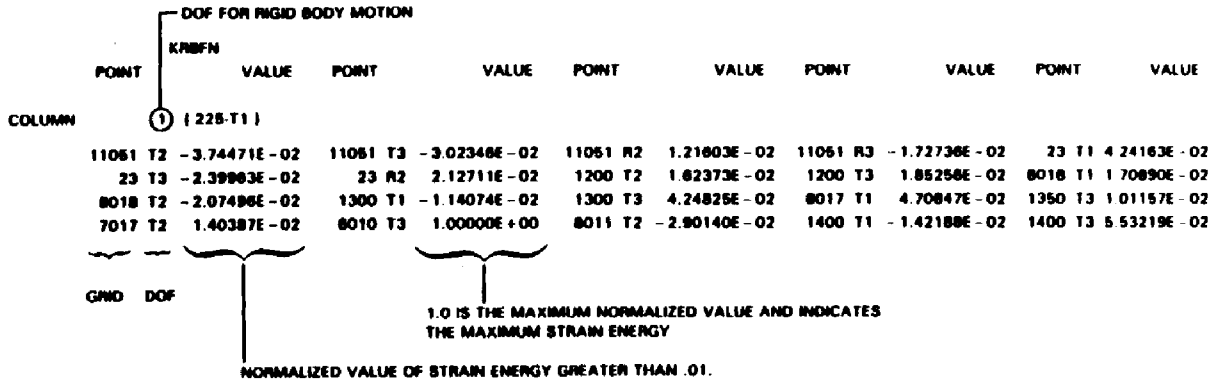
MATRIX KRBF ( GINO NAME 101 ) IS A DB PREC 6 COLUMN X 6 ROW SQUARE MATRIX.

| | | | | | | |
|---|---|---|---|---|---|---|
| COLUMN 1 ROWS 1 THRU 6 | | | | | | |
| ROW 1) | 5.6036D-06 | 6.2035D-08 | -9.8440D-08 | -8.1671D-07 | 1.7130D-04 | 1.1562D-05 |
| COLUMN 2 ROWS 1 THRU 6 | | | | | | |
| ROW 1) | 6.2502D-08 | 6.3425D-06 | 8.2305D-07 | -8.1379D-05 | 1.1740D-07 | -9.2047D-06 |
| COLUMN 3 ROWS 1 THRU 6 | | | | | | |
| ROW 1) | -9.7452D-08 | 8.2517D-07 | 5.5577D-06 | -1.9905D-05 | -1.3735D-06 | 1.2302D-06 |
| COLUMN 4 ROWS 1 THRU 6 | | | | | | |
| ROW 1) | -7.2425D-07 | -8.2143D-05 | -2.0377D-05 | 2.5692D+01 | -3.6712D-04 | 2.2372D-04 |
| COLUMN 5 ROWS 1 THRU 6 | | | | | | |
| ROW 1) | 1.7143D-04 | 7.7165D-08 | -3.1777D-06 | -3.7383D-04 | 7.8407D+00 | 4.3714D-01 |
| COLUMN 6 ROWS 1 THRU 6 | | | | | | |
| ROW 1) | 1.1508D-05 | -9.3303D-06 | 1.1475D-06 | 2.3130D-04 | 4.3714D-01 | 1.4307D+00 |

- Forces due to translations (circled) should be less than 1.E–2.
- Moments due to rotations (rectangle) should be less than 2.0E+2.
- Off diagonal terms should be less than 2.0E+2.

Also printed in the matrix KRBFN, the forces at the grid points normalized to a maximum of 1.0 (see Table 2-14). As seen from Table 2-14, this check easily identifies which grid points and DOFs are causing problems.

**Table 2-14. KRBFN Matrix**




## Rigid Body Check

If the model is primarily used for dynamic analysis, the CHECKA DMAP alter performs an equilibrium check in Solution 103. In this case, the unit translations and rotations are considered to be the rigid body displacements of the structure in output coordinates about the SUPORT point. Strain energy, which should be very small, is calculated and printed out.

The strain energy, the Epsilons, and the Max Ratio should also be checked at this point, as they were for the static case (see Table 2-15).

**Table 2-15. Epsilon, Strain Energy, and Max Ratio from Rigid Body Modes Run**

```
*** USER INFORMATION MESSAGE 4158-STATISTICS FOR SYMMETRIC DECOMPOSITION OF DATA BLOCK KOO
    FOLLOW MAXIMUM RATIO OF MATRIX DIAGONAL TO FACTOR DIAGONAL = 7.4E+04 AT ROW NUMBER 702

*** USER INFORMATION MESSAGE 3035 FOR DATA BLOCK KLR

SUPPORT
 PT. NO.       EPSILON          STRAIN  ENERGY   EPSILONS LARGER THAN 0.001 ARE FLAGGED WITH ASTERISKS

      1   1.2895072E-16    0.0000000E+00
      2   1.2895072E-16   -7.2759576E-12
      3   1.2895072E-16   -5.4569682E-12
      4   1.2895072E-16   -2.2351742E-08
      5   1.2895072E-16   -4.4703484E-08
      6   1.2895072E-16   -7.4505806E-09
```

- The SUPORT entry is required for epsilon and strain energy.

This option requires a SUPORT entry. The DOF supported must form a determinate interface or high strain energy will result. The model displacements printed represent the displacements caused by moving a support DOF one unit while holding the other support DOFs fixed. The model displacements should be checked for unit value (see Table 2-16).

**Table 2-16. Rigid Body Mode Shapes**

EIGENVALUE = 0.000000E+00
CYCLES = 0.000000E+00

REAL EIGENVECTOR NO. 2

| POINT ID. | TYPE | T1 | T2 | T3 | R1 | R2 | R3 |
|---|---|---|---|---|---|---|---|
| 2001 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2002 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2003 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2102 | G | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2131 | G | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2704 | G | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2732 | G | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 | 0.0 |
| 7010 | G | 4.446353E-01 | 8.957117E-01 | 0.0 | 0.0 | 0.0 | 0.0 |
| 7011 | G | -4.446353E-01 | 8.957117E-01 | 0.0 | 0.0 | 0.0 | 0.0 |
| 7012 | G | -4.446353E-01 | 8.957117E-01 | 0.0 | 0.0 | 0.0 | 0.0 |

GRIDS 7010 THRU 7012 OUTPUT
IN A LOCAL COORDINATE SYSTEM

EIGENVALUE = 0.000000E+00
CYCLES = 0.000000E+00

REAL EIGENVECTOR NO. 3

| POINT ID. | TYPE | T1 | T2 | T3 | R1 | R2 | R3 |
|---|---|---|---|---|---|---|---|
| 2001 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2002 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2003 | G | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2102 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |
| 2131 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |
| 2704 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |
| 2732 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |
| 7010 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |
| 7011 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |
| 7012 | G | 0.0 | 0.0 | 1.000000E+00 | 0.0 | 0.0 | 0.0 |

- Eigenvalues and cycles are set equal to zero by the DMAP alter package.

Please note that these alters assume max normalization on the EIGR entry.

## Thermal Test Case

As a further check on connectivity and the stiffness matrix, an isothermal expansion test case can be run with a statically determinate interface. This is done on SOL 101 with a TEMPD for the constant temperature load. All of the coefficients of expansion should be set to the same value. This check will not be rigid if rigid elements or bar offsets are present. Rigid elements will not expand and may generate distortion forces and stresses unless the appropriate degrees-of-freedom are released.

## Modal Analysis

There are several diagnostic tools that can be used to further assess the integrity of the model. These tools (effective mass, strain energy, kinetic energy, deformed plots) are outlined in the following paragraphs.

The two major means of reducing the number of dynamic degrees-of-freedom in the modal analysis are outlined below.

A major concern in dynamic analysis is the choice of an appropriate ASET when using the Guyan reduction. The quality of the solution depends upon the reduced mass matrix formed by this ASET. Will it retain sufficient mass in correct distribution to adequately predict mode shapes and frequencies?

The results of a Component Mode Reduction run can be used to select an ASET for use in future processing if desired (see the flow chart in **Figure 2-7**). A proven rule-of-thumb for ASET selection is to include all DOFs that have more than 2% of the system KE for all major modes (determined by comparing EFMASS to the system weight) and more than 5% for the other modes. One must be careful in the case of assemblies with a fine mesh; although the whole assembly may be moving in a mode, there may not be any individual DOFs with greater than 2% of the system kinetic energy. This is usually evident when a mode has a sizeable EFMASS, and either no terms with a large kinetic energy or the terms with noticeable kinetic energy do not account for the EFMASS of the mode. If this is the case, then there is no substitute for common sense determination of DOFs that describe the subsystem motion.

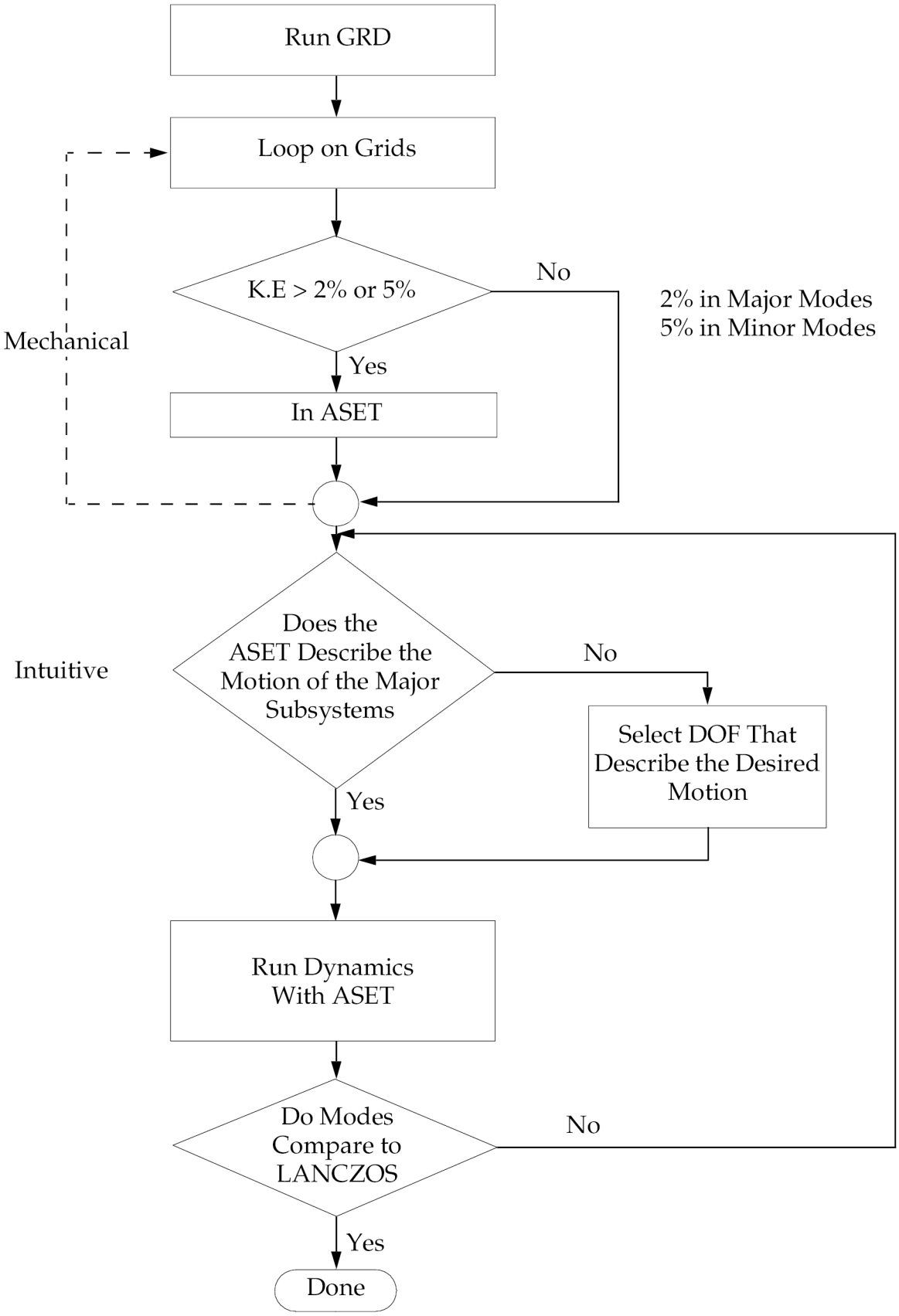



**Figure 2-7. ASET Selection Flow Chart**

Note also that this removes the rigid body mode check from the run, since the SUPORT entry must be removed (unless the run is a free-free run).

## MER – MREMER – MR – EFMASS

An important diagnostic tool is the elastic-rigid coupling matrix (MER), a triple product of the elastic mode shape matrix, the ASET mass matrix, and the rigid body displacement matrix with respect to the interface. This *N*×*6* matrix gives the square root of the effective mass in each retained mode so that the product of each element of this matrix will itself give the effective mass contained in that mode in the associated direction. These values are in EFMASS. This matrix can be used to determine which modes are energetic in terms of interface loading. **Table 2-17** shows a sample MER matrix taken from **Reference 3** . The elements of this matrix have been squared and units converted to give recognizable weight units.

**Table 2-17. Galileo Spacecraft Baseline Model Modal Weight and Inertia (Lb, Lb-in 2)**

| Mode | X | Y | Z | RX | RY | RZ |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 2.5E+00 | 2.3E+03 | 8.3E+01 |
| 2 | 0 | 0 | 1 | 2.3E+03 | 3.4E-01 | 5.0E-02 |
| 3 | 1188 | 479 | 0 | 7.9E+06 | 2.1E+07 | 2.2E+03 |
| 4 | 489 | 1144 | 0 | 2.0E+07 | 8.7E+06 | 6.2E+02 |
| 5 | 2 | 1888 | 3 | 8.2E+06 | 1.2E+04 | 1.0E+02 |
| 6 | 1097 | 5 | 0 | 1.7E+05 | 5.7E+06 | 9.2E+05 |
| 7 | 1007 | 10 | 0 | 3.1E+05 | 5.5E+06 | 3.2E+05 |
| 8 | 1 | 296 | 0 | 5.2E+06 | 8.9E+03 | 2.1E+05 |
| 9 | 45 | 3 | 0 | 9.4E+04 | 2.4E+05 | 1.2E+05 |
| 10 | 11 | 554 | 33 | 3.5E+06 | 5.3E+04 | 1.5E+02 |
| 11 | 533 | 0 | 31 | 2.0E+02 | 3.2E+06 | 2.8E+04 |
| 12 | 5 | 6 | 221 | 3.5E+04 | 2.9E+04 | 9.0E+05 |
| 13 | 41 | 25 | 770 | 1.1E+05 | 1.9E+05 | 4.0E+05 |
| 14 | 3 | 29 | 84 | 1.8E+05 | 3.5E+04 | 7.0E+05 |
| 15 | 9 | 0 | 7 | 7.9E+02 | 9.2E+04 | 5.2E+05 |
| 16 | 0 | 24 | 6 | 3.3E+04 | 3.4E+02 | 1.4E+04 |
| 17 | 2 | 0 | 63 | 3.9E+04 | 4.6E+03 | 3.6E+03 |
| 18 | 50 | 86 | 1 | 6.9E+04 | 7.7E+04 | 6.1E+05 |
| 19 | 2 | 0 | 35 | 2.8E+04 | 3.4E+03 | 1.1E+04 |
| 20 | 27 | 196 | 0 | 1.6E+05 | 3.4E+04 | 3.8E+05 |
| 21 | 55 | 6 | 1 | 2.4E+03 | 7.3E+04 | 7.8E+03 |
| 22 | 15 | 51 | 40 | 4.6E+04 | 2.9E+04 | 1.2E+04 |
| 23 | 6 | 19 | 3055 | 2.2E+04 | 2.6E+04 | 6.5E+04 |
| 24 | 100 | 0 | 479 | 4.3E+04 | 2.9E+05 | 1.2E+05 |
| 25 | 8 | 15 | 5 | 1.0E+04 | 2.5E+04 | 1.4E+04 |
| 26 | 157 | 7 | 15 | 3.6E+03 | 2.2E+05 | 5.2E+03 |
| 27 | 1 | 1 | 149 | 3.1E+04 | 1.8E+03 | 1.5E+04 |
| 28 | 21 | 1 | 30 | 7.0E+04 | 2.8E+04 | 1.5E+04 |
| 29 | 0 | 1 | 0 | 6.6E+03 | 1.6E+03 | 5.0E+00 |
| 30 | 10 | 1 | 0 | 1.2E+03 | 1.2E+04 | 9.0E+03 |
| 31 | 71 | 138 | 9 | 4.2E+05 | 1.3E+05 | 7.0E+03 |
| 32 | 67 | 5 | 0 | 2.6E+04 | 5.3E+04 | 1.6E+02 |
| 33 | 22 | 38 | 20 | 8.9E+04 | 6.7E+04 | 3.8E+02 |
| 34 | 2 | 47 | 6 | 1.3E+03 | 4.2E+04 | 3.1E+01 |
| 35 | 67 | 49 | 1 | 8.2E+04 | 2.0E+05 | 2.2E+02 |
| 36 | 0 | 0 | 0 | 1.5E+02 | 1.4E+01 | 1.4E+01 |
| 37 | 0 | 0 | 0 | 3.5E+02 | 4.1E+02 | 1.2E+02 |
| 38 | 0 | 10 | 0 | 5.4E+04 | 2.7E+04 | 3.0E+02 |
| 39 | 13 | 1 | 6 | 4.7E+03 | 8.6E+03 | 1.8E+03 |
| 40 | 6 | 22 | 0 | 1.5E+03 | 1.8E+03 | 1.8E+04 |
| 41 | 7 | 2 | 8 | 2.3E+01 | 3.4E+03 | 8.7E+03 |
| 42 | 12 | 11 | 41 | 4.9E+04 | 1.4E+04 | 5.6E+03 |
| 43 | 0 | 8 | 1 | 8.9E+03 | 3.6E+01 | 1.9E+04 |

Galileo Spacecraft Baseline Model Modal Weight and Inertia (Lb, Lb-in$^2$) (Cont.)

| Mode | X | Y | Z | RX | RY | RZ |
|---|---|---|---|---|---|---|
| 44 | 0 | 3 | 8 | 1.4E+04 | 1.2E+03 | 2.9E+04 |
| 45 | 0 | 0 | 49 | 1.3E+04 | 4.1E+01 | 6.7E+03 |
| 46 | 9 | 29 | 0 | 1.7E+04 | 2.2E+04 | 5.3E+03 |
| 47 | 1 | 0 | 4 | 1.6E+03 | 6.2E-01 | 4.9E+03 |
| 48 | 0 | 2 | 13 | 1.9E+04 | 5.0E+02 | 4.1E+03 |
| 49 | 4 | 3 | 66 | 1.2E+04 | 1.8E+04 | 1.7E+04 |
| 50 | 0 | 0 | 4 | 1.0E+03 | 6.7E+02 | 6.1E+02 |
| 51 | 0 | 18 | 0 | 1.6E+04 | 6.2E+03 | 3.7E-02 |
| 52 | 1 | 4 | 3 | 1.8E+04 | 8.9E+02 | 4.6E+02 |
| 53 | 2 | 7 | 1 | 2.6E+04 | 4.1E+03 | 4.2E+02 |
| 54 | 0 | 0 | 1 | 2.6E+02 | 7.5E+03 | 8.3E+02 |
| 55 | 14 | 3 | 0 | 1.4E+04 | 8.4E+04 | 2.5E+02 |
| 56 | 0 | 6 | 0 | 7.5E+03 | 9.3E+03 | 2.8E+03 |
| 57 | 9 | 14 | 1 | 1.1E+04 | 1.3E+03 | 3.5E+02 |
| 58 | 0 | 0 | 2 | 1.8E+03 | 2.6E+03 | 4.6E+03 |
| 59 | 27 | 15 | 0 | 3.8E+04 | 3.6E+04 | 8.4E+03 |
| 60 | 1 | 25 | 4 | 6.4E+04 | 6.5E+03 | 3.6E+03 |
| 61 | 42 | 12 | 3 | 3.7E+04 | 6.8E+04 | 9.1E+03 |
| 62 | 15 | 11 | 0 | 1.5E+04 | 2.6E+04 | 3.3E+04 |
| 63 | 0 | 1 | 0 | 1.4E+03 | 3.8E+03 | 2.8E+00 |
| 64 | 31 | 11 | 0 | 3.5E+03 | 6.6E+04 | 1.2E+05 |
| 65 | 3 | 1 | 1 | 1.0E+02 | 6.9E+02 | 2.1E+04 |
| 66 | 0 | 0 | 0 | 6.6E+02 | 3.4E+01 | 1.5E+03 |
| 67 | 2 | 0 | 4 | 4.5E+02 | 1.3E+04 | 1.4E+04 |
| 68 | 3 | 3 | 0 | 3.4E+03 | 9.3E+02 | 1.7E+03 |
| 69 | 0 | 0 | 2 | 4.1E-01 | 2.5E+02 | 8.1E+01 |
| 70 | 0 | 8 | 0 | 4.1E+04 | 4.7E+01 | 1.0E+04 |
| TOTAL: | 5316 | 5355 | 5287 | 4.69E+07 | 4.61E+07 | 5.76E+06 |
| RIGID | 5591 | 5591 | 5591 | 4.74E+07 | 4.67E+07 | 6.19E+06 |
| MODAL% | 0.95 | 0.96 | 0.95 | 0.99 | 0.99 | 0.93 |

The product of MER with itself transposed (MREMER) gives the total effective mass retained using the ASET DOF number of modes. The diagonal of MREMER is compared with that of the rigid mass matrix (MR) to determine if sufficient mass is contained in the selected modes to consider the model valid. The generation of the MR matrix also makes use of the stiffness matrix (see the sketch below). The MR matrix should also be compared to the mass properties of the Grid Point Weight Generator. The inertias are calculated about the SUPORT grid and, unless this is the point selected with PARAM,GRDPNT, the inertias will not agree. A typical allowable is a 5% loss of mass. This comparison can then be used to modify the choice of ASET DOF or to increase the number of modes. This diagonal is shown as the TOTAL line of **Table 2-17** and is compared against the full model weight.

Table 2-18. Galileo Baseline S/C Modal Frequencies

| Mode | Freq | Mode Description | Kinetic Energy Distribution |
|---|---|---|---|
| 1 | 6.3 | PWS Antenna | PWS, 100% |
| 2 | 6.8 | PWS Antenna | PWS, 100% |
| 3 | 13.7 | SXA bending, X | SXA 45%, RPM 22% |
| 4 | 13.7 | SXA bending, Y | SXA 48%, RPM 21% |
| 5 | 16.7 | Probe, Y | Probe 83% |
| 6 | 18.2 | Sciboom X, core torsion | Sciboom 45%, SXA 20% |
| 7 | 18.5 | Probe X | Probe 53% |
| 8 | 18.6 | SXA Y | SXA 41%, Probe 9% |
| 9 | 19.0 | Probe X, SXA X | SXA 34%, Probe 32% |
| 10 | 22.2 | Sciboom Z bounce | Sciboom 57% |
| 11 | 22.7 | RTGs Z bounce, -phase | RTGs 61%, RPM 19% |
| 12 | 23.8 | Nutation damper lateral | Nutation Damper 51% |
| 13 | 24.3 | RTGs Z bounce, in phase | RTGs 61% |
| 14 | 25.7 | Nutation damper lateral | Nutation Damper 52%, |
| 15 | 27.0 | Nutation damper lateral | Nutation Damper 70%, |
| 16 | 28.5 | RRH Antenna, Y | RRH 48%, EPD 22% |
| 17 | 28.7 | Scan platform theta X | Scan platform 96% |
| 18 | 29.7 | -X RTG lateral (Y) | -X RTG 96% |
| 19 | 31.9 | Scan sunshade (Z) | Sunshade 92% |
| 20 | 32.5 | +X RTG lateral (Y) | +X RTG 93% |
| 21 | 33.7 | PLS+RRH+Sciboom | PLS 31% |
| 22 | 35.4 | EPD+PLS+Sciboom | PLS 31% |
| 23 | 38.0 | S/C Z bounce | RPM 54%, Probe 10% |
| 24 | 38.3 | Scan platform X | Scan 73%, RPM 16% |
| 25 | 38.6 | PLS+EPD+Thrusters | PLS 40%, EPD 21% |
| 26 | 40.4 | Probe torsion, RPM | Probe 50% |
| 27 | 40.8 | -X Thruster, RPM | RPM 53% |
| 28 | 41.3 | Sciboom mag tip, box | Sciboom 47% |
| 29 | 43.1 | Thrusters lateral (Y) | Thrusters 89% |
| 30 | 43.3 | RRH antenna X | RRH 95% |
| 31 | 44.9 | RPM tanks, probe torsion | RPM 38%, Probe 17% |
| 32 | 45.8 | -X RTG X motion | -RTG 57% |
| 33 | 46.1 | Sciboom, misc | Sciboom 49%, Scan 20% |
| 34 | 46.7 | Scan platform Z, X | Scan platform 53% |
| 35 | 47.7 | 400N engine, Probe tors | 400N 13%, Probe 10% |
| 36 | 48.0 | -X Thruster torsion | -X Thruster 95% |
| 37 | 48.1 | +X Thruster torsion | +X Thruster 95% |
| 38 | 48.3 | 400N engine X, SBA | 400N 56%, SBA 27% |
| 39 | 48.8 | 400N engine Y, SBA | 400N 43%, SBA 19% |
| 40 | 49.6 | RPM+Despun Box Z bounce | RPM 32%, Despun box 17% |

To calculate the MER and MREMER and to print them along with the MR, one should use the CHECKA DMAP alter from the SSALTER library when there is a SUPORT entry in the Bulk Data Section. The model must be cantilevered from the SUPORT; that is, the SUPORT DOFs are the only constraints preventing rigid-body motion. The G factor 385.0886 in/sec$^2$) is in the ALTER. If another value is desired, it must be substituted.

## Kinetic and Strain Energy

The CHECKA alters may also be used for printing the kinetic energy distribution as a fraction of the total, in the form of eigenvectors, for each mode. This information is useful for showing the energy distribution *within a mode*, whereas the MER matrix is useful for determining *which mode* is energetic. The various grid points within a model can be grouped together and the energy then given by subsystem. The table **Galileo Baseline S/C Modal Frequencies** shows a sample taken from **Reference 3**. Examination of the tables **Galileo Spacecraft Baseline Model Modal Weight and Inertia (Lb, Lb-in 2)** and **Galileo Baseline S/C Modal Frequencies** show that for modes 1 and 2 the PWS subsystem has 100% of the KE, but in terms of effective mass it is negligible (the Plasma Wave Subsystem (PWS) is a very light antenna). In contrast, the probe in mode 5 not only has the predominant percent KE but is also an energetic mode (see the table **Galileo Baseline S/C Modal Frequencies**).

Element strain energy can also be used to determine which elastic elements are participating within a given mode. These two diagnostic tools can help isolate a weak or very flexible area of the model that may only be a modeling error.

The whole package is intended for use with SOL 3. The model must have a determinate support point without any SPCs to fix the base.

## Deformed Plots

Another diagnostic tool is plotting. Deformed geometry plots, the eigenvector output, and MER help to identify mode shapes and classify them. Plots will also highlight excessive deflections that point to stiffness matrix problems.

Information from these diagnostics will feed back to both the stiffness and mass matrices and the ASET degrees-of-freedom. In only a few iterations, all mass and stiffness problems can be rectified and the model will be a realistic representation of the true structure.

The quality of the above checks depends on the quality of ASET chosen. It is good practice to make at least one run using Guyan Reduction and to verify the model by comparing the resulting modes to those obtained using the ASET. If the two runs are not comparable, the LANCZOS run should be used to select an ASET. The inclusion of all terms with significant KE usually gives a good ASET, but since large, stiff assemblies might not have large energy at individual grid points, common sense should also be used.

# References

1. *MSC.Nastran Handbook for Linear Analysis*, Version 64, The MacNeal-Schwendler Corporation, August 1985.

2. “Equivalent Spring-Mass System for Normal Modes,” R. M. Bamford, B. K. Wada, and W. H. Gayman, NASA Technical Memorandum 33-380, February 1971.

3. “Galileo S/C NASTRAN Model, Descriptive Data, and Checkcase Results,” R. Calvet and G. Wang, May 1985.

4. “Checks that Pay,” J. C. Auker, *Proceedings*, the MSC.Nastran European User’s Conference, Munich, West Germany, May 1984.

5. JSC-20545, “Simplified Design Options for STS Payloads,” D. A. Hamilton, May 1985.

# 3. System Matrix Formulations

## 3.1 Overview of System Matrix Formulations

The solutions used in dynamic analysis are more complicated than the solution of a simple matrix equation. Compared to linear static analysis, the running time for a dynamics problem is more costly, the required file space is more excessive, and the larger amounts of output data can be difficult to digest. Therefore, attention should be paid to reducing the number of solution equations to manageable levels.

Other special effects enter into many cases of dynamic analysis. Certain loads, such as those used in rotating structures and in aerodynamics, require special matrix formulations and solution strategies. These problems use conventional matrices to represent structural properties, but they can also require extra degrees-of-freedom and may need to account for fundamental changes such as nonlinear effects. For example, in rotating structures the connections vary with phase angles. In aeroelasticity the effects of the air flow vary with frequency.

This chapter begins with an explanation of the various options available in Simcenter Nastran for reducing the size of the solution matrices in dynamic analysis. Direct methods, modal formulations, and combinations are discussed.

The next section presents methods for the analysis of finite element models in rotating coordinate systems. Applications include turbo machinery, disk drives, and flexible shafts.

The last section is an introduction to the use of aeroelasticity, which can be applied to any problem with a steady fluid flow attached to a flexible structure.

## 3.2 Theoretical Basis for Reduction Methods

In the following development we will start from the full-size structural matrix equations and derive the equations for Static Condensation, Guyan Reduction, and Component Mode Synthesis. These operations will apply to both single structures and superelement models. We will also try to explain the physical consequences of the assumptions involved in reducing the systems.

### Definition

The basic dynamic equation before reduction is given in the $u_f$ set (after SPC and MPC constraints have been applied, but before DMIGs and extra points). The standard matrix equation to be reduced is:

$$\left[\begin{array}{c:c} \bar{M}_{aa} & M_{ao} \\ \hdashline M_{oa} & M_{oo} \end{array}\right] \left\{\begin{array}{c} \ddot{u}_a \\ \hdashline \ddot{u}_o \end{array}\right\} + \left[\begin{array}{c:c} \bar{B}_{aa} & B_{ao} \\ \hdashline B_{oa} & B_{oo} \end{array}\right] \left\{\begin{array}{c} \dot{u}_a \\ \hdashline \dot{u}_o \end{array}\right\} + \left[\begin{array}{c:c} \bar{K}_{aa} & K_{ao} \\ \hdashline K_{oa} & K_{oo} \end{array}\right] \left\{\begin{array}{c} u_a \\ \hdashline u_o \end{array}\right\} = \left\{\begin{array}{c} \bar{P}_a \\ \hdashline P_o \end{array}\right\}$$

**Equation 3-1.**

where:

| | |
|---|---|
| $u_a$, $\dot{u}_a$, $\ddot{u}_a$ | are the displacements, velocities, and accelerations of the analysis (a) set, to be retained. |
| $u_o$, $\dot{u}_o$, $\ddot{u}_o$ | are the displacements, velocities, and accelerations of the omit (o) set, to be eliminated. |
| $M$, $B$, $K$ | are the mass, damping, and stiffness matrices (assumed to be real and symmetric). |
| $\bar{P}_a$, $P_o$ | are the applied loads. |

Note that all free-body motions must be included in the $u_a$ partition. Otherwise, $K_{oo}$ will be singular. The bar quantities ($\bar{P}$, etc.) indicate unreduced values.

## Statics

For statics problems, we may ignore the mass and damping effects and solve the lower partition of the equation in **Definition** for $u_o$:

$$\{u_o\} = -[K_{oo}^{-1}]([K_{oa}]\{u_a\} + \{P_o\})$$

**Equation 3-2.**

The two parts of **Eq. 3-2** become the Guyan matrix $G_o$ and the static corrective displacement $u^o_o$:

$$[G_o] = -[K_{oo}^{-1}][K_{oa}]$$

**Equation 3-3.**

$$\{u_o^o\} = [K_{oo}^{-1}]\{P_o\}$$

**Equation 3-4.**

The exact static solution system is obtained by substituting **Eq. 3-2** through **Eq. 3-4** into the upper partition terms of the equation in **Definition**, resulting in the reduced equations used in the static solution

$$[K_{aa}]\{u_a\} = \{P_a\}$$

**Equation 3-5.**

and

$$\{u_o\} = [G_o]\{u_a\} + \{u_o^o\}$$

**Equation 3-6.**

where:

$$[K_{aa}] = [\bar{K}_{aa}] + [K_{ao}][G_o]$$

**Equation 3-7.**

$$\{P_a\} = \{\bar{P}_a\} + [G_o^T]\{P_o\}$$

**Equation 3-8.**

In actual practice the size of the $u_a$ set is usually small compared to $u_o$, but the reduced matrices are dense, resulting in no savings in cost. The savings in solving **Eq. 3-5** are usually offset by the costs of calculating $G_o$ and $u^o_o$.

However, for dynamics, we also may approximate the vectors $\ddot{u}_o$ and $\dot{u}_o$ to reduce the order of the system. A good place to start is to use the static properties. From **Eq. 3-6**, define the transformation

$$\{u_f\} = \begin{Bmatrix} u_a \\ \hdashline u_o \end{Bmatrix} = [H_f]\{u_f'\}$$

**Equation 3-9.**

where:

$$\{u_f'\} = \begin{Bmatrix} u_a \\ \hdashline u_o^o \end{Bmatrix}$$

**Equation 3-10.**

$$[H_f] = \left[\begin{array}{c:c} I & 0 \\ \hdashline G_o & I \end{array}\right]$$

**Equation 3-11.**

Here $u^o_o$ are the incremental displacements relative to the static shape. The system described in the equation in **Definition** may be transformed to the new coordinates with no loss of accuracy. The stiffness matrix in the transformed system is

$$[K'_{ff}] = \begin{bmatrix} I & G_o^T \\ 0 & I \end{bmatrix} \begin{bmatrix} \overline{K}_{aa} & K_{ao} \\ K_{oa} & K_{oo} \end{bmatrix} \begin{bmatrix} I & 0 \\ G_o & I \end{bmatrix}$$

**Equation 3-12.**

Performing the multiplication and substituting **Eq. 3-3** results in

$$[K'_{ff}] = \begin{bmatrix} K_{aa} & 0 \\ 0 & K_{oo} \end{bmatrix}$$

**Equation 3-13.**

Although the stiffness matrix becomes decoupled, the mass and damping matrices tend to have more coupling than the original system. Since the damping terms have the same form as the mass, we will not include them here. The exact transformed system becomes

$$\begin{bmatrix} M'_{aa} & M'_{ao} \\ M'_{oa} & M'_{oo} \end{bmatrix} \begin{Bmatrix} \ddot{u}_a \\ \ddot{u}_o \end{Bmatrix} + \begin{bmatrix} K_{aa} & 0 \\ 0 & K_{oo} \end{bmatrix} \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \begin{Bmatrix} P_a \\ P_o \end{Bmatrix}$$

**Equation 3-14.**

where:

$$[M'_{aa}] = [M_{aa}] + [M_{ao}][G_o] + [G_o]^T[M_{oa} + M_{oo}G_o]$$

**Equation 3-15.**

$$[M'_{ao}] = [M'^T_{oa}] = [M_{ao}] + [G_o^T M_{oo}]$$

**Equation 3-16.**

$$[M'_{oo}] = [M_{oo}]$$

**Equation 3-17.**

The damping matrix terms of $B'_{ff}$ are similar in form to the mass matrix partitions. An alternative derivation which does not rely on symmetric transformation is given below. Starting from equation

$$\begin{bmatrix} \bar{M}_{aa} & M_{ao} \\ M_{oa} & M_{oo} \end{bmatrix} \begin{Bmatrix} \ddot{u}_a \\ \ddot{u}_o \end{Bmatrix} + \begin{bmatrix} \bar{B}_{aa} & B_{ao} \\ B_{oa} & B_{oo} \end{bmatrix} \begin{Bmatrix} \dot{u}_a \\ \dot{u}_o \end{Bmatrix} + \begin{bmatrix} \bar{K}_{aa} & K_{ao} \\ K_{oa} & K_{oo} \end{bmatrix} \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \begin{Bmatrix} \bar{P}_a \\ P_o \end{Bmatrix}$$

in **Definition** through **Eq. 3-8** in that topic, we may estimate the acceleration effects of the omitted points by the equation

$$\{\ddot{u}_o\} \cong [G_o]\{\ddot{u}_a\}$$

**Equation 3-18.**

Substituting **Eq. 3-18** into the lower partition of the equation in **Definition**, and solving for $u_o$, with damping neglected, we obtain the approximation

$$\{u_o\} = [K_{oo}^{-1}](\{P_o\} - [K_{oa}]\{u_a\} - [M_{oa} + M_{oo}G_o]\{\ddot{u}_a\})$$

**Equation 3-19.**

Substituting **Eq. 3-3** for $K_{oa}$ and **Eq. 3-16** for the mass terms into **Eq. 3-19**, we obtain

$$\{u_o\} \cong [G_o]\{u_a\} + K_{oo}^{-1}[\{P_o\} - [M'_{oa}]\{\ddot{u}_a\}]$$

**Equation 3-20.**

Substituting **Eq. 3-18** and **Eq. 3-20** into the upper half of the equation in **Definition** (ignoring damping), we obtain

$$[\bar{M}_{aa} + M_{ao}G_o]\{\ddot{u}_a\} + [\bar{K}_{aa} + K_{ao}G_o]\{u_a\} - [K_{ao}K_{oo}^{-1}][M_{oa} + M_{oo}G_o]\{\ddot{u}_a\} = \{\bar{P}_a\} - [K_{ao}][K_{oo}^{-1}]\{P_o\}$$ Eq. 3-21

**Equation 3-21.**

Combining the terms, we obtain the same results as **Eq. 3-14** through **Eq. 3-17**.

The significance of this exercise is to show that the Guyan transformation has very interesting properties, namely:

1. The approximation occurs only on the acceleration terms (**Eq. 3-18**).

2. The stiffness portion of the reduced system is exact.

3. The interior displacements defined by **Eq. 3-20** and **Eq. 3-14** are nearly identical.

The significant aspects of the partially decoupled system described by **Eq. 3-9** through **Eq. 3-21** are that most of the Simcenter Nastran reduction methods are easily developed from this form and the approximations are conveniently explained in these terms. The Guyan reduction and the modal synthesis methods are described below.

## Guyan Reduction

In the software's Guyan reduction process, the omitted relative accelerations, $\ddot{u}^o_o$, in

$$\left[\begin{array}{c:c} M'_{aa} & M'_{ao} \\ \hdashline M'_{oa} & M'_{oo} \end{array}\right] \left\{\begin{array}{c} \ddot{u}_a \\ \hdashline \ddot{u}_o \end{array}\right\} + \left[\begin{array}{c:c} K_{aa} & 0 \\ \hdashline 0 & K_{oo} \end{array}\right] \left\{\begin{array}{c} u_a \\ \hdashline u_o \end{array}\right\} = \left\{\begin{array}{c} P_a \\ \hdashline P_o \end{array}\right\}$$

(for more information, see **Statics**) are approximated by

$$\{\ddot{u}_o\} \cong [G_o]\{\ddot{u}_a\}$$

(for more information, see **Statics**) and the solution system is

$$[M'_{aa}]\{\ddot{u}_a\} + [K'_{aa}]\{u_a\} = \{P'_a\}$$

**Equation 3-22.**

The system described by **Eq. 3-22** has several desirable properties. The overall mass and center of gravity properties are preserved in the mass matrix. Also note that the static stiffness is exact. If accelerations occur, the errors may be estimated by solving for $u^o_o$ after $u_a$ is obtained. The omitted points, $u^o_o$, could be recovered by solving the lower part of the equation

$$\begin{bmatrix} M'_{aa} & M'_{ao} \\ M'_{oa} & M'_{oo} \end{bmatrix} \begin{Bmatrix} \ddot{u}_a \\ \ddot{u}_o \end{Bmatrix} + \begin{bmatrix} K_{aa} & 0 \\ 0 & K_{oo} \end{bmatrix} \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \begin{Bmatrix} P_a \\ P_o \end{Bmatrix}$$

However, in most applications, the stiffness terms of the above equation dominate and the normal modes of the omitted degrees-of-freedom are of higher frequency than the solution set. Therefore, we may ignore the left-hand mass and solve for $u_o$ directly from

$$\{u_o\} = -[K_{oo}^{-1}]([K_{oa}]\{u_a\} + \{P_o\})$$

(for more information, see **Statics**) by the equation

$$[u_o] = [G_o]\{u_a\}$$

**Equation 3-23.**

The errors in **Eq. 3-23** are proportional to the vector $\{u^o_o\}$, which becomes small when the individual masses and applied loads in the omit set are small. Because of the same assumption, the errors of the upper half of

$$\begin{bmatrix} M'_{aa} & M'_{ao} \\ M'_{oa} & M'_{oo} \end{bmatrix} \begin{Bmatrix} \ddot{u}_a \\ \ddot{u}_o \end{Bmatrix} + \begin{bmatrix} K_{aa} & 0 \\ 0 & K_{oo} \end{bmatrix} \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \begin{Bmatrix} P_a \\ P_o \end{Bmatrix}$$

(for more information, see **Statics**) are also small. (In Simcenter Nastran, $u^o_o$ is neglected in the dynamic data recovery process.)

## Dynamic Reduction

In most applications, dynamic reduction is used directly to find the modes of the $u_f$ set and no $u_a$ points are retained. The method then becomes a straightforward Rayleigh-Ritz/Lanczos procedure. However, it is a recommended practice in Simcenter Nastran to extend the $u_a$ set to include any large masses to avoid numerical roundoff problems and provide more accurate results.

### Approximate Mode Shapes

If $u_a$ grid point displacements are retained in the dynamic reduction process, the $u^o_o$ set is approximated by a set of generalized coordinates, $u_q$, where

$$\{u_o^o\} \cong [\phi_{oq}]\{u_q\}$$

**Equation 3-24.**

where $\varphi_{oq}$ are a set of approximate eigenvectors or natural shapes. In its simple form[1], with $u_t$ replacing $u_a$, the dynamic reduction transformation corresponding to

$$\{u_f\} = \begin{Bmatrix} u_a \\ \hdashline u_o \end{Bmatrix} = [H_f]\{u_f'\}$$

(for more information, see **Statics**) is

$$\begin{Bmatrix} u_t \\ \hdashline u_o \end{Bmatrix} = \left[\begin{array}{c:c} I & 0 \\ \hdashline G_o & \phi_{oq} \end{array}\right] \begin{Bmatrix} u_t \\ \hdashline u_q \end{Bmatrix}$$

**Equation 3-25.**

The transformed matrix system is then

$$\left[\begin{array}{c:c} M'_{tt} & M_{tq} \\ \hdashline M_{qt} & M_{qq} \end{array}\right] \begin{Bmatrix} \ddot{u}_t \\ \hdashline \ddot{u}_q \end{Bmatrix} + \left[\begin{array}{c:c} K'_{tt} & 0 \\ \hdashline 0 & K_{qq} \end{array}\right] \begin{Bmatrix} u_t \\ \hdashline u_q \end{Bmatrix} = \begin{Bmatrix} P'_t \\ \hdashline P_q \end{Bmatrix}$$

**Equation 3-26.**

where $M'_{tt}$ is defined by equation

$$[M'_{aa}] = [M_{aa}] + [M_{ao}][G_o] + [G_o]^T[M_{oa} + M_{oo}G_o]$$

(for more information, see **Statics**) with $u_f$ replacing $u_a$, as follows:

$$[M_{tq}] = [M_{ot}^T][\phi_{oq}] + [G_o^T M_{oo}][\phi_{oq}]$$

**Equation 3-27.**

1 In these solution sequences $u_a$ includes $u_q$ and the previous equations for the $u_a$ set are replaced by the $u_t = u_b + u_c + u_r$ set.

$$[M_{qq}] = [\phi_{oq}^{t} M_{oo} \phi_{oq}]$$

**Equation 3-28.**

$$[K'_{tt}] = [K_{tt}] + [K_{to}][G_o]$$

**Equation 3-29.**

$$[K_{qq}] = [\phi_{oq}^{T} K_{oo} \phi_{oq}]$$

**Equation 3-30.**

$$\{P'_t\} = \{P_t\} + [G_o^T]\{P_o\}$$

**Equation 3-31.**

$$\{P_q\} = [\phi_{oq}^T]\{P_o\}$$

**Equation 3-32.**

### Error Analysis

Note that, as in Guyan Reduction, the exact static stiffness is retained. If accelerations exist, errors will occur when the modal displacements do not match the exact solution for $u^o_o$. Assume that the displacement error, $\epsilon_o$, is:

$$\{\varepsilon_o\} = [\phi_q]\{u_q\} - \{u_o^o\}$$

**Equation 3-33.**

The modal force error is

$$\{\delta_q\} = [\phi_{oq}^T K_{oo}]\{\varepsilon_o\} = [K_{qq}]\{u_q\} - [\phi_{oq}^T K_{oo}]\{u_o^o\}$$

**Equation 3-34.**

However, from equation

$$\begin{bmatrix} M'_{aa} & M'_{ao} \\ M'_{oa} & M'_{oo} \end{bmatrix} \begin{Bmatrix} \ddot{u}_a \\ \ddot{u}_o \end{Bmatrix} + \begin{bmatrix} K_{aa} & 0 \\ 0 & K_{oo} \end{bmatrix} \begin{Bmatrix} u_a \\ u_o \end{Bmatrix} = \begin{Bmatrix} P_a \\ P_o \end{Bmatrix}$$

(for more information, see **Statics**), we obtain

$$[K_{oo}]\{u_o^o\} = \{P_o\} - [M_{oo}]\{\ddot{u}_o^o\} - [M'_{ot}]\{\ddot{u}_t\}$$

**Equation 3-35.**

and from the lower half of **Eq. 3-26** and **Eq. 3-30** through **Eq. 3-32**, we obtain

$$[K_{qq}]\{u_q\} = [\phi_a^T]\{P_o\} - [\phi_a^T M_{oo}\phi_o]\{\ddot{u}_q\} - [\phi_o^T M'_{ot}]\{\ddot{u}_b\}$$

**Equation 3-36.**

Substituting **Eq. 3-35** and **Eq. 3-36** into **Eq. 3-34**, we obtain

$$\{\delta_q\} = [\phi_{oq}^T M_{oo}]\left\{\ddot{u}_o^o - [\phi_{oq}]\{\ddot{u}_q\}\right\}$$

**Equation 3-37.**

Therefore, we observe that the loading errors for dynamic reduction will be proportional to the size of the omitted masses, $M_{oo}$, and with the quality of the approximation

$$\ddot{u}_o \cong [\phi_q]\{\ddot{u}_q\}$$

**Equation 3-38.**

Relative to **Eq. 3-37**, the equivalent load error for Guyan reduction was

$$\{\delta_o\} = [M_{oo}]\{\ddot{u}_o^o\}$$

**Equation 3-39.**

Even when the mode shapes are only approximate, as in dynamic reduction, we conclude that the results (for the same $u_o$ set) will be improved from the Guyan results. However, note that when a $u_t$ set

is not used, the modes of the $u^o_o$ system have the same lower frequencies as the system modes and the approximation might not improve. In other words, the use of physical $u_t$ points in the dynamic reduction process combines the best parts of both Guyan and Rayleigh-Ritz reductions.

## Component Mode Synthesis

In Simcenter Nastran, the Component Mode Synthesis procedure is identical to the dynamic reduction process except that:

1. The approximate eigenvectors are replaced with results from the module which calculates modes of the superelement.

2. Boundary set points ($u_t$, $u_c$, or $u_r$) will be retained for connection to other superelements.

3. Special operations are performed for free-body motions.

Note that in Guyan Reduction the relative displacements, $u^o_o$, for the omitted point are only approximated in dynamics. In the dynamic reduction and modal synthesis methods, interior results are obtained by solving for its eigenvector displacements and added directly into the matrix solution system.

A special case, known as the inertia relief effect, occurs when the structural component is excited at low frequencies (below the first modal frequency). The effect is defined by

$$\{u_o\} \cong [G_o]\{u_a\} + K_{oo}^{-1}[\{P_o\} - [M'_{oa}]\{\ddot{u}_a\}]$$

(for more information, see **Statics**) when $\ddot{u}_a$ is a low frequency excitation, having little effect on the vibration modes, resulting in a relative quasi-static displacement, $u^o_o$, where

$$\{u^o_o\} = -[K_{oo}]^{-1}[M'_{oa}]\{\ddot{u}_a\}$$

**Equation 3-40.**

The displacement vector is called the inertial relief shape when $\ddot{u}_a$ is a rigid body acceleration vector. It may cause a significant redistribution of the internal forces in the structure.

In the solution of a single structural component, the inertia relief effects may be obtained after the solution, $u_a$, is obtained. However, for multiple superelements connected at the $u_t$ boundary points, the load distributions on the boundary will be affected by these internal loads. Note that the errors defined in **Eq. 3-41** and **Eq. 3-42** will be affected since the eigenvectors may only approximate the vectors defined in **Eq. 3-43**.

$$[u_o] = [G_o]\{u_a\}$$

**Equation 3-41.**

$$\{\delta_q\} = [\phi_{oq}^T K_{oo}]\{\varepsilon_o\} = [K_{qq}]\{u_q\} - [\phi_{oq}^T K_{oo}]\{u_o^o\}$$

**Equation 3-42.**

$$\{\delta_q\} = [\phi_{oq}^T M_{oo}]\left\{\ddot{u}_o^o - [\phi_{oq}]\{\ddot{u}_q\}\right\}$$

**Equation 3-43.**

For more information, see **Guyan Reduction** and **Dynamic Reduction**.

The solution used in Simcenter Nastran provides for inertia relief modes by adding six vectors to the approximation set $\varphi_o$. These vectors are obtained from **Eq. 3-40** and use the equation:

$$[\phi_o^{ir}] = [K_{oo}]^{-1}[M'_{oa}][D_a]$$

**Equation 3-44.**

where $D_a$ is a matrix of six rigid body motions defined by the grid point geometry. Each column of the matrix $\varphi^{ir}{}_o$ is scaled to a reasonable value and added to the $\varphi_o$ matrix, and a corresponding $u_q$ displacement is also included.

## 3.3 Dynamics of Rotating Structures

The following sections provide a description of the dynamic loads on a rotating structure defined in the rotating coordinate system. While the stiffness and damping terms measured in a rotating system are the same as those measured in a stationary system, the terms due to inertial resistance are dependent upon the rotation of the structure. These inertial dependent terms need to be determined and added to the total impedance of the structure before static and dynamic analyses in a rotating system can be performed.

### Technical Approach

A rotating structure is usually modeled under the assumption that the coordinate system used in the description of the structure will rotate at a constant rate about a fixed axis. Displacements of the structure and forces applied to the structure are measured in this rotating system. Because the rotating system is accelerating relative to a stationary inertial system, the inertial or mass dependent impedance

cannot be directly calculated in the rotating system. The inertial impedance in the rotating system must first be determined in the stationary reference frame, and then transformed to the rotating system.

The impedance due to stiffness and damping is unaffected by the choice of reference frames. The stiffness and damping impedances in a rotating reference system are identical to those in a stationary reference system.

To determine the inertial impedance in a rotating system requires the development of general transformation techniques between stationary and rotating systems. The following section describe the general transformation of a vector between stationary and rotating system coordinates. The second section discusses inertial forces defined in a rotating coordinate system. The third section applies these results to develop the impedance of a rotating structure. The fourth section describes the DMAP procedure to add the gyroscopic terms to the structural matrices for analysis in the rotating system.

## Vector Transformation from Stationary to Rotating Coordinates

It is important to keep in mind that the essential features of a vector are magnitude and direction, but not location. A vector defined in one coordinate system can just as easily be defined in a different coordinate system through the use of a coordinate transformation matrix. Also keep in mind that vectors may be time dependent in both their magnitude and direction and that coordinate systems may rotate with respect to each other resulting in a time-dependent transformation matrix.

The remainder of this discussion concerns transforming a vector defined in stationary system coordinates to rotating system coordinates. The choice of which system is rotating and which system is stationary is completely arbitrary at this point. No physical significance (such as position, velocity, etc.) is assigned to the vector; the only important qualities are its magnitude and direction.

The general transformation of a time-dependent vector from a stationary coordinate system to a rotating coordinate system at the same origin may be written as

$$\{v(t)_r\} = [A(t)]\{v(t)_s\}$$

**Equation 3-45.**

where:

$\{v(t)_r\}$ = rotating coordinate system definition of time-dependent vector.
$[A(t)]$ = time-dependent transformation matrix from rotating to stationary system.
$\{v(t)_s\}$ = stationary coordinate system definition of time-dependent vector.

The above transformation is valid for any vector, real or complex.

The fundamentals of this transformation are demonstrated for the two systems shown below.

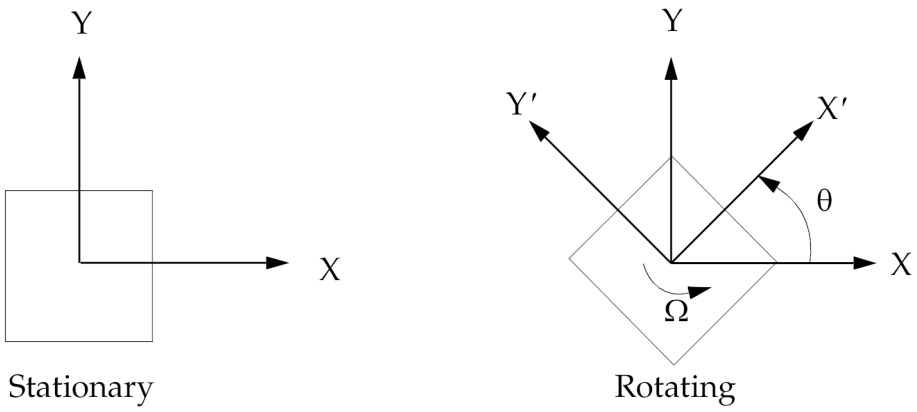



**Figure 3-1. Stationary and Rotating Coordinate Systems**

Assume the rotating system spins about the z-axis at a constant rate, Ω, relative to the stationary system. $x(t)_s$, $y(t)_s$, and $z(t)_s$ are the time-dependent components of the vector in the stationary system. $x(t)_r$, $y(t)_r$, and $z(t)_r$ are the time-dependent components of the vector in the rotating system.

For this example the transformation of the coordinates from the stationary to rotating system is

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-46.**

where $\theta = \Omega t$. Equation Eq. 3-46 is simply an expanded form of Eq. 3-45.

Using complex identities for cosθ and sinθ,

$$\cos\theta = \frac{1}{2}(e^{i\theta} + e^{-i\theta})$$

**Equation 3-47.**

$$\sin\theta = -\frac{i}{2}(e^{i\theta} - e^{i\theta})$$

**Equation 3-48.**

Equation Eq. 3-46 is written as

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix} = \left( \frac{e^{i\theta}}{2} \begin{bmatrix} 1 & -i & 0 \\ i & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \frac{e^{-i\theta}}{2} \begin{bmatrix} 1 & i & 0 \\ -i & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-49.**

or

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix} = (e^{i\theta}[T_1] + e^{-i\theta}[T_1]^* + [T_0]) \begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-50.**

where:

$$[T_1] = \frac{1}{2} \begin{bmatrix} 1 & -i & 0 \\ i & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

**Equation 3-51.**

and the conjugate matrix is

$$[T_1]^* = \frac{1}{2} \begin{bmatrix} 1 & i & 0 \\ -i & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

**Equation 3-52.**

$$[T_0] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

**Equation 3-53.**

Additionally, we can write the vectors in terms of complex components

$$\begin{Bmatrix} x(t) \\ y(t) \\ z(t) \end{Bmatrix} = \begin{Bmatrix} \bar{x}(t)e^{i\psi_x(t)} \\ \bar{y}(t)e^{i\psi_y(t)} \\ \bar{z}(t)e^{i\psi_z(t)} \end{Bmatrix}$$

**Equation 3-54.**

where:

$\bar{x}(t)$, $\bar{y}(t)$, and $\bar{z}(t)$ are time-dependent magnitudes and

$\Psi_x(t)$, $\Psi_y(t)$, and $\Psi_z(t)$ are time-dependent phase functions.

Substituting the above definition into **Eq. 3-50** produces the complex transformation

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix} = [T_1]\begin{Bmatrix} \bar{x}(t)_s e^{i\psi_x(t)+\Omega t} \\ \bar{y}(t)_s e^{i\psi_y(t)+\Omega t} \\ \bar{z}(t)_s e^{i\psi_z(t)+\Omega t} \end{Bmatrix} + [T_1]^*\begin{Bmatrix} \bar{x}(t)_s e^{i\psi_x(t)-\Omega t} \\ \bar{y}(t)_s e^{i\psi_y(t)-\Omega t} \\ \bar{z}(t)_s e^{i\psi_z(t)-\Omega t} \end{Bmatrix}$$
$$+ [T_0]\begin{Bmatrix} \bar{x}(t)_s e^{i\psi_x(t)} \\ \bar{y}(t)_s e^{i\psi_y(t)} \\ \bar{z}(t)_s e^{i\psi_z(t)} \end{Bmatrix}$$

**Equation 3-55.**

Inspection of the above equation shows that a vector defined in stationary system coordinates can be defined as three vectors in rotating system coordinates. One vector will have the phasing of its motion advanced by $\Omega t$ with respect to the stationary system definition, the second will have its phasing delayed by $\Omega t$, and the third will have its phase unaffected.

**Progressive and Regressive Vectors**

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix} = \begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}_+ + \begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}_- + \begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}_0$$

**Equation 3-56.**

where

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}_+ = e^{i\Omega t}[T_1]\begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-57.**

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}_- = e^{-i\Omega t}[T_1]^*\begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-58.**

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}_0 = [T_0]\begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-59.**

It is convenient to specify the vector defined by **Eq. 3-57** as the progressive vector in rotating system coordinates, the vector defined by **Eq. 3-58** as the regressive vector in rotating system coordinates, and the vector defined by **Eq. 3-59** as the collective vector in rotating system coordinates. Care must be taken when discussing progressive and regressive vectors to establish whether the definition is in terms of the rotating or nonrotating system. By reformulating **Eq. 3-46** as a transformation from rotating to nonrotating coordinates, one can define progressive and regressive vectors in terms of the nonrotating system.

$$\begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix} = (e^{i\theta}[T_1]^* + e^{-i\theta}[T_1] + [T_0]) \begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}$$

**Equation 3-60.**

In this discussion, progressive and regressive are defined relative to the rotating system.

## Inertial Forces in a Rotating Coordinate System

The calculation of inertial forces in a rotating coordinate system requires the introduction of an inertial reference system. An inertial reference system is defined as a coordinate system in which the motion of a particle with respect to these coordinates adhere to Newton's laws of motion. Implied in these laws is that acceleration of a particle is measured in a nonaccelerating and nonrotating coordinate system. In the above discussions the stationary coordinate system can be considered an inertial coordinate system.

Inertial force is defined as the resistance of a particle to a change in its state of rest or straight-line motion. It is expressed as Newton's well-known equation

$$F = ma$$

**Equation 3-61.**

or

$$F - ma = 0$$

**Equation 3-62.**

where:

F = applied force
$m$ = mass of the particle
$a$ = acceleration of the particle

The term $-ma$ is referred to as the inertial force. Inertial force represents the resistance of a particle to a change in its velocity vector. It is equal in magnitude to the applied force but acts in the opposite direction.

To determine the inertial forces in a rotating coordinate system, it is necessary to transform the vector representing the acceleration of a particle in the inertial coordinate system to the rotating coordinate system. This requires: 1) determining the acceleration vector in the inertial coordinate system, then 2)

transforming the acceleration vector from the inertial system coordinates to the rotating system coordinates. This results in the inertial coordinate system acceleration vector being defined in terms of rotating system vectors.

Eq. 3-63 defines a vector transformation from rotating system coordinates to stationary system coordinates.

$$\begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix} = (e^{i\theta}[T_1]^* + e^{-i\theta}[T_1] + [T_0]) \begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix}$$

**Equation 3-63.**

For more information, see **Vector Transformation from Stationary to Rotating Coordinates**.

Let a vector ρ*s* be defined as the position vector for a particle in the stationary system coordinates, and ρ*r* be defined as the position vector in the rotating system coordinates.

$$\{\rho(t)_s\} = (e^{i\theta}[T_1]^* + e^{-i\theta}[T_1] + [T_0])\{\rho(t)_r\}$$

**Equation 3-64.**

Taking the second derivative with respect to time produces

$$\begin{aligned} \frac{d^2\{\rho(t)_s\}}{dt^2} &= (e^{i\theta}[T_1]^* + e^{-i\theta}[T_1] + [T_0])\frac{d^2\{\rho(t)_r\}}{dt^2} \\ &+ i2\Omega(e^{i\theta}[T_1]^* - e^{i\theta}[T_1])\frac{d\{\rho(t)_r\}}{dt} \\ &- \Omega^2(e^{i\theta}[T_1]^* + e^{-i\theta}[T_1])\{\rho(t)_r\} \end{aligned}$$

**Equation 3-65.**

The equation above defines the acceleration vector in the stationary or inertial coordinate system.

Eq. 3-66 represents the vector transformation from stationary system coordinates to rotating system coordinates.

$$\begin{Bmatrix} x(t)_r \\ y(t)_r \\ z(t)_r \end{Bmatrix} = (e^{i\theta}[T_1] + e^{-i\theta}[T_1]^* + [T_0]) \begin{Bmatrix} x(t)_s \\ y(t)_s \\ z(t)_s \end{Bmatrix}$$

**Equation 3-66.**

For more information, see **Vector Transformation from Stationary to Rotating Coordinates**.

Applying this transformation to the stationary system acceleration vector given by **Eq. 3-65** results in the inertial system acceleration defined in rotating system coordinates, as follows:

$$\left(\frac{d^2\{\rho(t)_s\}}{dt^2}\right)_r =$$

$$(e^{i\theta}[T_1] + e^{-i\theta}[T_1]^* + [T_0])(e^{i\theta}[T_1]^* + e^{-i\theta}[T_1] + [T_0])\frac{d^2\{\rho(t)_r\}}{dt^2}$$

$$+ i2\Omega(e^{i\theta}[T_1] + e^{-i\theta}[T_1]^* + [T_0])(e^{i\theta}[T_1]^* - e^{-i\theta}[T_1])\frac{d\{\rho(t)_r\}}{dt}$$

$$- \Omega^2(e^{i\theta}[T_1] + e^{-i\theta}[T_1]^* + [T_0])(e^{i\theta}[T_1]^* + e^{-i\theta}[T_1])\{\rho(t)_r\}$$

**Equation 3-67.**

Using the identities

$$[T_1][T_1]^* = [T_1]^*[T_1] = [0]$$

**Equation 3-68.**

$$[T_1][T_0] = [T_1]^*[T_0] = 0$$

**Equation 3-69.**

$$[T_1][T_1] = [T_1]$$

**Equation 3-70.**

$$[T_1]^*[T_1]^* = [T_1]^*$$

**Equation 3-71.**

$$[T_0][T_0] = [T_0]$$

**Equation 3-72.**

$$[T_1]^* + [T_1] = 2\mathrm{Re}([T_1])$$

**Equation 3-73.**

$$[T_1]^* - [T_1] = -i2\mathrm{Im}([T_1]) = i2\mathrm{Re}(i[T_1])$$

**Equation 3-74.**

Eq. 3-67 reduces to

$$\left(\frac{d^2\{\rho(t)_s\}}{dt^2}\right)_r = \frac{d^2\{\rho(t)_r\}}{dt^2} - 4\Omega\mathrm{Re}(i[T_1])\frac{d\{\rho(t)_r\}}{dt} - 2\Omega^2\mathrm{Re}([T_1])\{\rho(t)_r\}$$

**Equation 3-75.**

The term $4\Omega\mathrm{Re}(i[T_1])$ is commonly referred to as the Coriolis acceleration and the term $-2\Omega^2\mathrm{Re}(i[T_1])$ is referred to as the centripetal acceleration. This equation provides a means of representing inertial or stationary system acceleration in terms of rotating system coordinates.

Using Eq. 3-75, the inertial force on a particle whose position is measured relative to a rotating coordinate system is written as

$$\begin{aligned}\{F(t)_{inertial}\}_r &= -[M]\left(\frac{d^2\{\rho(t)_s\}}{dt^2}\right)_r \\ &= -[M]\left(\frac{d^2\{\rho(t)_r\}}{dt^2} - 4\Omega\mathrm{Re}(i[T_1])\frac{d\{\rho(t)_r\}}{dt} - \Omega^2 2\mathrm{Re}([T_1])\{\rho(t)_r\}\right)\end{aligned}$$

**Equation 3-76.**

This equation can be written as

$$\{F(t)_{inertial}\}_r = -\left([M]\left(\frac{d^2\{\rho(t)_r\}}{dt^2}\right) - \Omega[B^c]\left(\frac{d\{\rho(t)_r\}}{dt}\right) - \Omega^2[K^c]\{\rho(t)_r\}\right)$$

**Equation 3-77.**

where (for rotation about the z-axis)

$$[M] = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & m \end{bmatrix}$$

**Equation 3-78.**

$$[B^c] = \begin{bmatrix} 0 & 2m & 0 \\ -2m & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

**Equation 3-79.**

$$[K^c] = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

**Equation 3-80.**

## Structural Impedance of a Rotating Structure

In **Eq. 3-81**, the inertial forces are dependent on the time-varying position of a mass particle.

$$\{F(t)_{inertial}\}_r = -\left([M]\left(\frac{d^2\{\rho(t)_r\}}{dt^2}\right) - \Omega[B^c]\left(\frac{d\{\rho(t)_r\}}{dt}\right) - \Omega^2[K^c]\{\rho(t)_r\}\right)$$

**Equation 3-81.**

For more information, see **Inertial Forces in a Rotating Coordinate System**.

This position vector can be written in terms of an initial component and a time-varying component.

$$\{\rho(t)\}_r = \{\rho_r\} + \{u(t)_r\}$$

**Equation 3-82.**

Eq. 3-81 can be rewritten using the above equation as

$$\{F(t)_{inertial}\}_r = -\left([M]\left(\frac{d^2\{u(t)_r\}}{dt^2}\right) - \Omega[B^c]\left(\frac{d\{u(t)_r\}}{dt}\right) - \Omega^2[K^c](\{\rho_r\} + \{u(t)_r\})\right)$$

**Equation 3-83.**

Inertial forces due to the mass are resisted by structural stiffness and damping.

$$[B]\left(\frac{d\{u(t)_r\}}{dt}\right) + [K]\{u(t)_r\} = \{F(t)_{inertial}\}_r$$

**Equation 3-84.**

Substituting Eq. 3-83 in the above equation and collecting all time-dependent terms on the left-hand side results in

$$[M]\left(\frac{d^2\{u(t)_r\}}{dt^2}\right) + ([B] - \Omega[B^c])\left(\frac{d\{u(t)_r\}}{dt}\right) + ([K] - \Omega^2[K^c])\{u(t)_r\} = \Omega^2[K^c]\{\rho_r\}$$

**Equation 3-85.**

The above equation is the equation of motion for a particle in a rotating elastic structure with no externally applied forces. The loading on the right-hand side of Eq. 3-85 is often called the centripetal force. The centripetal force is always present on a rotating structure and produces an additional stiffness-like term. This additional term can be determined from consideration of the standard static equilibrium equation for a single element.

$$Ku = P$$

**Equation 3-86.**

where:

*K* is the element stiffness in global coordinates (coordinates used to define motion).

*u* is the displacement in global coordinates.

*P* is the load in global coordinates.

The deformation, *u*, produces element reaction forces, *F*, which balance the applied load, *P*. The stiffness, *K*, can be viewed as the change in the element reaction forces with respect to a change in displacement.

$$K = \frac{dF}{du} = \frac{d(TF_e)}{du}$$

**Equation 3-87.**

$$K = T\left(\frac{dF_e}{du}\right) + F_e\left(\frac{dT}{du}\right)$$

**Equation 3-88.**

$$K = T\left(\frac{du_e}{du}\right)\left(\frac{dF_e}{du_e}\right) + F_e\left(\frac{dT}{du}\right)$$

**Equation 3-89.**

where:

*F* is the internal element force in global coordinates.

*T* is a transformation from element to global coordinates.

$F_e$ is the internal element force in element coordinates.

$u_e$ is a displacement in element coordinates.

For a linear material and a loading which is not dependent on displacement, the term $dF_e/du_e$ is constant, the element forces are linearly related to the element displacements. The terms $T$ and $du_e/du$ are dependent only on the orientation of the element coordinate system with respect to the global coordinate system. For small displacement problems, $T$ and $du_e/du$ are assumed constant and are calculated from the undeformed geometry. The three terms ($T$, $du_e/du$, and $dF_e/du_e$) are used to calculated the linear element stiffness matrix in global coordinates. The element stiffness matrices are assembled to create the complete linear stiffness matrix for the structure.

The second part of the stiffness calculation, $F_e(dT/du)$, is usually ignored in standard linear analysis. It arises from small changes in the element orientation with respect to the applied load. For example, consider a bar under axial and transverse loading.

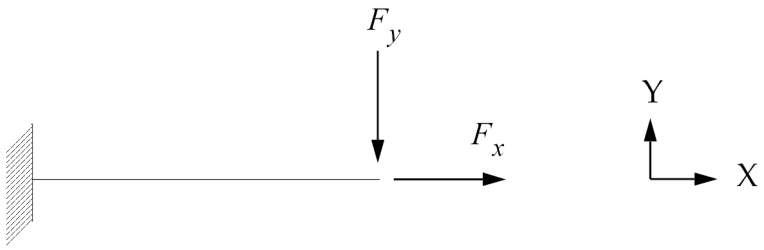



**Figure 3-2. Cantilevered Bar With Axial and Transverse Loading**

In a standard linear analysis, the $y$-directed load is reacted by the bar bending and shear forces. The $x$-directed load is reacted by the bar axial forces. The calculated $y$ displacement is independent of the $x$-directed load. Likewise, the calculated $x$ displacement is independent of the $y$-directed load. On closer inspection, one would find that as the free end of the bar rotates with respect to the applied loads, a component of the bar axial force reacts against the $y$-directed load. Likewise, a component of the bar bending and shear react to the $x$-directed load.

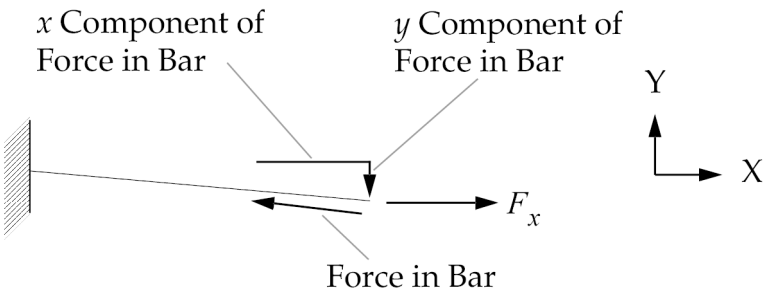



**Figure 3-3. Bar Forces Due to Axial Loading**

This results in a stiffness coupling which is not taken into account by standard linear analysis. This effect is due to the $F_e(dT/du)$ term in **Eq. 3-89** and is referred to as the *differential* stiffness term. Its calculation requires knowledge of the element force $F_e$ and the transformation matrix $T$. The total stiffness matrix, often referred to as the tangent stiffness matrix, is the sum of the linear and differential stiffnesses.

Eq. 3-85 shows that a centripetal load is always present in a rotating structure. This constant loading results in a differential stiffness term which must be included for accurate analysis of rotating structures. The differential stiffness is added to the stiffness terms on the left-hand side of Eq. 3-85. The complete equation of motion for a rotating structure, neglecting the centripetal loading is

$$[M]\left(\frac{d^2\{\Delta\rho(t)_r\}}{dt^2}\right) + ([B] - \Omega[B^c])\left(\frac{d\{\Delta\rho(t)_r\}}{dt}\right)$$

$$+ ([K] + \Omega^2[K^d] - \Omega^2[K^c])\{\Delta\rho(t)_r\} = 0$$

**Equation 3-90.**

The above equation can be used to determine the motion of a rotating structure about the deformation due to the centripetal loading.

## DMAP Procedure to Add Rotation-Dependent Terms to Simcenter Nastran Dynamic Analyses

These alters may be used for the following problems:

- To determine critical speeds of rotating machinery, such as shafts and turbines.
- To predict effects of unbalanced forces at selected rotational speeds.
- To simulate transient behavior relative to the steady-state rotation.

> Note:
>
> In rotor dynamics shafts are often modeled with BEAM elements and the axisymmetric disks as rigid bodies (CONM2s). Due to the polar movement of inertia of these disks, there is a gyroscopic effect on the critical speeds. Unfortunately, the DMAP alters discussed below do not allow for this gyroscopic effect.

The procedure to add rotation-dependent terms to a Simcenter Nastran analysis is done in two steps: 1) Calculate the rotation-dependent terms and add them to the structural damping and stiffness matrices, 2) Restart analysis with these modified matrices.

The first step is performed in SOL 101 (linear static analysis) or SOL 114 (cyclic linear static analysis). A centripetal load is specified (RFORCE entry in the bulk data) as the only loading (one subcase only). A DMAP alter uses this information to calculate all rotation-dependent terms, adds them to the structural damping and stiffness matrices, and stores the modified matrices on the database.

The second step is performed as a restart in any structured linear solution sequence. The modified matrices are reduced and the analysis is performed. The user may change the SPCs, MPCs, and loading, but not the actual structure.

Three alters are available in the SSSALTER library; they are:

| | |
|---|---|
| **segyroa.vxx** | Adds gyroscopic and differential stiffness terms to noncyclic symmetry superelement models. The alter is added to SOL 101. Restarts to any structured superelement solution sequence are allowed. |
| **cygroa.vxx** | Adds gyroscopic and differential stiffness terms to cyclic symmetric superelement models. The alter is added to SOL 114. Restarts to any structured cyclic symmetry solution sequence is allowed. |
| **nlgyroa.vxx** | Adds gyroscopic terms to the normal modes calculation in SOL 106 (nonlinear statics). No restart is required, normal modes are calculated in SOL 106 if PARAM,NMLOOP is included in the input file. |

A readme file (readme.gyro) and example problems are also included in the SSSALTER library.

## 3.4 Overview of Aeroelastic Analysis

Simcenter Nastran provides efficient solutions to the problems of aeroelasticity, which is a branch of applied mechanics that deals with the interaction of aerodynamic, inertial, and structural forces. It is important in the design of airplanes, helicopters, missiles, suspension bridges, and even tall chimneys and power lines. Aeroservoelasticity is a variation in which the interaction of automatic controls requires additional consideration.

The primary concerns of aeroelasticity include flying qualities (stability and control), flutter, and structural loads arising from maneuvers and atmospheric turbulence. Methods of aeroelastic analysis differ according to the time dependence of the inertial and aerodynamic forces that are involved. For the analysis of flying qualities and maneuvering loads wherein the aerodynamic loads vary relatively slowly, quasi-static methods are applicable. The remaining problems are dynamic, and methods of analysis differ according to whether the time dependence is arbitrary (transient or random) or simply oscillatory in the steady state.

Simcenter Nastran considers three classes of problems in aeroelasticity:

- Static Aeroelastic Response
- Aerodynamic Flutter
- Dynamic Aeroelastic Response

Each is described in the text that follows. In addition, information is provided on the ability to include aeroelastic responses within the Simcenter Nastran optimization capability, the aerodynamic methods available, and on special features that make the Simcenter Nastran aeroelastic capability unique.

## Static Aeroelastic Response

For the analyses of flying qualities and maneuvering loads, the assumption of quasi-steady motion is valid, i.e., the dynamics of the flexible structure are neglected and quasi-static methods are applicable. By assuming linear behavior of the aerodynamic, inertial and structural forces during the motion, the equations of equilibrium in quasi-steady flight are solved in closed algebraic (matrix) form. Linear and/or surface splines may be used to connect the aerodynamic and structural grid points.

The static aeroelastic analysis solves for the trim condition in a prescribed maneuver. The formulation of the equilibrium equations provides the aerodynamic stability and control derivatives as an integral part of the trim process. The external flight loads and the corresponding internal loads and stresses on the finite elements, are available as postprocessing operations on the trim solution.

Static aeroelastic divergence is a non-oscillatory instability condition that can occur when the aerodynamic forces overpower the stiffness of the structure. For free-flying vehicles, this phenomenon is typically not of concern, but it can be critical in the structural design of restrained wind tunnel models. Static aeroelastic divergence analysis can be performed as an option within the overall static aeroelastic capability.

## Aerodynamic Flutter

Flutter is the oscillatory aeroelastic instability that occurs at some airspeed at which energy extracted from the airstream during a period of oscillation is exactly dissipated by the hysteretic damping of the structure. The motion is divergent in a range of speeds above the flutter speed. Flutter analysis utilizes complex eigenvalue analysis to determine the combination of airspeed and frequency for which the neutrally damped motion is sustained.

Three methods of flutter analysis are provided: the American flutter method (called the K-method in Simcenter Nastran), an efficient K-method (called the KE-method) for rapid flutter evaluations, and the British flutter method (called the PK-method) for more realistic representation of the unsteady aerodynamic loads as frequency dependent stiffness and damping terms. The complex eigenvalue analysis is specified by the user with the K-method, and the QR-transformation method is used with the KE- and PK-methods. Again, linear and/or surface splines may be used to connect the aerodynamic and structural grid points.

## Dynamic Aeroelastic Response

The dynamic aeroelastic response problem is one of determining the response of the aircraft to time-varying excitations. Atmospheric turbulence is the primary example of this type of excitation, but store ejection loads and landing gear impact can also have an aeroelastic component. Methods of generalized harmonic (Fourier) analysis are applied to the linear system to obtain the response to the excitation. The turbulence model may be regarded either as a stationary random loading or as a discrete gust.

The gust analysis capability computes response to random atmospheric turbulence and discrete one-dimensional gust fields. The random response parameters calculated are the power spectral density, root mean square response, and mean frequency of zero-crossings. The response to the discrete gust is calculated by direct and inverse Fourier transform methods since the oscillatory aerodynamics are only

known in the frequency domain. Time histories of response quantities are the output in the discrete case.

## Aeroelastic Optimization

The integration of the aeroelastic analysis capability contained in Simcenter Nastran with a design sensitivity and optimization capability provides a design tool for the aeroelastician. Sensitivity analysis entails the determination of the effects that changes in structural properties have on response quantities, such as displacements or stresses. Optimization utilizes information on the response values and their sensitivities to automatically determine a design that meets a design objective, such as limits on stresses, deformations, or flutter characteristics. The static aeroelastic and flutter analyses are available within the Simcenter Nastran optimization capability and can be used in multidisciplinary fashion along with standard static analysis, normal modes analysis, and dynamic response analysis. Dynamic aeroelastic response is not available for optimization. Aeroelastic responses available for the sensitivity and optimization include stability derivatives and trim settings from the static aeroelastic analysis and flutter damping level from the flutter analysis.

## Aerodynamic Methods

Four oscillatory aerodynamic theories are available for flutter analysis. There is one subsonic method, the Doublet-Lattice Method with body interference, and two supersonic methods: the Mach Box Method and the Piston Theory. The forth method is rudimentary Strip Theory, which can be applied at any Mach number.

## Special Features

Aeroelastic analysis in Simcenter Nastran provides several advances in the state of-the art. The fundamental problem of interconnecting the aerodynamic and structural grids in the finite element models is solved by a closed form solution to an infinite plate over multiple supports. This two-dimensional interpolation was developed in addition to a generalization of the one-dimensional spline for a bending, twisting beam (elastic axis) on multiple collinear supports.

The implementation of the lined-up British flutter method, called the PK-method in Simcenter Nastran, was the first attempt to popularize the British approach to flutter analysis in the United States. This, along with the transfer function capability for control systems, makes analysis of aeroservoelastic problems a routine matter. The transfer function representation is for second order systems: a single output from multiple inputs.

The analysis of response to a discrete gust requires Fourier transform methods, because the aerodynamics assume harmonic motion. First, a direct transform of the discrete gust profile is necessary to place the forcing function in the frequency domain. Second, an inverse Fourier transform of the forced frequency response is necessary to obtain the transient response of vehicles to the gust. Both the direct and inverse Fourier transform calculations have been implemented in Simcenter Nastran.

The quasi-steady equations of motion of a free-flying vehicle require consideration of the inertial relief effects. For the unrestrained vehicle, the inertial effects are contained in the basic stability and control derivatives. However, derivatives that are independent of weight distribution are desirable for use in

flight simulators, and are obtained by assuming the aircraft to be restrained in some reference support configuration. The equations of motion using restrained aeroelastic derivatives require not only additional inertial derivatives, but also the rotations of the mean axes relative to the support for each aerodynamic variable (e.g., angle of attack, elevator rotation, pitch rate). These additional aeroelastic coefficients permit the support to be unloaded and angular momentum to be conserved. Simcenter Nastran provides both restrained and unrestrained aeroelastic derivatives, and 01in the restrained case, the inertial derivatives and mean axis rotations.

Control systems can also be included in dynamic response analysis using Simcenter Nastran transfer functions. With this feature, the aeroservoelastic interactions of ride comfort and load alleviation systems can be investigated.

The Simcenter Nastran procedure has general capabilities that are beyond those listed here, and their application to aeroelastic design is limited only by the analyst's ingenuity. For example, aerothermoelasticity considers the effects of thermal stresses on structural stiffness and the subsequent aeroelastic interactions. Simcenter Nastran provides a capability for nonlinear static analysis that includes temperature loadings. Aerothermoelastic problems of high speed flight can therefore be addressed by restarting any of the three aeroelastic analyses from a database created by the nonlinear analysis that has generated the stiffness of the heated structure.

# 4. Boundary Conditions and Coupling to Other Disciplines

## 4.1 Overview of Boundary Conditions and Coupling to Other Disciplines

This chapter extends the discussion of basic dynamic loads and boundary conditions given in the *Simcenter Nastran Basic Dynamic Analysis User'sGuide*. This chapter describes some of the more advanced methods available in Simcenter Nastran for dealing with these problems.

Although applying dynamic loads to a structure is a relatively straightforward task when using the finite element method, the process of specifying a moving boundary requires more work. Examples of enforced motions of a structural boundary include earthquakes on a building, automobile suspensions, and any vibration affecting a light, flexible structure connected to a massive, moving body. This chapter describes approximate and exact solution techniques for this type of problem.

Another type of boundary condition is a fluid or gas coupled with the surface of a structure. If these fluids are assumed to have small motions, they generate linear matrices and allow the use of conventional solution methods. This chapter describes two of three approaches to the coupled fluid-boundary problem, each with a unique set of capabilities. The Overview of acoustic and vibro-acoustic analyses chapter in the *Simcenter Nastran Acoustics Simcenter Nastran User's Guide* describes the third approach.

## 4.2 Enforced Motion with Loads

Two distinct methods are used in Simcenter Nastran for dynamic enforced motions: the Large Mass/Spring approach, and the Inertial Loads method. These are covered in this section.

### Large Mass/Spring Method

The Large Mass/Spring approach is actually a modeling technique in which the user places an element with a large mass or stiffness at the points of known acceleration or displacement. In effect, this large element acts as a constraint on the connected point. The user then supplies a corresponding large force via RLOADi or TLOADi inputs to produce the desired motion. If the added element is sufficiently stiff or heavy, the reaction forces from the actual structure will not affect the input motions. The actual input data and controls are described in Overview of Enforced Motion in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*. Refer to them for modeling details. More advanced applications of both large mass and large spring techniques are discussed below.

Using a large mass to enforce boundary motions is a standard practice in Simcenter Nastran. It works well on a simply supported structure when a single grid point is excited by a well defined acceleration input. A typical example is the earthquake analysis of a tall building where a single base input is assumed. However the method may be abused when other types of boundary conditions are attempted.

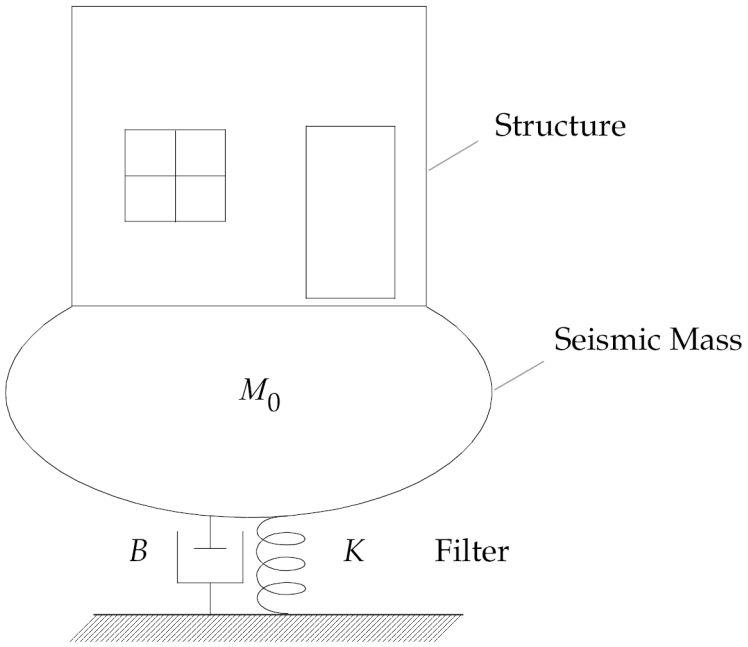
Some additional considerations for the advanced analyst are listed below:

1. If enforced motion is applied to a redundant set of boundary points, a danger exists that the large masses (or springs) may create fictitious forces and stresses in the structure. This occurs when the enforced displacements inputs are not precisely synchronized. Also, in a modal formulation, the extra masses will cause fictitious low frequency modes to occur. The recommended procedure is to connect the redundant points with RBE (rigid) elements to prevent their independent motions.

2. Small errors in the loading history may cause large errors in the structural response. When using enforced accelerations in a transient solution, a small bias in the inputs (from instrumentation or processing) may cause a large spurious drift where the structure displaces a large amount as a rigid body. Solutions to remove the drift are as follows:

   - Supply a corrective load function, obtained from an initial run, to cancel the measured drift;
   - Add dampers and springs in parallel with mass and tuned to filter the input signal; or
   - Use the modal method and drop the zero-frequency modes. (See item 4 below.)

   An example of a mechanical filter is shown in the sketch below. A large mass, $M_0$, is attached to the base of a structure to allow enforced accelerations. For control of the spurious displacements and velocities, attach a scalar spring, $K$, and damper element, $B$, between the mass and ground for each direction. If the first modal frequency is $f_1$ then set $K = 0.4\ Mf^2_1$ for a filter frequency approximately one tenth of the first vibration mode frequency, $f_1$. Set the damping coefficient to a value near critical to eliminate spurious oscillations.




3. In the nonlinear transient solutions, large springs can affect the error tests and convergence logic. The internally calculated error ratios are dimensionless numbers obtained by dividing the errors by

an average force or total energy. The forces and energy created by the large mass/spring approach will dominate these values, resulting in underestimates of the errors and false convergence. The solution is to decrease the error allowables on the TSTEPNL Bulk Data inputs.

4. Using these methods in a modal formulation requires some attention. For output of total displacements, the user should retain the zero or low frequency modes that the large masses produce (i.e., set the parameter *LFREQ* = – .01). Note that if the low frequency modes are dropped from the dynamic solution, the output will be the correct relative motion.
   Large springs for enforced displacements are not recommended for the modal formulations. They should generate high frequency modes that are usually missing from the system. The resulting dynamic solution is not valid since the large springs are not included in the modal stiffness matrix.

5. Numerical conditioning of the matrix solution may be affected by the method used to connect the large mass or spring. Numerical roundoff of the results may occur. MPCs, RBEi's, and ASET operations all use a matrix elimination procedure that may couple many degrees-of-freedom. If a large mass or spring is not retained in the solution set, its matrix coefficient will be distributed to other solution points. Then, matrix conditioning for decomposition operations becomes worse, when the large terms dominate the significant finite element coupling terms). On the other hand, if the degrees-of-freedom with the large terms remain in the solution set, they remain on the diagonal of the matrix and the matrix decomposition is unaffected.

In summary, the large mass method is recommended for cases with known accelerations at a single point. It works well with the modal formulation, providing good stress and forces near the mass, and is easy to understand and use. In many cases, the structure is actually excited by the motions of a large, massive, base (for instance, the geological strata) which can actually be used as a value for the mass.

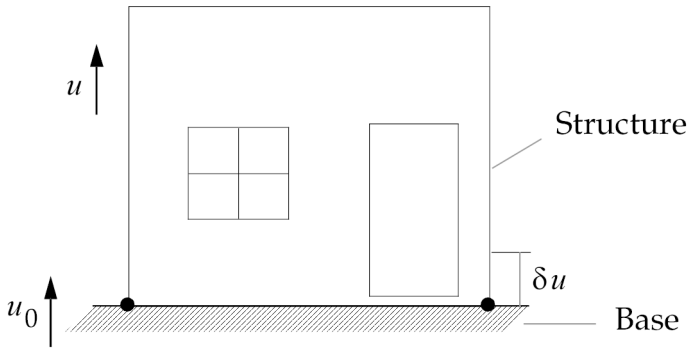
The large spring method is recommended for cases in which displacements are known at one point and a direct formulation is used. The primary advantage is its simplicity. However, the inertial loads approach described below, is more general and has better reliability.

## The Inertial Loads Approach

By changing to a moving coordinate system, the reference accelerations may be converted to inertial loads. If the dynamic problem involves a free body connected to a single rigid base, the GRAV and dynamic load inputs may be used to supply forces proportional to the base acceleration. The solution will be defined by the displacements measured relative to the fixed base.

### Approach

The basic approach is shown in the sketch below:




We assume that the motions of the rigid base are known and the structure is fixed only to the base. Define the displacement vector, $u$, as a sum of the known base motion, $u_0$ and the relative displacements, $\delta u$ by the equation

$$\{u\} = \{\delta u\} + [D]\{u_0\}$$

**Equation 4-1.**

where $[D]$ is the rigid body transformation matrix that includes the effects of coordinate systems, offsets, and multiple directions. If the structure is a free body and aerodynamic effects are neglected, the base motions should not cause any static or viscous forces and

$$[K][D] = 0$$
$$\text{and } [B][D] = 0$$

**Equation 4-2.**

where $[K]$ is the stiffness matrix and $[B]$ is the damping matrix. These properties allow us to remove the reference motions from the solution as shown below.

The equilibrium equation for the whole system is

$$[M]\{\ddot{u}\} + [B]\{\dot{u}\} + [K]\{u\} = \{P\}$$

**Equation 4-3.**

Substituting **Eq. 4-1** and **Eq. 4-2** into **Eq. 4-3**, we obtain

$$[M]\{\delta\ddot{u}\} + [B]\{\delta\dot{u}\} + [K]\{\delta u\} = \{P\} - [M][D]\{\ddot{u}_0\}$$

**Equation 4-4.**

The solution matrices have not changed, but the solution, $\delta u$, must be zero at the base attachment points. The right-hand term in **Eq. 4-4** may be calculated in Simcenter Nastran using one of several options. One option is the response spectra analysis described in **"Shock and Response Spectrum Analysis"** . Another option allows you to define time-dependent gravity loads. By definition, the GRAV input generates load vectors of the form

$$\{P^g\} = [M][D]\{g\}$$

**Equation 4-5.**

where $g$ is the gravity vector. Then if $g$ were time dependent, it could be used to replace $u_0$ in **Eq. 4-4** if

$$\{g(t)\} = -[\ddot{u}_0(t)]$$

**Equation 4-6.**

## Input Data

The inertial loads method is available for all dynamic response solutions. Refer to the *Simcenter Nastran Basic Dynamic Analysis User's Guide* for the complete input procedure for using static loads in a dynamic response analysis. The method is valid for both direct and modal formulations and is automatic in nonlinear and superelement models. Briefly, the necessary input data is as follows:

1. LOADSET = N – a Case Control request for processing a LSEQ (load sequence) set.

2. LSEQ – Bulk Data that connects the forces resulting from the GRAV data to a dynamic DAREA set. Several load vectors may be included in the same LSEQ set.

3. GRAV– Bulk Data defining an acceleration vector. Several may be included to define different functions of time for the different directions.

4. SPC – remember to fix the base attachments.

5. TLOADi or RLOADi Bulk Data for the dynamic load definition. Refers to the DAREA set defined in the LSEQ data.

Other necessary inputs are the conventional DLOAD and TABLEDi data to help define the time functions. DAREA Bulk Data is not necessary.

### Recommendations

The following comments describe the benefits and drawbacks to the method:

1. The solution will avoid the problem of rigid body drift that occurs in the other methods when small errors are present in the input accelerations.

2. By eliminating large constant terms, the stress and force calculations may be numerically more precise.

3. The method is compatible with superelement and nonlinear solution sequences.

4. No special sets or SUPORT data is necessary.

5. The main drawback is that the output accelerations are also relative and will not correlate to measured accelerations.

## 4.3 Virtual Fluid Mass

A virtual fluid volume produces a mass matrix which represents the fluid coupled to a boundary consisting of structural elements and other effects, such as free surfaces, planes of symmetry, and infinite fluids. The incompressible fluid produces a mass matrix defined with full coupling between accelerations and pressures on the flexible structural interfaces. For each requested volume, the boundaries may be combinations of the following:

1. Structural finite element faces, with one or two wetted sides.

2. Free surfaces with zero pressure, but no sloshing effects.

3. Planes of symmetry with symmetric or antisymmetric motion.

4. Infinite boundaries, by default, on nonclosed volumes.

The virtual mass method is well suited for the following problem types:

1. An infinite fluid surrounding part of a structure.

2. A fluid with a free surface contained within a flexible structure.

3. Multiple fluid volumes with combinations of the options above such as a half of a ship floating in water with several internal tanks, each with free surfaces and symmetric boundaries.

Note that compressibility and surface gravity effects are neglected. It is assumed that the important frequency range for the structural modes is above the gravity sloshing frequencies and below the compressible acoustic frequencies. It is further assumed that the density within a volume is constant

and no viscous (rotational flow) or aerodynamic (high velocity) effects are present. In other words, a ship traveling at a high rate through a thick oil patch would require some extra modeling effort.

## User Interface

The user interface for a virtual fluid mass analysis is simple and straightforward. The sketch below illustrates some of the features.

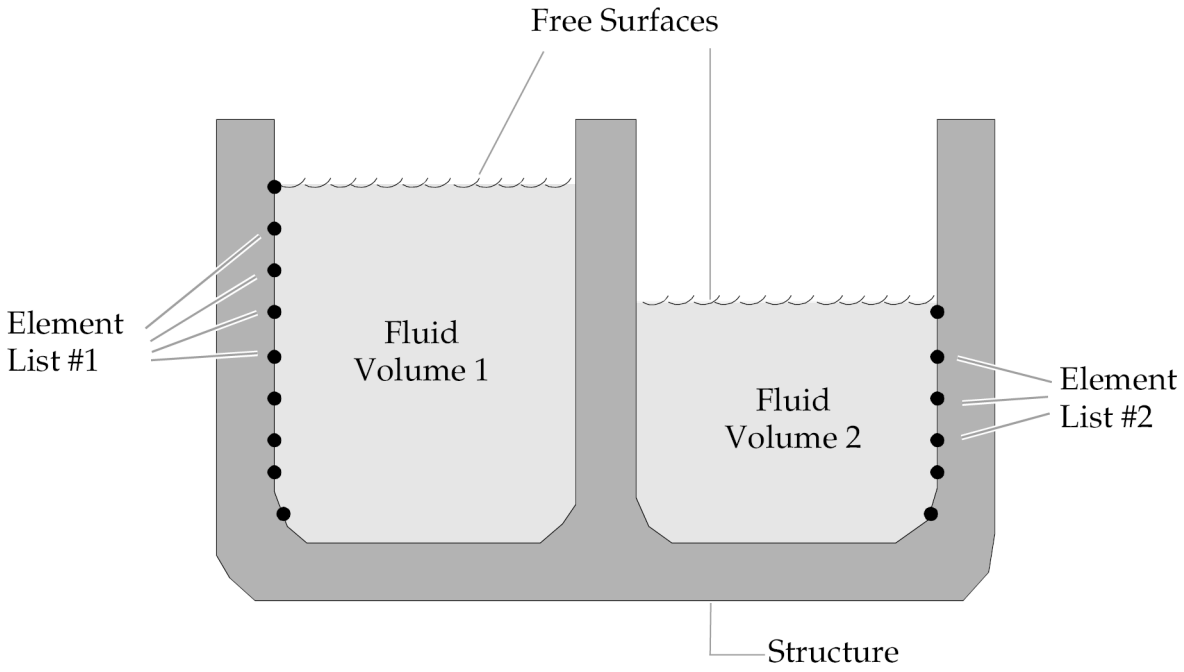



**Figure 4-1. Virtual Mass Terminology**

1. The fluid/structure interface is defined with ELIST Bulk Data, which specify a set of wetted TRIA3 and QUAD4 elements that define the structural portion of the fluid boundary.

2. Each fluid volume is defined on an MFLUID Bulk Data input, which defines the fluid density, the ELISTs, and other boundaries.

3. A set of MFLUID volumes are requested in the case control request MFLUID = SID.

4. If pressure outputs are desired they will be printed if the case control request MPRES = is used to define a set of elements. The elements must also be active in an ELIST/MFLUID volume.

An example set of input data for a small problem is shown below.

### Case Control

```
MFLUID = 25
MPRES= ALL
etc.
```

### Bulk Data

```
1       ,2       ,3       ,4       ,5       ,6       ,7       ,8       ,9       ,1
0
$MFLUID, SID    , CID    , ZFS    , RHO    , ELIST1, ELIST2, PLANE1, PLANE2
MFLUID , 25 , 0 , 115.4, .0246, 255 , , S
$ELIST , ELID  ,E1      , E2     , E3     , etc.
ELIST , 255 , 1 , THRU , 25
ETC...
```

In the input above we have defined a fluid volume with a free surface normal to the basic z direction. The xz plane (PLANE1) is a plane of symmetry and the density is 0.0246. ELIST set 255 defines the positive faces of elements in the range 1 through 25 (missing numbers are allowed) as the structural/fluid boundary.

The connected elements may be wetted on one or both sides by the same volume. For instance a baffle plate extends partially into a tank and have two sides in the same fluid volume. If a plate completely separates the two parts of the tank, it is recommended that a separate volume be used for each side. Each side of an element should be listed on a separate ELIST.

Special terms are calculated if enclosed fluid volumes do not have a free surface or a plane of anti-symmetry. Otherwise, the incompressible fluid would produce a spurious low frequency mode with a mode shape corresponding to the volume change. This mode will not affect the structural response, but it is eliminated. Because the free surface allows the fluid volume to expand, it does not have these problems.

A free surface is defined as an x-y plane in any local rectangular coordinate system. The user simply specifies a value of z as the upper limit of the fluid volume. Different fluid volumes may have different levels and orientation. It is not required that the surface coincide with the element properties since partially wetted elements are allowed.

The virtual mass fluid option may be used in all Simcenter Nastran dynamics solutions, including the following special approaches:

- Superelements – (Residual Only)
- Nonlinear Analysis – (SOL 129)
- Optimization – (does not create sensitivity matrices)

## Theoretical Summary

The following is a brief overview of the virtual mass approach. For more information, refer to the standard references on boundary elements.

The Helmholtz method used by Simcenter Nastran solves Laplace's Equation by distributing a set of sources over the outer boundary, each producing a simple solution to the differential equation. By matching the assumed known boundary motions to the effective motion caused by the sources, we can solve a linear matrix equation for the magnitude of the sources. The values of the sources determine the effective pressures and, thereby, the forces on the grid points. Combining all of these steps into a matrix equation results in a virtual mass matrix as derived below.

If $\sigma_j$ is the value of a point source of fluid (units are volume flow rate per area) located at location $r_j$, and is assumed acting over an area $A_j$, the vector velocity $\dot{u}_i$ at any other point $r_i$ is

$$\dot{u}_i = \sum_j \int_{A_j} \frac{\sigma_j e_{ij}}{|r_i - r_j|^2} dA_j$$

**Equation 4-7.**

where $e_{jj}$ is the unit vector in the direction from point $j$ to point $i$. Note that the gradient of the vector $\dot{u}_i$ is the potential function which satisfies Laplace's Equation on a term by term basis.

The other set of necessary equations are the pressures, $p_i$, at any point, $i$, in terms of the density, $\rho$, sources and geometry, namely

$$p_i = \sum_j \int_{A_j} \frac{\rho \dot{\sigma}_j e_{ij}}{|r_i - r_j|} dA_j$$

**Equation 4-8.**

The results of integrating **Eq. 4-7** and **Eq. 4-8** over the finite element surfaces are collected respectively in two matrices, $[\chi]$ and $[\Lambda]$ where

$$\{\dot{u}\} = [\chi][\sigma]$$

**Equation 4-9.**

and

$$\{F\} = [\Lambda]\{\dot{\sigma}\}$$

**Equation 4-10.**

where *F* are the forces at the grid points. The matrix [Λ] is obtained by integrating **Eq. 4-8**. An additional area integration is necessary to convert the pressures to forces. A mass matrix may now be defined using **Eq. 4-9** and **Eq. 4-10** as

$$\{F\} = [M^f]\{\ddot{u}\}$$

**Equation 4-11.**

where the virtual fluid mass matrix, $[M^f]$ is

$$[M^f] = [\Lambda][\chi]^{-1}$$

**Equation 4-12.**

## Singularities for Enclosed Volumes

Note that if [χ] is singular or nearly singular, the mass matrix will cause problems with the coupled solution. The singularity is active when a set of nonzero sources can produce zero velocities on the structural surfaces. This occurs when the fluid is completely enclosed by the structure and with planes of only symmetric motion. Large pressures could occur within the fluid volume from an infinitesimal volume change on the incompressible fluid.

An alternate calculation is performed when the [χ] matrix is potentially singular. See Remark 8 on the MFUID Bulk Data description for the specific instances. An additional source, $s_2$, is added near the center of each MFLUID volume. The expanded versions of

$$\{\dot{u}\} = [\chi][\sigma]$$

(for more information, see **Theoretical Summary**) and

$$\{F\} = [\Lambda]\{\dot{\sigma}\}$$

(for more information, see **Theoretical Summary**) are

$$\{\dot{u}\} = [\chi]\{\sigma\} + \{\chi_2\}s_2$$

**Equation 4-13.**

$$\{F\} = [\Lambda]\{\dot{\sigma}\} + \{\Lambda_2\}s_2$$

**Equation 4-14.**

An additional equation provides that the sum of the squares of the ordinary sources in the vector, $\{\sigma_1\}$ is minimized and a set of Lagrange multipliers, $\lambda$ are used to enforce **Eq. 4-13** as constraints. The function, *U*, to be minimized is defined as

$$U = \frac{1}{2}[\sigma]^T\{\sigma\} + [\lambda]^T\{\{\dot{u}\} - [\chi]\{\sigma\} - \{\chi_2\}s_2\}$$

**Equation 4-15.**

Taking the derivatives of *U* with respect to $\lambda$, $\sigma_1$, and $s_2$, we obtain three sets of equations. Combining them, we obtain the following matrix equation:

$$\begin{bmatrix} 0 & \chi & \chi_2 \\ \chi^T & -I & 0 \\ \chi_2^T & 0 & 0 \end{bmatrix} \begin{Bmatrix} \lambda \\ \sigma \\ s_2 \end{Bmatrix} = \begin{Bmatrix} \dot{u} \\ 0 \\ 0 \end{Bmatrix}$$

**Equation 4-16.**

The vector $\{\sigma\}$ may be eliminated by solving the second row partition and substituting into the first row partition of the matrix, with the result:

$$\begin{bmatrix} \chi\chi^T & \chi_2 \\ \chi_2^T & 0 \end{bmatrix} \begin{Bmatrix} \lambda \\ s_2 \end{Bmatrix} = \begin{Bmatrix} \dot{u} \\ 0 \end{Bmatrix}$$

**Equation 4-17.**

**Eq. 4-17** may be solved for $\{\lambda\}$ and $s_2$, which are then used to obtain pressure. To obtain the pressure, we substitute for $\{\sigma\}$ in **Eq. 4-14** to obtain the matrix equation

$$\{p\} = \begin{bmatrix} \Lambda\chi^T & \Lambda_2 \end{bmatrix} \begin{Bmatrix} \dot{\lambda} \\ \dot{s}_2 \end{Bmatrix}$$

**Equation 4-18.**

Eq. 4-17 and Eq. 4-18 may then be combined into a single matrix defining the fluid, namely

$$\{F\} = [M_f]\{\ddot{u}\}$$

**Equation 4-19.**

where:

$$[M_f] = \begin{bmatrix} \Lambda\chi^T & \Lambda_2 \end{bmatrix} \begin{bmatrix} \chi\chi^T & \chi_2 \\ \chi_2^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} I \\ 0 \end{bmatrix}$$

**Equation 4-20.**

Note that the size of the matrix equation is only one term larger than the nonconstrained case. Although it will be well behaved for the enclosed volume case, the overall incompressible constraint is lost.

Other methods to avoid the singular matrix are as follows:

1. Put one or more small holes in the boundary by removing an element ID in an unimportant area from the ELIST. The fluid will then leak out to an infinite domain of fluid.

2. Define a free surface near the top of the container and modify the ELIST to remove elements above the surface.

3. Constrain the structure to eliminate any net change to the enclosed volume. In other words generate an MPC equation such that

$$\sum_i A_i \bar{n}_i \cdot \bar{u}_i = 0$$

**Equation 4-21.**

where $i$ is a boundary grid point and $A_i$ and $\bar{u}_i$ are the effective area and normal vector, respectively. (This is not an easy task.)

Fortunately, there are no problems in which free surfaces or planes of symmetry with antisymmetric motion are present since the pressure must be zero on these boundaries.

## Using Phantom Structural Boundaries

In many cases an ELIST boundary is desired where no CQUAD4 or CTRIA3 elements exist. Examples occur:

- When grid points connected by other element types, such as CHEXA, CSHEAR, CBEAMs, form the boundary.
- When only a small hole connects two separate fluid volumes (since poor numerical conditioning (less roundoff) is expected with a single MFLUID).
- If the fluid contains two free surfaces due to entrapped air at a different pressure.
- If the fluid volume is a complex labyrinth, such as a boiler made with many tubes.

The phantom boundary may be constructed from CTRIA3 and CQUAD4 elements which have zero or nearly zero stiffness in the normal direction. The PSHELL bending thickness is used for this purpose. The membrane stiffness is optional. If no other structural elements are present, a small stiffness is desired for both directions to avoid automatic constraints.

Note that this method will couple only the fluid displacements in the normal direction. Edge/corner effects and tangential motions will be approximate.

## Gravity Effects

The free surfaces produced by the Virtual Fluid Mass option are simple planes of antisymmetric motion with a null pressure assumed at the location of the free surface plane. Effects such as fluid sloshing due to gravity waves are assumed to be uncoupled from the higher frequency structural modes.

An approximation to the gravity effects for fluids with finite boundaries may be modeled using a phantom boundary instead of a free surface. Spread grid points and plate elements over the surface and constrain the in-plane motions and rotations to zero. Give the plates a small membrane thickness and no bending material property. Add scalar springs (CELASi) in the direction normal to the plane with stiffnesses, $K_i$, calculated from

$$K_i = A_i \cdot \rho g$$

**Equation 4-22.**

where $A_i$ is the area under the point, $\rho$ is the density, and $g$ is the gravitational constant. Note that these springs will affect the rigid body motion of the whole system and this method should be used with caution. Another drawback is that the extra gravity boundary may result in a completely enclosed fluid volume.

## Examples

The legal and illegal types of fluid boundaries are illustrated in **Figure 4-2**. The categories are GOOD = legal configuration, BAD = illegal boundary, and MAYBE = conditionally legal, which could be permitted if a phantom boundary is used.

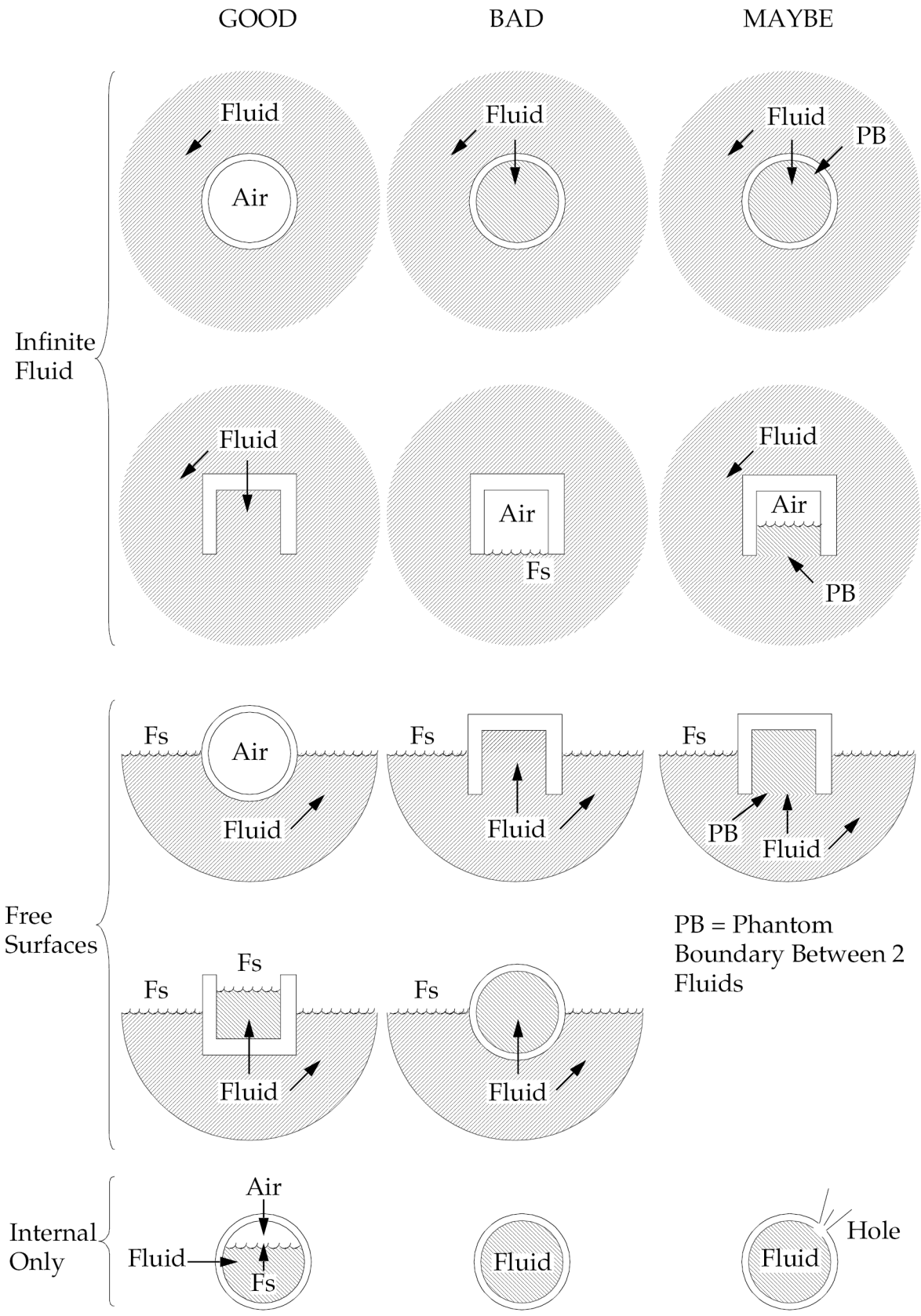



**Figure 4-2. Fluid Combinations**

# MFLUID Performance Options

When virtual fluid mass (MFLUID) is combined with structural elements in a model, the dense virtual fluid mass portion can cause normal mode solution times to be large. There are a couple of techniques which will reduce the mass matrix, and subsequently reduce this solution time:

- The VMOPT method requires the parameter definition VMOPT=2 in the input file.
- The CMS (Component Modal Synthesis) method is similar to the VMOPT method, but requires SPOINT and QSET bulk data entries in the input file.

There is no significant performance or accuracy advantage in using one method instead of the other. The VMOPT method is simpler to setup. Both are discussed as follows.

## VMOPT Method

The VMOPT method requires the parameter definition VMOPT=2 in the input file. Specifically, when parameter VMOPT=2, the normal mode solution will automatically be split into two steps:

- The VMOPT method first computes the normal modes of the structure without the fluid mass, producing the “dry” modes.
- Then the fluid mass is added into the modal basis during a second normal mode solution to produce the “wet” modes.

You save considerable solution time since the fluid mass is added to the reduced, modal basis in the second normal mode solution, as opposed to the entire (structural + fluid) mass matrix in a physical basis.

For best accuracy, you should compute 2 to 4 times the number of “dry” modes than the desired number of “wet” modes. For example, if you want 25 “wet” modes, you should specify that the software solve for 50 to 100 “dry” modes. In general, the ratio of “dry” modes to “wet” modes increases with the density of the fluid. For gases, a ratio of 2 might be sufficient, but for liquids a ratio of 4 might be justified.

If you need a different eigenmethod (EIGR/EIGRL selection) for the “dry” and “wet” portions of the solution, use the RSMETHOD case control command to specify the “dry” method, and the METHOD case control command to specify the “wet” method. If you use the same eigenmethod for both the “dry” and “wet” modes, the single METHOD case control command is used in both the “dry” and “wet” portions.

See the VMOPT parameter in the *Simcenter Nastran Quick Reference Guide*.

### VMOPT METHOD INPUT EXAMPLE

```
SOL 103
CEND
$*
```

```
$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*  CASE CONTROL
$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*
$* SELECTING 'DRY' EIGEN METHOD
RSMETHOD = 50
$*
$* SELECTING 'WET' EIGEN METHOD
METHOD = 1
$*
$* SELECTING VIRTUAL FLUID DEFINITION
MFLUID=200
$*
SPC=1
$*
$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*  BULK DATA
$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*
BEGIN BULK
$* ELEMENT SETS DEFINING FLUID VOLUME
ELIST        100    4282    THRU    7754
$*
$* VIRTUAL FLUID DEFINITION
MFLUID       200      10     30.   1.0-9     100                N       N
$*
$*  PARAM CARDS
PARAM   AUTOSPC YES
PARAM   POST          -2
$* SELECTING THE VMOPT METHOD
PARAM   VMOPT   2
$*
$* EIGEN METHOD FOR 'DRY' MODES
EIGRL         50                      90
$*
$* EIGEN METHOD FOR 'WET' MODES
EIGRL          1     0.1   200.0               0       7        MASS
$*
$* NO SCALAR POINTS OR QSET REQUIRED FOR VMOPT METHOD
....
```

## CMS Method

The CMS method is similar to the VMOPT method. Since a “dry” eigensolution without the virtual fluid mass occurs first, the “dry” modes are reduced to modal coordinates, the virtual fluid mass is added, then the “wet” eigensolution occurs to find the modes of the structure and fluid together. The “dry” modes which are reduced to modal coordinates are assigned to DOF which you define in the input file. Specifically, the CMS method requires you to:

- Add scalar points using the SPOINT bulk data entry. The number of scalar points should be at least as many as the desired “dry” modes.

- Include the SPOINT IDs in a QSET bulk data entry.

Similar to the VMOPT method, if you need a different eigenmethod (EIGR/EIGRL selection) for the “dry” and “wet” portions of the solution, use the RSMETHOD case control command to specify the “dry” method, and the METHOD case control command to specify the “wet” method. If you use the same eigenmethod for both the “dry” and “wet” modes, the single METHOD case control command is used in both the “dry” and “wet” portions.

## CMS METHOD INPUT EXAMPLE

```
SOL 103
CEND
$*
$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*$*  CASE CONTROL
$*$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*
$* SELECTING 'DRY' EIGEN METHOD
RSMETHOD = 50
$*
$* SELECTING 'WET' EIGEN METHOD
METHOD = 1
$*
$* SELECTING VIRTUAL FLUID DEFINITION
MFLUID=200
$*
SPC=1
$*
$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*$*  BULK DATA
$*$*$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$*
BEGIN BULK
$* ELEMENT SETS DEFINING FLUID VOLUME
ELIST        100    4282    THRU    7754
$*
$* VIRTUAL FLUID DEFINITION
MFLUID       200      10     30.   1.0-9     100                N       N
$*
$*  PARAM CARDS
PARAM   AUTOSPC YES
PARAM   POST          -2
$*
$* EIGEN METHOD FOR 'DRY' MODES
```

```
EIGRL         50                        90
$*
$* EIGEN METHOD FOR 'WET' MODES
EIGRL          1      0.1    200.0               0       7        MASS
$*
$* NUMBER OF SCALAR POINTS SHOULD BE AT LEAST THE NUMBER OF 'DRY' MODES
SPOINT    600001    THRU  600500
$*
$* ALL SCALAR POINT ID's SHOULD BE INCLUDED IN THE QSET
QSET1          0  600001    THRU  600500
$*
....
```

## 4.4 Advanced Methods for Vibro-Acoustic Analysis

Many of the other Simcenter Nastran features should be available for acoustic analysis. Some potential advanced modeling and solution features that will be useful are as follows:

- Modal formulation with superelements: The use of Component Mode Synthesis is recommended for reducing the size of the matrices and the solution costs. The degrees-of-freedom of the interior fluid points will be replaced by the modes of the fluid with rigid container boundaries. The uncoupled structural points may be replaced by modal coordinates defined by a structure in a vacuum.

- Virtual Fluids: The MFLUID Bulk Data input may be used to represent incompressible fluids attached to the structure. This may be used to approximate the low frequency effects of an infinite exterior fluid.

- Decoupled Response Solutions: In most acoustic problems with a load-carrying structure, the structural modes are not affected by the small mass of the air. In turn, the high stiffness of the structure is effectively rigid for the purposes of computing acoustic modes in the air. Forced vibration problems may then be analyzed as two decoupled symmetric solutions:

  - Constrain the entire structural model and perform a forced response analysis on the air (use SLOAD inputs to define the volume inputs from a known source) and save the resulting boundary pressures.

  - Restart the job with the fluid points constrained and a DMAP alter to add the pressure forces to the frequency- or time-dependent load vectors. The resulting structural model will respond to the pressures from the first run.

- Random and Response Spectrum Methods: The Simcenter Nastran random analysis options will also be useful for acoustic analysis. This capability requires a set of unit forced responses in the frequency domain (SOLs 108 and 111) and spectral densities of the forcing functions (RANDPS and RANDPEX inputs). It will calculate the combination spectral densities and autocorrelations of selected output quantities. In acoustic problems, it could be used for problems having several correlated sources such as an automobile interior.

# 5. Dynamic Solution Techniques

## 5.1 Overview of Dynamic Solution Techniques

While planning the modeling process and making provisions for the reduction steps and boundary conditions, the dynamics engineer should also consider the various options in the actual solution. In addition to the basic linear transient and frequency response processes, which are described fully in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*, other solution options and variations for dynamic analysis are available.

Complex eigenvalue analysis is used for determining stability and assessing the overall dynamic response of a system. Complex eigenvalue analysis must be used for eigenvalue analysis when unsymmetrical matrix terms or large damping effects are present.

Nonlinear transient analysis accounts for nonlinear element effects caused by nonlinear materials, variable surface contact, and large displacements. The procedures for the data input and controls are significantly different than those for the basic linear transient solutions.

An option for using Fourier transform methods is available for the solution of special problems that involve repeating loads in time.

Viscoelastic parameters, which can be applied in the solution of problems involving rubber-like materials, are available as a special process in the direct frequency response solutions.

The solution methods for free bodies are both necessary analysis procedures for unconstrained models and diagnostic tools for constrained bodies. For instance, an airplane or missile in flight (without proper supports) will have a singular stiffness matrix and (will) produce large displacements in the results.

Finally, aeroelastic solutions such as flutter analysis, gust response, and static aeroelasticity are also discussed in this chapter because of their unique algorithms and solution procedures.

## 5.2 Review of Dynamic Excitations

The methods used for generating dynamic loads in Simcenter Nastran are very different from those used for static loads. Dynamic loads generally vary with time or frequency. They may also be applied with different phases or time lags to different portions of the structure. The most general case would be a structure having a different load history for each direction on each point on the model, which could require thousands of tabular inputs.

Fortunately, most applications require only a simple pattern of load variations with time and geometry. The engineer usually encounters a single time function applied to a portion of the structure or a general time function applied to a few structural points. An example of the first case would be wind gust loads on an aircraft. An example of the second case would be an automobile traveling over a rough road. Simcenter Nastran requires only a minimum input for either case.

Because of the many possible types of dynamic loading conditions, the input options are also numerous. Most of the basic options are described in "**Modal Versus Direct Frequency Response** " and "**Modal Versus Direct Transient Response** " in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*. Some of the advanced methods are described below.

## Subcases in Dynamic Analysis

The Case Control options in dynamic analysis are less flexible than statics and nonlinear analysis. Some of the statics options that are *not* available in dynamic response analysis and complex eigenvalue analysis include:

- Changes in boundary conditions between subcases.
- SUBSEQ and SUBCOM solution combinations.
- Grid point forces and element strain energy outputs.

Otherwise, each type of solution uses the Case Control subcase for a different purpose. Some uses for subcases in dynamics analysis are given below.

1. Subcases are essential for most dynamic optimization problems (SOL 200). Each subcase may define a different type of analysis or a different load.
2. In a superelement analysis, the subcases may be used to specify different upstream LOADSET data and output processing for different SE components.
3. Subcases are recommended in nonlinear transient analysis (SOL 129) for applying loads in a sequence or changing the time step method. A single load may be applied in steps or a second load may be added with new subcases. Changing time step size or the solution method may be necessary to overcome a troublesome divergence region. Subcases also define the allowable solutions that may be used in restarts. Use them liberally.
4. multiple subcases are available for frequency response analysis (SOLs 108, 111 and 118) for the purpose of solving multiple loading conditions more efficiently. (Each frequency requires a matrix decomposition and each additional load vector may be processed at this time with small cost.) Another use is in random analysis where several loads need to be combined, each with a different spectral density distribution.
5. Subcases are used in complex eigenvalue analysis (SOLs 107 and 110) for processing multiple direct input matrices. Typical applications are multiple control system parameters or externally generated aerodynamic matrices representing a variety of flight conditions.
6. Cyclic symmetry analysis (SOLs 114 through 118) requires subcases to distribute the loads over different petals (i.e., sections of the structure). The methods are identical to cyclic static analysis.
7. multiple subcases are not recommended for linear transient analysis.

## Loading Methods

An overview of the dynamic load algorithm is shown in **Figure 5-1**. The input data follow two major paths: the DAREA method versus the LSEQ method. The details of the data inputs are given in the appendices and in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*.

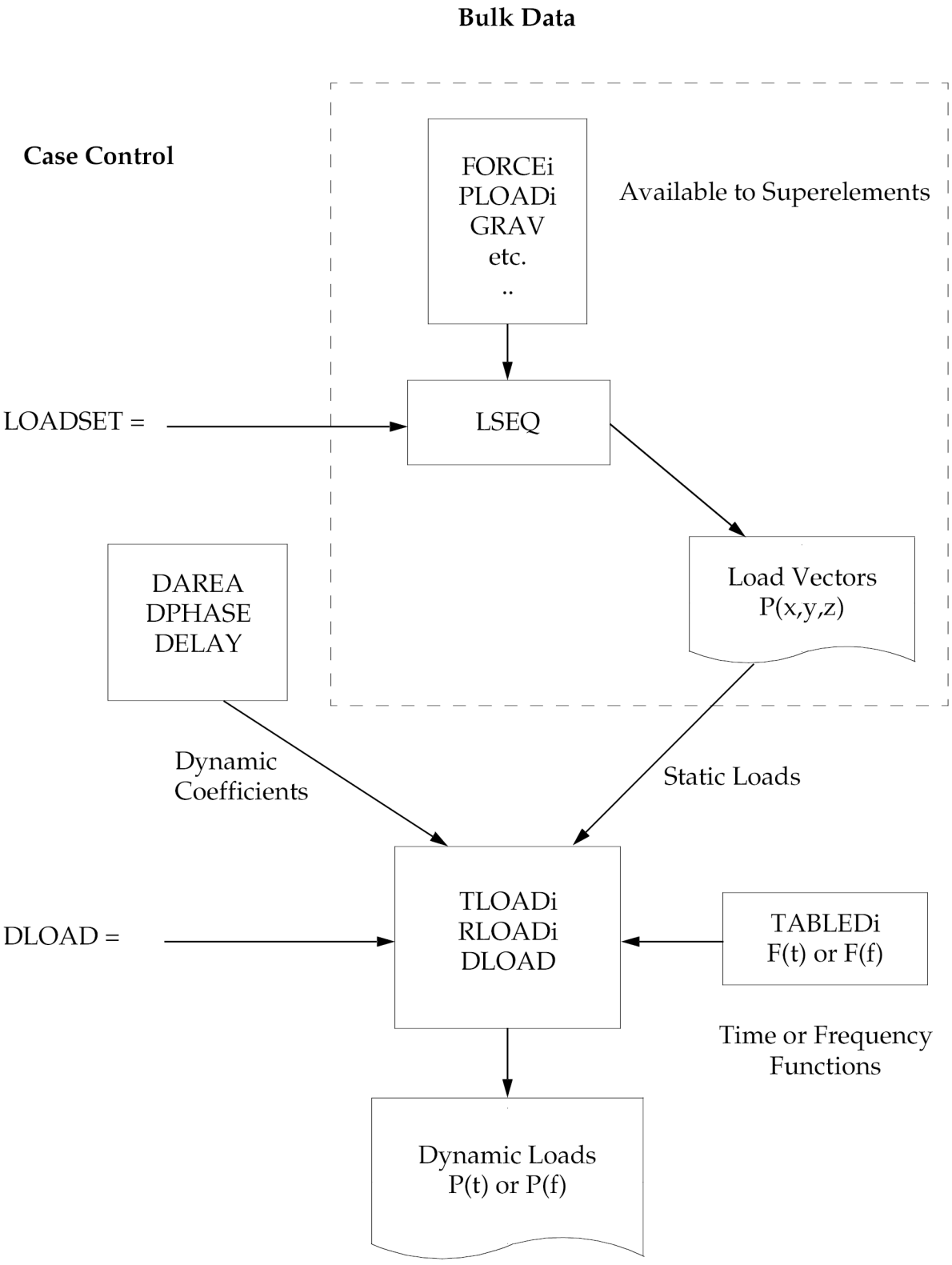



**Figure 5-1. Input Data Relationships for Dynamic Loads**

- The LSEQ Bulk Data act much like a set of Case Control commands to generate static loads and assign them to a specific static load vector. These vectors are assembled, reduced, and combined for all superelements. A different load vector may be required for each unique function of time or frequency.

- The DAREA, DPHASE, and DELAY Bulk Data provide a direct method of distributing the dynamic loads over the grid points. The DAREA factors correspond to a specific load factor for a specific degree-of-freedom, much like an area under a pressure load. The DPHASE and DELAY data are used when the same load versus time function is applied to different points at different times. An example of a DPHASE application is the frequency response of an automobile traveling over a highway with a sinusoidal undulation. As the speed increases the input frequency changes, but the phase difference between the axles is constant.

The TLOADi Bulk Data inputs each define a function of time with coefficients or TABLEDi references. The RLOADi Bulk Data inputs each define a function of load versus frequency with complex TABLEDi inputs. The DLOAD entry is used to combine the different load functions, either time or frequency dependent. These functions may be associated with LSEQ-type vectors or DAREA coefficients, or both.

## Recommendations

The following guidelines should be observed when applying dynamic loads.

### TABLEDi

1. Linear interpolation is used between tabular points. This may lead to accumulated roundoff and drift errors over a long-period transient analysis. Fix this problem by adding a correcting load function to the same points on subsequent runs.

2. If a jump occurs (two points with equal values of X), the value of Y at the jump is the average of the two points.

### Static Preloads

1. An initial linear static preload that is released suddenly for a transient analysis may be performed in two stages in one run. In stage 1, apply the static loads and use large time steps to suppress the higher frequency motions. In stage 2, remove the loads and use small time steps to capture the dynamic action. Alternatively, a nonlinear analysis may be restarted from a nonlinear static analysis, making the process nearly automatic.

2. Simple static loads may be included through the dynamic load methods. Define a TLOAD2 or an RLOADi with a constant function and add it via the DLOAD Bulk Data input.

3. Thermal loads may be applied via the LSEQ method, but note that the dynamic stress recovery method will not account for the additional strain or stress.

### Matching Initial Conditions

1. In linear transient analysis, the load at time, t = 0.0 is always ignored. If the load is actually a preload, the initial conditions (TIC data) should correspond to this condition. Otherwise, if the load is being suddenly applied, the loading functions will start at t = $\Delta$t.

2. If a free body is being accelerated by an applied load, the calculated response will also be delayed by a fraction of a time step.

### Setup for Random Analysis

A Simcenter Nastran random analysis requires a preliminary frequency response analysis to generate the proper transfer functions that define the output/input ratios. The squared magnitudes of the results are then multiplied by the spectral density functions of the actual loads. Normally, the inputs are unit loads (e.g., one g constant magnitude base excitation or a unit pressure on a surface).

### Fluids and Acoustic Loads

Normally, fluid and acoustic elements are not loaded directly, but are excited by the connected structures. However, loads on these special grid points may be used to represent sources of fluid motion such as a small loudspeaker. The actual units of these loads are the second derivative of volume flow with respect to time.

## 5.3 Complex Eigenvalue Analysis

Complex eigenvalue analysis is necessary when the equation of motion for a mechanical system contains any of the following:

- A damping matrix (to account for viscous damping).
- A complex stiffness matrix (to account for structural damping).
- An unsymmetrical mass or stiffness matrix.
- A mass matrix that is not positive semidefinite.

Complex eigenvalue analysis is required in the analysis of aeroelastic flutter, acoustics, and rotating bodies, among others. The results of a complex eigenvalue analysis are typically used to examine the effect of damping or determine the stability of a system when the system contains sources of energy like rotating components.

You can use either of two solution sequences to perform complex eigenvalue analysis:

- SOL 107 Direct Complex Eigenvalue Analysis
- SOL 110 Modal Complex Eigenvalue Analysis

When you use either solution sequence, you must specify the complex eigenvalue method. Simcenter Nastran uses this method to calculate the eigenvalues and eigenvectors.

Although both solution sequences have the same applicability, modal complex analysis allows you to reduce the size of the complex eigenvalue problem. The corresponding computational efficiency is advantageous when the model is very large. However, reducing the size of the complex eigenvalue problem can lead to the loss of accuracy in the results.

## Real and Complex Eigenvalue Problems

The unforced motion of a mechanical system can be expressed as the superposition of characteristic motions called *modes*. In the most general form, the unforced motion can be written as:

$$u(t) = Real\left(\sum \{\phi_i\} e^{(\alpha_i + i\omega_i)t}\right)$$

**Equation 5-1.**

where $(\alpha_i + i\omega_i)$ and $\varphi_i$ are the eigenvalue and eigenvector for the $i^{th}$ mode, respectively.

The real part of the eigenvalue $\alpha_i$ represents the rate of decay or rate of divergence for the $i^{th}$ mode. The numerical value $\omega_i$ of the imaginary part of the eigenvalue represents the natural frequency for the $i^{th}$ mode. The eigenvectors $\varphi_i$ contain complex numbers that represent the relative magnitude and phase of motion between the various degrees-of-freedom (DOF).

A *real eigenvalue problem* occurs if viscous and structural damping are ignored, the mass and stiffness matrices are real and symmetric, and the mass matrix is positive semidefinite. A real eigenvalue problem is given by:

$$[K - \lambda_i M]\{\phi_i\} = 0$$

**Equation 5-2.**

where *K* and *M* are the mass and stiffness matrices, respectively, and $\lambda_i = \omega_i^2$ and $\varphi_i$ are the eigenvalue and eigenvector for the $i^{th}$ mode, respectively.

For a real eigenvalue problem:

- The real part of the eigenvalues $\alpha_i$ are zero and the motion neither decays nor diverges, but simply oscillates with a constant magnitude indefinitely.
- The eigenvectors are real and, for a given mode, the motion of all the DOF is in phase.
- The eigenvectors are orthogonal. Orthogonality of the eigenvectors allows:

- The equations of motion to be transformed into modal space where they are uncoupled.
- The size of the eigenvalue problem to be reduced.

For additional information on real eigenvalue problems, see the *Simcenter Nastran Basic Dynamic Analysis User's Guide*.

If any of the conditions required for a real eigenvalue problem are not met, the problem becomes a *complex eigenvalue problem*. A complex eigenvalue problem is given by:

$$[K + p_i B + p_i^2 M]\{\phi_i\} = 0$$

**Equation 5-3.**

where:

$$p_i = \alpha_i + i\omega_i$$

**Equation 5-4.**

are the eigenvalues and $\varphi_i$ are the eigenvectors for the $i^{th}$ mode. Unlike real eigenvalue analysis, only weak forms of orthogonality apply to complex eigenvalue analysis.

To solve a complex eigenvalue problem, the software uses a complex eigenvalue method that you specify. Simcenter Nastran supports four general types of complex eigenvalue methods including Hessenberg methods, the Inverse Power method, Complex Lanczos methods, and the Iterative Schur-Rayleigh-Ritz method.

The Hessenberg methods are direct methods, whereas the other complex eigenvalue methods are iterative. To provide faster convergence, some of these methods like the Inverse Power method use a special form of Eq. 5-3 called the *shifted form*. From a shift point, $p_0$, the relative distance to a root of the characteristic equation is:

$$\Delta_i = p_i - p_0$$

**Equation 5-5.**

and Eq. 5-3 becomes:

$$[K + p_0 B + p_0^2 M]\{\phi_i\} = -\Delta_i[B + 2p_0 M + M\Delta_i]\{\phi_i\}$$

**Equation 5-6.**

Simcenter Nastran iterates on Eq. 5-6 by solving for trial eigenvectors on the left side of the equation using previous estimates for the eigenvectors on the right side of the equation. The iterative process continues until the solution converges.

## Complex Eigenvalue Methods

Both SOL 107 and SOL 110 require you to specify a complex eigenvalue method. Simcenter Nastran uses the complex eigenvalue method that you specify to solve the complex eigenvalue problem. Simcenter Nastran supports four general types of complex eigenvalue methods. They include:

- Hessenberg methods:

  QZ Hessenberg method

  QR Hessenberg method

  LR Hessenberg method

- Inverse Power method
- Complex Lanczos methods:

  Block Complex Lanczos method

  Single Vector Complex Lanczos method

- Iterative Schur-Rayleigh-Ritz (ISRR) method

The complex eigenvalue method that Simcenter Nastran uses depends on the method you specify on an EIGC bulk entry. However, you can override the method specified on the EIGC bulk entry by specifying system cell 108. A summary of each general complex eigenvalue method that Simcenter Nastran supports is provided in the following table.

| Method | Description and Applicability |
|---|---|
| Hessenberg | The Hessenberg methods are generally reliable and economical for small to moderate-size problems. The Hessenberg methods compute all eigenvalues and a specified number of eigenvectors. The applicability of the QR Hessenberg and LR Hessenberg methods has been superseded by the QZ Hessenberg method.<br>The QZ Hessenberg method is used by default when the Hessenberg method is specified on the EIGC bulk entry. However, the QR Hessenberg and LR Hessenberg methods can still be selected with system cell 108. |
| Inverse Power | The applicability of the Inverse Power method has been superseded by the Block Complex Lanczos method. |

| Method | Description and Applicability |
|---|---|
| Complex Lanczos | The Block Complex Lanczos method is more reliable and will not accept inaccurate eigenvalues, which the Single Vector Complex Lanczos method has a tendency to do. Given the same input, the Block Complex Lanczos method may often accept fewer eigenvalues. The applicability of the Single Vector Complex Lanczos method has been superseded by the Block Complex Lanczos method.<br><br>The Block Complex Lanczos method is used by default when the Complex Lanczos method is specified on the EIGC bulk entry. However, the Single Vector Complex Lanczos method can still be selected with system cell 108. |
| ISRR | The Iterative Schur-Rayleigh-Ritz method works well on sparse matrices, confines the search region to a circle centered on the origin of the complex plane, and provides some reliability that all modes within the circle have been found. |

For additional details on complex eigenvalue methods, see “Complex Eigenvalue Analysis” in the *Simcenter Nastran Numerical Methods User’s Guide*.

## Direct Complex Eigenvalue Analysis (SOL 107)

When you use direct complex eigenvalue analysis (SOL 107) to determine the eigenvalues and eigenvectors for a complex eigenvalue problem, Simcenter Nastran:

1. Creates the matrices of the equation of motion and uses them to formulate the corresponding complex eigenvalue problem.

2. Uses the complex eigenvalue method that you specify to extract eigenvalues and eigenvectors from the complex eigenvalue problem formulation.

## Modal Complex Eigenvalue Analysis (SOL 110)

Modal complex analysis (SOL 110) combines real eigenvalue analysis with complex eigenvalue analysis. When you use modal complex eigenvalue analysis to determine the eigenvalues and eigenvectors for a complex eigenvalue problem, Simcenter Nastran:

1. Formulates a real eigenvalue problem using the structural stiffness and mass matrices.

2. Solves the real eigenvalue problem using the real eigenvalue method that you specify to obtain a set of real eigenvalues and eigenvectors.

3. Applies orthogonality to reduce the mass, stiffness, and damping matrices to h-set size matrices.

4. Formulates the complex eigenvalue problem from the reduced matrices.

5. Uses the complex eigenvalue method that you specify to extract eigenvalues and eigenvectors from the complex eigenvalue problem formulation.

Because the complex eigenvalue method solves a reduced system, modal complex eigenvalue analysis typically requires less computational effort than direct complex eigenvalue analysis. This is particularly true when large models are reduced to a much smaller size. However, results from modal complex eigenvalue analysis are typically less accurate than those from direct complex eigenvalue analysis. The difference in accuracy is depends on:

- The extent the model size is reduced.
- The influence of non-structural effects on the behavior of the structure. For example, servo controls act like large masses that can reduce the frequency of local high frequency modes.

## Input File Requirements

The input file requirements for direct complex eigenvalue analysis (SOL 107) and modal complex eigenvalue analysis (SOL 110) are the same except that you must also specify real eigenvalue analysis parameters for a modal complex eigenvalue analysis (SOL 110). To do so, include a METHOD case control command that references either an EIGR or EIGRL bulk entry.

For both solution sequences, you must include a CMETHOD case control command that references an EIGC bulk entry. On the EIGC bulk entry, you specify a complex eigenvalue method and parameters associated with the method. The complex eigenvalue methods that you can select with an EIGC bulk entry include:

- QZ Hessenberg method
- Inverse Power method
- Block Complex Lanczos method
- Iterative Schur-Rayleigh-Ritz (ISRR) method

You can optionally use system cell 108 to override the method specified on the EIGC bulk entry and select the following complex eigenvalue methods:

- QR Hessenberg method.
- LR Hessenberg method.
- Single Vector Complex Lanczos method.

Even though you can override the complex eigenvalue method specified on an EIGC bulk entry by specifying system cell 108, you still need to include an EIGC bulk entry. Regardless of which complex

eigenvalue method you specify, the parameters for the complex eigenvalue method are defined on an EIGC bulk entry.

Depending on the complex eigenvalue method selected, continuation lines may be required for the EIGC bulk entry. The continuation lines are used to define search regions or shift points when using the Inverse Power, Complex Lanczos, and ISRR methods.

For more information on the EIGC bulk entry or system cell 108 settings, see EIGC in the *Simcenter Nastran Quick Reference Guide*.

Complex eigenvalues are output by default, but you must request eigenvector output.

- During a direct complex eigenvalue analysis (SOL 107), complex eigenvectors in physical space are calculated. To request output of these results, include either a DISPLACEMENT or SDISPLACEMENT case control command.

- During a modal complex eigenvalue analysis (SOL 110), both complex eigenvectors and real eigenvectors are calculated.

  - To request output of the real eigenvectors, include an SVECTOR case control command.

  - To request output of the complex eigenvectors expressed in physical space, include a DISPLACEMENT case control command.

  - To request output of the complex eigenvectors expressed in modal space, include an SDISPLACEMENT case control command.

## Complex Eigenvalue Analysis Output

Complex eigenvalues are output by default. Often, complex eigenvalues occur as either conjugate pairs or pairs having opposite signs but identical numerical values.

- If the complex eigenvalue problem has real, symmetric mass, damping, and stiffness matrices, the complex eigenvalues occur in conjugate pairs.

- If the complex eigenvalue problem does not contain a damping matrix, the complex eigenvalues occur in plus/minus pairs.

Because the Hessenberg methods compute all the eigenvalues, you will be able to observe pairing of eigenvalues, if such pairing exists. For other methods, you may or may not be able to observe pairing of eigenvalues. For example, methods that use shift points focus on one half of the complex plane and not the other. Such methods may not compute pairs of complex eigenvalues.

Typical output for a complex eigenvalue problem without a damping matrix is as follows:

```
                C O M P L E X   E I G E N V A L U E   S U M M A R Y
ROOT     EXTRACTION           EIGENVALUE                FREQUENCY
DAMPING
 NO.        ORDER         (REAL)           (IMAG)          (CYCLES)
COEFFICIENT
  1          31       5.812789E+01   -5.747926E+03    9.148109E+02
-2.022569E-02
  2          32      -5.812789E+01    5.747926E+03    9.148109E+02
2.022569E-02
  3          33       6.295126E+01   -6.293827E+03    1.001694E+03
-2.000413E-02
  4          34      -6.295126E+01    6.293827E+03    1.001694E+03
2.000413E-02
  5          35       6.928994E+01   -6.839596E+03    1.088556E+03
-2.026141E-02
  6          36      -6.928994E+01    6.839596E+03    1.088556E+03
2.026141E-02
......
......
```

In the “ROOT NO.” column, the eigenvalues are listed in order of ascending frequency, not in the order of extraction. The column labeled “(REAL)” contains $\alpha_i$ in rad/unit time, and the column labeled “(IMAG)” contains $\omega_i$ in rad/unit time. The column labeled “FREQUENCY” contains the frequency in cycles/unit time. The values listed in the “DAMPING COEFFICIENT” column are dimensionless and computed from:

$$(g) = -2\alpha / |\omega|$$

The damping coefficient is approximately twice the value of the conventional modal damping ratio. If the magnitude of the damping coefficient is less than $5.0 \times 10^{-4}$, it is reset to zero. Damping coefficient has historically been popular with aeroelasticians who are one of the primary users of the complex eigenvalue methods.

Typical complex eigenvector output is as follows:

```
                    C O M P L E X   E I G E N V E C T O R   N O .   1
                                 (REAL/IMAGINARY)

POINT ID.   TYPE     T1          T2             T3             R1
R2        R3
     100      G           7.138289E-01   8.158927E-03   5.122094E-03
                          1.891167E-03  -1.291169E-04   1.674138E-05
     211      G           9.845528E-01   1.892782E-04
                          2.257681E-04  -2.121470E-06
     212      G           9.845528E-01   1.892782E-04
                          2.257681E-04  -2.121471E-06
     213      G                         -9.993178E-05
                                        -7.522478E-07
```

```
      214      G                    -9.993178E-05
                                    -7.522480E-07
      221      G          1.000000E+00   1.797627E-02
                         -2.485324E-16  -2.859004E-04
      222      G          1.000000E+00   1.797627E-02
                          0.0           -2.859004E-04
      223      G          8.619662E-01   1.166052E-02
                          2.253959E-03  -1.986565E-04
      224      G          8.619662E-01   1.166052E-02
                          2.253959E-03  -1.986565E-04
      301      G                                      2.272419E-06
                                                     -4.103030E-08
      302      G                                      2.272419E-06
                                                     -4.103030E-08
  ......
  ......
```

The “TYPE” column indicates the coordinate basis for the point listed in the “POINT ID.” column. For instance, “G” designates physical space. The “TYPE” column for other eigenvectors might include “M”, which denotes modal space. The remaining six columns contain the eigenvector (mode shape) coefficients. For each point, two lines of data are provided. The first line contains the real coefficients and the second line contains the imaginary coefficients. Optionally, you can request that the first line contain magnitude of displacement and the second line contain phase angles by specifying the PHASE describer on the DISPLACEMENT or SDISPLACEMENT case control command. Blank spaces indicate that the DOF is part of the non-analysis set DOF, such as the m-set or omitted points. Only analysis set DOF are considered when scaling eigenvectors. Consequently, dependent points for MPC equations and omitted points can be larger or smaller than unity.

## Avoiding Repeated Eigenvalues

None of the complex eigenvalue methods in Simcenter Nastran have provisions for handling repeated eigenvalues. They may return the same eigenvector for repeated eigenvalues. Repeated eigenvalues are usually the result of planes of symmetry. Even if the design intent of a structure includes planes of symmetry, there is never absolute symmetry because of manufacturing imperfections, material imperfections, and so on. You can use this fact to eliminate repeated eigenvalues by perturbing the structure very slightly to destroy absolute symmetry. For example, if a structure contains a plane of symmetry, you can add a small mass to the structure at a location that does not coincide with the plane of symmetry. Doing so is usually sufficient to make repeated eigenvalues distinct and eliminate the associated numerical problems.

## Complex Eigenvalue Analysis Example

### Problem Description

In this example, you use direct complex eigenvalue analysis (SOL 107) to identify whether a simple braking mechanism has divergent modes. The prediction of divergent modes is an indicator that brake squeal might occur. The braking mechanism is illustrated in Figure 5-2. It consists of three bodies:

- A rigid housing.
- A brake pad.
- A moving surface that contacts the brake pad.

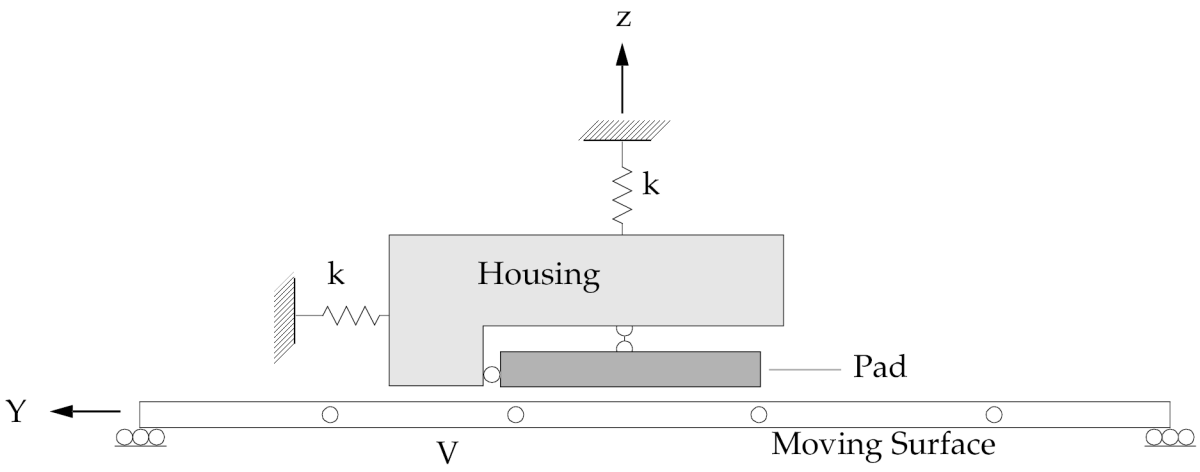



**Figure 5-2. Braking Mechanism**

## Modeling of the Braking Mechanism

In modeling the braking mechanism, the following assumptions are made:

1. The speed of the moving surface is assumed to be much less than the speed of traveling vibrational waves. Therefore, the elements representing the surface can be limited to small motions and the effect of traveling vibrational waves can be ignored.

2. The magnitude of the brake pad vibration may be very small for the onset of a divergent mode.

3. The static preload pressing the moving surface and brake pad together is large enough to maintain full contact between the moving surface and the brake pad.

4. The frictional surfaces are dry and the Coulomb friction model is applicable. The coefficient of friction is constant over the contacting surfaces. Two cases should be considered – one where the coefficient of friction is 0.05 and one where the coefficient of friction is 0.5.

The frictional forces acting on the brake pad and moving surface are proportional to the normal contact forces, which in turn may vary with dynamic response. For small dynamic motions the relationship is:

$$F_{py} = -F_{sy} = \mu N$$

**Equation 5-7.**

where $F_{py}$ is the frictional force exerted on the brake pad, $F_{sy}$ is the frictional force exerted on the moving surface, $\mu$ is the coefficient of friction, and $N$ is the normal force transmitted between the brake pad and the moving surface.

The value of $N$ is only available indirectly. You can obtain an approximation of $N$ by defining the following:

- Coincident pairs of grid points at the contacting surface for the brake pad and moving surface.
- Stiff springs between the coincident pairs of grid points to represent the contact constraint.

You can write the normal force between each pair of coincident grid points as:

$$N = -K(u_{pz} - u_{sz})$$

**Equation 5-8.**

where $u_{pz}$ and $u_{sz}$ are the grid point displacements in the normal direction, and $K$ is the stiffness of the springs.

You can then combine Eq. 5-7 and Eq. 5-8 to obtain:

$$\begin{Bmatrix} F_{py} \\ F_{sy} \end{Bmatrix} = \mu K \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \begin{Bmatrix} u_{pz} \\ u_{sz} \end{Bmatrix} \quad \text{etc.}$$

**Equation 5-9.**

## Modeling the Braking Mechanism in Simcenter Nastran

Because the model is small, there is no benefit in using modal complex eigenvalue analysis. To initiate a direct complex eigenvalue analysis, you must include:

```
SOL 107
```

in the executive control section. To include the frictional relationship given by Eq. 5-9 in the stiffness formulation, use a K2PP direct input stiffness case control command and DMIG bulk entries. Include:

```
K2PP = FRICTN
```

in the case control section to select the DMIG bulk entries named FRICTN, which are given by:

```
DMIG,FRICTN, 0, 1, 1, 0
DMIG,FRICTN, 211,3,, 211,2, 4.0+9,,+C211
+C211, 311,2, -4.0+9
DMIG,FRICTN, 311,3,, 211,2, -4.0+9,,+C311
```

```
+C311, 311,2, 4.0+9
DMIG,FRICTN, 212,3,, 212,2, 4.0+9,,+C212
+C212, 312,2, -4.0+9
DMIG,FRICTN, 312,3,, 212,2, -4.0+9,,+C312
+C312, 312,2, 4.0+9
DMIG,FRICTN, 213,3,, 213,2, 4.0+9,,+C213
+C213, 313,2, -4.0+9
DMIG,FRICTN, 313,3,, 213,2, -4.0+9,,+C313
+C313, 313,2, 4.0+9
DMIG,FRICTN, 214,3,, 214,2, 4.0+9,,+C214
+C214, 314,2, -4.0+9
DMIG,FRICTN, 314,3,, 214,2, -4.0+9,,+C314
+C314, 314,2, 4.0+9
```

In the DMIG bulk entry, after the header row, each successive pair of DMIG bulk entries defines the stiffness relation for a pair of coincident grid points. An arbitrarily large value of $4.0 \times 10^9$ is used for the stiffness of the springs.

Because you will be evaluating the mechanism using a variety of coefficients of friction, the DMIG entry accounts for spring rates only, and does not account for the coefficient of friction in Eq. 5-9. To account for the coefficient of friction, you can define a user parameter named FRIC in the bulk section:

```
PARAM, FRIC, 0.05
```

In the executive control section, you can include the following DMAP alter to multiply the K2PP matrix by the coefficient of friction:

```
ALTER 'MTRXIN'
ADD K2PP, /K2PPX/V,Y,FRIC=(1.0,0.0)
EQUIVX K2PPX/K2PP/-1
ENDALTER
```

The resulting stiffness matrix is unsymmetric, which necessitates the use of complex eigenvalue analysis. To specify the complex eigenvalue method, include a CMETHOD case control command that references an EIGC bulk entry:

```
CMETHOD= 10 $ QZ HESSENBERG
$CMETHOD= 11 $ INVERSE POWER
$CMETHOD= 12 $ BLOCK COMPLEX LANCZOS
$CMETHOD= 13 $ ISRR
```

The input file has provisions for using either the QZ Hessenberg, Inverse Power, Block Complex Lanczos, or ISRR methods as the complex eigenvalue method. Because lines starting with "$" are treated as comments, the input file as presented will use the QZ Hessenberg method.

The EIGC bulk entries selected by the CMETHOD commands are as follows:

```
EIGC, 10, HESS , MAX, , , , 12
$EIGC, 11, INV , MAX, , , , , ,+EINV
$+EINV, 0.0, 10000.0, 0.0, 50000., 20000., 20, 12
$EIGC, 12, CLAN, MAX, , , , , ,+ELAN
$+ELAN, 0.0, 6000.0, , , , , 12
$EIGC, 13, ISRR , MAX, , , , , ,+EISR
$+EISR, 0.0, 6000.0, , , , 1,12
```

Singularities in the mass matrix arise for grids 301 through 332. For the Hessenberg methods only, you must include an OMIT1 bulk entry to eliminate these singularities. When using the Inverse Power, Complex Lanczos, or ISRR methods, a “$” should precede the OMIT1 entry.

```
OMIT1, 4, 301, THRU, 332
```

To simplify the model and eliminate spurious roots, use RBEi and CONM2 elements.

### Simcenter Nastran Input File

The completed input file is as follows:

```
DIAG 12 $ COMPLEX EIGENVALUE METHOD
COMPILE GMA SOUIN=MSCSOU NOLIST NOREF
ALTER 'MTRXIN'
ADD K2PP, /K2PPX/V,Y,FRIC=(1.0,0.0)
EQUIVX K2PPX/K2PP/-1
ENDALTER
SOL 107
DIAG 8 $ MATRIX TRAILERS
TIME 5
CEND
$
TITLE = COMPLEX EIGENVALUE ANALYSIS
K2PP = FRICTN
CMETHOD= 10 $ QZ HESSENBERG
$CMETHOD= 11 $ INVERSE POWER
$CMETHOD= 12 $ BLOCK COMPLEX LANCZOS
$CMETHOD= 13 $ ISRR
MPC = 200
DISP = ALL $ PRINT COMPLEX EIGENVECTORS
SET 1 = 102, THRU, 202
$
BEGIN BULK
PARAM, GRDPNT, 0
PARAM, G, 0.02
PARAM, FRIC, 0.05
$
$ RIGID HOUSING MOUNTED ON SPRINGS, YZ MOTIONS ONLY
GRID,100,,0.0,0.0,35.0,,156
```

```
GRID,101,,0.0,0.0,15.0,,1456
GRID,102,,-50.0,30.0,7.5,,1456
GRID,103,,50.0,30.0,7.5,,1456
RBE2,101,100,23,101,102,103
$
$ LUMPED MASS AT CG
CONM2,100,100,,2.0 , , , , ,+CM1
+CM1,7200.0,,7200.0 , , ,7200.0
$
$ SPRINGS TO GROUND
CELAS2,102, 78.96+6, 100, 2
CELAS2,103, 78.96+6, 100, 3
CELAS2,104, 2.846+11,100,4
$
$ BRAKE PAD IS A SINGLE HEXA WITH A POINT FOR THE HOUSING
CHEXA,200, 200, 211, 212, 214, 213, 221, 222,+CHX1
+CHX1, 224, 223
RBE3,201,,201,123456,1.0,123,221,222,+RBE3
+RBE3,223,224
CONM2,202,201,,0.2
GRID, 201, ,0.0,0.0,15.0
GRID, 211, , -50., -30., 0.0,,1456
GRID, 212, , 50., -30., 0.0,,1456
GRID, 213, , -50., 30., 0.0,,1456
GRID, 214, , 50., 30., 0.0,,1456
GRID, 221, , -50., -30., 15.0,,1456
GRID, 222, , 50., -30., 15.0,,1456
GRID, 223, , -50., 30., 15.0,,1456
GRID, 224, , 50., 30., 15.0,,1456
$
PSOLID, 200, 200, , THREE
MAT1,200,1.0+6,,0.3,1.111-6,,,0.02
$
$ RIGID LINKS FROM PAD FOR TANGENTIAL STOPS
MPC, 200, 213,2, 1.0, 223,2, 1.0, , +MPC21
+MPC21,, 102,2, -2.0
$MPC, 200, 213,3, 1.0, 223,3, 1.0, , +MPC22
$+MPC22,, 102,3, -2.0
MPC, 200, 214,2,1.0, 224,2, 1.0, , +MPC31
+MPC31,, 103,2, -2.0
$MPC, 200, 214,3, 1.0, 224,3, 1.0, , +MPC32
$+MPC32,, 103,3, -2.0
$
$RBE3,202,,102, 234, 1.0, 123, 213 ,223,+RB31
$+RB31,UM ,213,23, 223,2
$RBE3,203,,103, 234, 1.0, 23 ,214, 224,+RB32
$+RB32,UM ,214,23, 224,2
$
$ SPRINGS CONNECTING HOUSING TO CENTER OF PAD
```

```
CELAS2,201, 15.791+6, 201,3 ,101,3
CELAS2,202, 15.791+6, 201,2 ,101,2
$
$ MOVING SURFACE SIMULATED WITH PLATES
CQUAD4,301,300,301,302,304,303
CQUAD4,302,300,303,304,312,311
CQUAD4,303,300,311,312,314,313
CQUAD4,304,300,313,314,322,321
CQUAD4,305,300,321,322,332,331
$
GRID, 301,,-50., -150.0, 0.0,, 12356
GRID, 302,, 50., -150.0, 0.0,, 12356
GRID, 303,,-50., -90.0, 0.0,, 1256
GRID, 304,,50., -90.0, 0.0,, 1256
GRID, 311,,-50., -30.0, 0.0,, 1256
GRID, 312,, 50., -30.0, 0.0,, 1256
GRID, 313,,-50., 30.0, 0.0,, 1256
GRID, 314,, 50., 30.0, 0.0,, 1256
GRID, 321,,-50., 90.0, 0.0,, 1256
GRID, 322,, 50., 90.0, 0.0,, 1256
GRID, 331,,-50., 150.0, 0.0,, 12356
GRID, 332,, 50., 150.0, 0.0,, 12356
$
PSHELL, 300, 300, 15.0, 300,,300
MAT1, 300, 7.106+9, , 0.3, 4.444-6,,, 0.02
$
$ DUMMY SPRINGS TO MEASURE NORMAL FORCES
CELAS2, 311, 4.0+9, 311,3, 211,3
CELAS2, 312, 4.0+9, 312,3, 212,3
CELAS2, 313, 4.0+9, 313,3, 213,3
CELAS2, 314, 4.0+9, 314,3, 214,3
$
DMIG,FRICTN, 0, 1, 1, 0
DMIG,FRICTN, 211,3,, 211,2, 4.0+9,,+C211
+C211, 311,2, -4.0+9
DMIG,FRICTN, 311,3,, 211,2, -4.0+9,,+C311
+C311, 311,2, 4.0+9
DMIG,FRICTN, 212,3,, 212,2, 4.0+9,,+C212
+C212, 312,2, -4.0+9
DMIG,FRICTN, 312,3,, 212,2, -4.0+9,,+C312
+C312, 312,2, 4.0+9
DMIG,FRICTN, 213,3,, 213,2, 4.0+9,,+C213
+C213, 313,2, -4.0+9
DMIG,FRICTN, 313,3,, 213,2, -4.0+9,,+C313
+C313, 313,2, 4.0+9
DMIG,FRICTN, 214,3,, 214,2, 4.0+9,,+C214
+C214, 314,2, -4.0+9
DMIG,FRICTN, 314,3,, 214,2, -4.0+9,,+C314
+C314, 314,2, 4.0+9
```

```
$
EIGC, 10, HESS , MAX, , , , 12
$EIGC, 11, INV , MAX, , , , , ,+EINV
$+EINV, 0.0, 10000.0, 0.0, 50000., 20000., 20, 12
$EIGC, 12, CLAN, MAX, , , , , ,+ELAN
$+ELAN, 0.0, 6000.0, , , , , 12
$EIGC, 13, ISRR , MAX, , , , , ,+EISR
$+EISR, 0.0, 6000.0, , , , 1,12
$
OMIT1, 4, 301, THRU, 332
ENDDATA
```

## Simcenter Nastran Solution Results

The model was run using the:

- QZ Hessenberg method.
- Inverse Power method.
- Block Complex Lanczos method.
- ISRR method

### QZ Hessenberg Method Results:

The first 16 eigenvalues produced by the QZ Hessenberg method are as follows:

```
                C O M P L E X   E I G E N V A L U E   S U M M A R Y
ROOT     EXTRACTION           EIGENVALUE                FREQUENCY
DAMPING
 NO.         ORDER        (REAL)           (IMAG)          (CYCLES)
COEFFICIENT
  1           1      5.812789E+01   -5.747926E+03    9.148109E+02
-2.022569E-02
  2           2     -5.812789E+01    5.747926E+03    9.148109E+02
2.022569E-02
  3           3      6.295126E+01   -6.293827E+03    1.001694E+03
-2.000413E-02
  4           4     -6.295126E+01    6.293827E+03    1.001694E+03
2.000413E-02
  5           5      6.928994E+01   -6.839596E+03    1.088556E+03
-2.026141E-02
  6           6     -6.928994E+01    6.839596E+03    1.088556E+03
2.026141E-02
  7          11      3.305628E+02   -1.654657E+04    2.633468E+03
-3.995546E-02
  8          12     -3.305628E+02    1.654657E+04    2.633468E+03
```

```
3.995546E-02
  9           7      3.348726E+02   -1.751491E+04    2.787584E+03
-3.823858E-02
 10           8     -3.348726E+02    1.751491E+04    2.787584E+03
3.823858E-02
 11           9      3.531269E+02   -1.881514E+04    2.994523E+03
-3.753646E-02
 12          10     -3.531269E+02    1.881514E+04    2.994523E+03
3.753646E-02
 13          13      5.825679E+02   -2.914004E+04    4.637782E+03
-3.998401E-02
 14          14     -5.825679E+02    2.914004E+04    4.637782E+03
3.998401E-02
 15          25      6.813081E+02   -3.414209E+04    5.433882E+03
-3.991016E-02
 16          26     -6.813081E+02    3.414209E+04    5.433882E+03
3.991016E-02
```

**Inverse Power Method Results:**

For the Inverse Power method, only 12 eigenvalues are requested on the EIGC bulk entry. The 12 eigenvalues found by the Inverse Power method are as follows:

```
                C O M P L E X   E I G E N V A L U E   S U M M A R Y
ROOT     EXTRACTION           EIGENVALUE                FREQUENCY
DAMPING
 NO.         ORDER        (REAL)           (IMAG)          (CYCLES)
COEFFICIENT
  1           5     -3.305628E+02    1.654657E+04    2.633468E+03
3.995546E-02
  2           4     -3.327564E+02    1.752627E+04    2.789393E+03
3.797229E-02
  3           3     -3.531269E+02    1.881514E+04    2.994523E+03
3.753646E-02
  4           1     -5.825679E+02    2.914004E+04    4.637782E+03
3.998401E-02
  5           2     -6.813081E+02    3.414209E+04    5.433882E+03
3.991016E-02
  6           6     -9.462816E+02    4.733359E+04    7.533374E+03
3.998352E-02
  7           8     -9.998501E+02    5.001250E+04    7.959736E+03
3.998401E-02
  8           7     -9.727523E+02    5.035683E+04    8.014539E+03
3.863437E-02
  9           9     -1.070261E+03    5.304468E+04    8.442323E+03
4.035319E-02
 10          10     -1.200912E+03    6.031593E+04    9.599579E+03
3.982073E-02
 11          11     -1.312841E+03    6.581738E+04    1.047516E+04
```

```
3.989345E-02
 12          12     -1.569588E+03    7.851400E+04    1.249589E+04
3.998237E-02
```

**Block Complex Lanczos Method Results:**

The first 16 eigenvalues produced by the Block Complex Lanczos method are as follows:

```
                C O M P L E X   E I G E N V A L U E   S U M M A R Y
ROOT     EXTRACTION           EIGENVALUE                FREQUENCY
DAMPING
 NO.        ORDER         (REAL)           (IMAG)          (CYCLES)
COEFFICIENT
  1          31      5.812789E+01   -5.747926E+03    9.148109E+02
-2.022569E-02
  2          32     -5.812789E+01    5.747926E+03    9.148109E+02
2.022569E-02
  3          33      6.295126E+01   -6.293827E+03    1.001694E+03
-2.000413E-02
  4          34     -6.295126E+01    6.293827E+03    1.001694E+03
2.000413E-02
  5          35      6.928994E+01   -6.839596E+03    1.088556E+03
-2.026141E-02
  6          36     -6.928994E+01    6.839596E+03    1.088556E+03
2.026141E-02
  7          49      3.305628E+02   -1.654657E+04    2.633468E+03
-3.995546E-02
  8          50     -3.305628E+02    1.654657E+04    2.633468E+03
3.995546E-02
  9          39      3.348726E+02   -1.751491E+04    2.787584E+03
-3.823858E-02
 10          40     -3.348726E+02    1.751491E+04    2.787584E+03
3.823858E-02
 11          37      3.531269E+02   -1.881514E+04    2.994523E+03
-3.753646E-02
 12          38     -3.531269E+02    1.881514E+04    2.994523E+03
3.753646E-02
 13          47      5.825679E+02   -2.914004E+04    4.637782E+03
-3.998401E-02
 14          48     -5.825679E+02    2.914004E+04    4.637782E+03
3.998401E-02
 15          45      6.813081E+02   -3.414209E+04    5.433882E+03
-3.991016E-02
 16          46     -6.813081E+02    3.414209E+04    5.433882E+03
3.991016E-02
```

**ISRR Method Results:**

The first 16 eigenvalues produced by the ISRR method are as follows:

```
                C O M P L E X   E I G E N V A L U E   S U M M A R Y
ROOT     EXTRACTION          EIGENVALUE                FREQUENCY
DAMPING
 NO.         ORDER        (REAL)           (IMAG)          (CYCLES)
COEFFICIENT
  1           1     -5.812789E+01    5.747926E+03    9.148109E+02
2.022569E-02
  2           2     -6.295126E+01    6.293827E+03    1.001694E+03
2.000413E-02
  3           3     -6.928994E+01    6.839596E+03    1.088556E+03
2.026141E-02
  4           4     -3.305628E+02    1.654657E+04    2.633468E+03
3.995546E-02
  5           5     -3.348726E+02    1.751491E+04    2.787584E+03
3.823858E-02
  6           6     -3.531269E+02    1.881514E+04    2.994523E+03
3.753646E-02
  7           7     -5.825679E+02    2.914004E+04    4.637782E+03
3.998401E-02
  8           8     -6.813081E+02    3.414209E+04    5.433882E+03
3.991016E-02
  9           9     -9.462816E+02    4.733359E+04    7.533374E+03
3.998352E-02
 10          10     -9.998501E+02    5.001250E+04    7.959736E+03
3.998401E-02
 11          11     -9.727523E+02    5.035683E+04    8.014539E+03
3.863437E-02
 12          12     -1.070261E+03    5.304468E+04    8.442323E+03
4.035319E-02
 13          13     -1.200912E+03    6.031593E+04    9.599579E+03
3.982073E-02
 14          14     -1.312841E+03    6.581738E+04    1.047516E+04
3.989345E-02
 15          15     -1.569588E+03    7.851400E+04    1.249589E+04
3.998237E-02
 16          16     -2.754720E+03    1.379062E+05    2.194846E+04
3.995063E-02
```

For this example, the QZ Hessenberg, Block Complex Lanczos, and ISRR methods returned essentially identical results except that the ISRR method has not recognized the plus/minus pairing of eigenvalues. The Inverse Power method has failed to extract some low frequency modes that the other methods found.

The results presented above are for a coefficient of friction of 0.05. The real part of the complex eigenvalues having positive natural frequencies are negative, which indicates that the system is stable. Review of the complete results contained in the .f06 files shows that modes were extracted that have positive real parts, but the frequency of these modes were extreme high ($1 \times 10^{21}$ or more).

Consequently, no divergent modes are predicted within a reasonable range of frequencies when the coefficient of friction is 0.05.

If you rerun the model using a coefficient of friction of 0.5, divergent behavior is observed in modes having much lower frequencies. When a coefficient of friction of 0.5 is used, the first 16 modes extracted by the ISRR method are as follows:

C O M P L E X  E I G E N V A L U E  S U M M A R Y

| ROOT NO. | EXTRACTION ORDER | EIGENVALUE (REAL) | EIGENVALUE (IMAG) | FREQUENCY (CYCLES) | DAMPING COEFFICIENT |
|---|---|---|---|---|---|
| 1 | 1 | -5.806441E+01 | 5.750383E+03 | 9.152018E+02 | 2.019497E-02 |
| 2 | 2 | -6.294889E+01 | 6.293917E+03 | 1.001708E+03 | 2.000309E-02 |
| 3 | 3 | -6.910709E+01 | 6.844853E+03 | 1.089392E+03 | 2.019243E-02 |
| 4 | 4 | -3.300979E+02 | 1.667092E+04 | 2.653260E+03 | 3.960164E-02 |
| 5 | 5 | -3.565693E+02 | 1.823559E+04 | 2.902285E+03 | 3.910695E-02 |
| 6 | 6 | -3.308914E+02 | 1.887382E+04 | 3.003861E+03 | 3.506354E-02 |
| 7 | 7 | -5.825679E+02 | 2.914004E+04 | 4.637782E+03 | 3.998401E-02 |
| 8 | 8 | -6.834198E+02 | 3.404017E+04 | 5.417662E+03 | 4.015372E-02 |
| 9 | 9 | -9.458578E+02 | 4.735407E+04 | 7.536633E+03 | 3.994833E-02 |
| 10 | 10 | -9.998501E+02 | 5.001250E+04 | 7.959736E+03 | 3.998401E-02 |
| 11 | 11 | 3.294983E+03 | 5.022236E+04 | 7.993137E+03 | -1.312158E-01 |
| 12 | 12 | -5.339673E+03 | 5.010448E+04 | 7.974375E+03 | 2.131415E-01 |
| 13 | 13 | -1.200202E+03 | 5.960435E+04 | 9.486327E+03 | 4.027230E-02 |
| 14 | 14 | -1.316790E+03 | 6.545470E+04 | 1.041744E+04 | 4.023517E-02 |
| 15 | 15 | -1.569608E+03 | 7.851716E+04 | 1.249639E+04 | 3.998128E-02 |
| 16 | 16 | -2.754750E+03 | 1.379203E+05 | 2.195070E+04 | 3.994698E-02 |

Notice that the real part of Mode 11 is positive and that Mode 11 occurs around 8000 Hz. Apparently, the stability of the mechanism bifurcates at a coefficient of friction between 0.05 and 0.5. You can obtain the precise threshold by iterating on the value for the coefficient of friction until the real part of Mode 11 is sufficiently close to zero.

To determine which component or components are associated with Mode 11, you can view the mode shapes using a post-processor. You can also identify whether Mode 11 is a bending mode or a twisting mode, among others. With this information, you can redesign the component to either eliminate the divergent behavior or move it to a much higher frequency.

## 5.4 Nonlinear Transient Response Analysis

Nonlinear effects in structures occur mainly due to nonlinear materials, preloads, and large rotations. Contact problems exhibit nonlinear effects due to changes in boundary conditions. All of these effects may be represented by nonlinear elements.

The nonlinear element library of Simcenter Nastran consists of:

- RODs, CONRODs, and TUBEs for unidirectional truss members.
- BEAMs for axially and laterally deforming line members.
- QUAD4s and TRIA3s for membrane, plate and shell modeling.
- HEXAs, PENTAs, and TETRAs for solid modeling.
- GAPs for contact and friction simulation.
- BCONPs for slideline contact.

Nonlinear elements may be combined with linear elements for computational efficiency if the nonlinear effects can be localized. Primary operations for nonlinear elements are the updating of element coordinates and applied loads for large displacements and the internal relaxation iteration for material nonlinearity. Refer to Table 5-2 for a summary of the nonlinear element properties.

**Table 5-2. Summary of Nonlinear Elements**

| Element | Connectivity |
|---|---|
| BCONP | Connection and type for slideline. |
| CBEAM | Connection for beam element. |
| CGAP | Connection for gap or frictional element. |
| CHEXA | Connection for six-sided solid element. |
| CONROD | Connection and properties for rod. |
| CPENTA | Connection for five-sided solid element. |
| CPYRAM | Connection for five-sided solid element. |
| CQUAD4 | Connection for quadrilateral element with bending and membrane stiffness. |
| CROD | Connection for rod with axial and torsional stiffness. |
| CTETRA | Connection for four-sided solid element. |

| Element | Connectivity |
|---|---|
| CTRIA3 | Connection for triangular element with bending and membrane stiffness. |
| CTUBE | Connection for a tube. |
| **Element** | **Properties** |
| PBCOMP | Properties for composite CBEAM. |
| PBEAM | Properties for CBEAM. |
| PCOMP | Properties for composite material laminate. |
| PGAP | Properties for CGAP. |
| PLSOLID | Hyperelastic properties for CHEXA, CPENTA, and CTETRA. |
| PROD | Properties for CROD. |
| PSHELL | Properties for CTRIA3 and CQUAD4. |
| PSOLID | Properties for CHEXA, CPENTA, CPYRAM, and CTETRA. |
| PTUBE | Properties for CTUBE. |

The geometric nonlinearity becomes discernible when the structure is subjected to large displacement and rotation. Geometric nonlinear effects are prominent in two different aspects: (1) geometric stiffening due to initial displacements and stresses, and (2) follower forces due to a change in loads as a function of displacements. These effects are included in the current release of Simcenter Nastran, but the large deformation effect resulting in large strains is yet to be implemented.

Material nonlinearity is an inherent property of any engineering material. Material nonlinear effects may be classified into many categories. Included in the current release of Simcenter Nastran are plasticity, nonlinear elasticity, creep, and viscoelasticity. Creep and viscoelasticity, implemented as a generalized nonlinear viscoelastic capability, may be coupled with plasticity. Many sophisticated options are available for yield criteria and hardening behavior in plasticity.

The primary solution operations are load and time steps, iterations with convergence tests for acceptable equilibrium error, and stiffness matrix updates. The iterative process is based on the modified-Newton's method combined with optional methods such as the quasi-Newton (BFGS) update and the line search. The tangent matrix updates are performed automatically to improve the computational efficiency, and may be overridden at the user's discretion.

For static analysis, a number of options such as arc-length methods is available for snapthrough or post-buckling analysis. For the transient response analysis, a number of options is available for implicit direct time integration, combined with adaptive and other iteration strategies similar to those implemented for static analysis.

The adaptive method is implemented using the two-point recurrence (or one-step) formula as its foundation. The optimum time step size, which is required for accuracy and efficiency, changes continuously in the transient dynamic environment. The primary concept of automatic time step adjustment is that the proper size of the time step can be predicted based on the dominant frequency in the incremental deformation pattern at the previous time step. This concept presents a deficiency of time lag involved in the prediction process. Furthermore, changes in nonlinearity cannot be predicted from the deformation pattern at the previous time step. This deficiency is overcome by the bisection process, which is activated when any difficulties arise in terms of convergence during the iteration.

## User Interface

The input data required for SOL 129 is a combination of direct time integration control data, similar to SOL 109 (for direct linear transient with superelements), and nonlinear modeling data similar to SOL 106 (for nonlinear statics). The nonlinear properties and/or effects are defined by nonlinear material data (MATS1 and TABLES1), gap elements (GAP) for nonlinear interfaces, and PARAM,LGDISP for geometric nonlinearity. The transient effects are produced by time-dependent loading functions (TLOADi, DAREA, LSEQ, etc.), damping (parameters, elements and material data), and mass properties.

The unique data required for SOL 129 is supplied on the TSTEPNL Bulk Data entry. The TSTEPNL entry in itself is a combination of the TSTEP entry for direct time integration and the NLPARM entry for nonlinear iteration control. Restarts are controlled by parameters (LOOPID, STIME, SLOOPID and SDATA) which can be specified either in the Case Control Section or Bulk Data Section. Some optional parameters (TSTATIC, NDAMP) are provided for additional control or capabilities.

## Case Control

Each subcase defines a time interval starting from the last time step of the previous subcase, subdivided into smaller time steps. The output time is labeled by the cumulative time, including all previous subcases. There are advantages in dividing the total duration of analysis into many subcases so that each subcase does not have excessive number of time steps. The data blocks containing solutions are generated at the end of each subcase for storage in the database for output processing and restarts. As such, converged solutions are apt to be saved at many intermediate steps in case of divergence and more flexible control becomes possible with multiple subcases.

The input loading functions may be changed for each subcase or continued by repeating the same DLOAD request. However, it is recommended that one use the same TLOAD Bulk Data for all the subcases in order to maintain the continuity between subcases, because TLOADi data defines the loading history as a function of cumulative time. Static loads (PLOADi, FORCEi, MOMENTi) may be associated with time-dependent functions by the Bulk Data LSEQ, which can be selected by a Case Control command LOADSET. However, no thermal loads or enforced displacements (SPCD) are allowed in the nonlinear transient analysis. Nonlinear forces as functions of displacements or velocities (NOLINi) may be selected and printed by the Case Control commands NONLINEAR and NLLOAD, respectively. Each subcase may have a different time step size, time interval, and iteration control selected by the TSTEPNL request. The Case Control requests that may not be changed after the first subcase are SPC, MPC, DMIG, and TF.

Output requests for each subcase are processed independently. Requested output quantities for all the subcases are appended after the computational process for actual output operation. Available outputs are DISPLACEMENT, VELOCITY, ACCELERATION, OLOAD, STRESS, FORCE, SDISPLACEMENT, SVELOCITY, SACCELERATION, NONLINEAR (NLLOAD), and SPCFORCE. However, element force output and GRID point stresses are not available for nonlinear elements.

Initial conditions (displacement or velocity) can be specified by the Bulk Data input, TIC, selectable by the Case Control command IC. If initial conditions are given, all of the nonlinear element forces and stresses must be computed to satisfy equilibrium with the prescribed initial displacements. On the other hand, initial conditions can be generated by applying static analysis for the preload using

PARAM,TSTATIC in the first subcase. Then the transient analysis can be performed in the ensuing subcases. Associated with the adaptive time stepping method, the PARAM,NDAMP is used to control the stability in the ADAPT method. The parameter NDAMP represents the numerical damping (a recommended value for usual cases is 0.01), which is often required to improve the stability and convergence in contact problems.

All the superelement model generation options and matrix reduction options are allowed for the linear portion of the structure. Component Mode Synthesis and Guyan reduction may be performed for upstream superelements. The residual superelement may contain scalar degrees-of-freedom representing linear modal formulations.

## Implicit Integration Control: TSTEPNL Data

The input fields of the TSTEPNL Bulk Data entry specify the time step size, the number of steps, and the output interval as well as the nonlinear iteration options. The TSTEPNL Bulk Data is selectable by the Case Control command TSTEPNL. Although the same TSTEPNL Bulk Data may be selected by more than one subcase, it is recommended to select a different TSTEPNL entry for each subcase in preparation for changes in the restarts.

The choice of time step size is determined primarily by the frequency content of the input load and the highest frequency mode of interest. A general guideline is that seven or more steps per cycle be provided for reasonable accuracy. Modes with shorter periods (higher frequency) will be attenuated by the numerical process. Highly nonlinear problems may require smaller step size. However, the adaptive time stepping capability will automatically adjust the time step size. Nevertheless, the initial time step size should be estimated by the user according to the aforementioned principles.

A caution is necessary in using the automatic time step adjustment if the forcing function is a short duration pulse. Since the automatic time step adjustment algorithm does not consider the loading history, short pulses could be skipped if the time step is automatically adjusted to a larger value than the pulse duration. It is advised that a drastic change in the time step size between subcases be avoided. A drastic change—e.g., ratio of the two time steps exceeding 1000—could cause a loss of accuracy upon subcase switch. In this case, an intermediate subcase should be provided for a transition period of short interval to reduce the ratio.

The parameters for controlling the accuracy and stability of the incremental and iterative process can be specified in the TSTEPNL Bulk Data entry. The controls are applicable to the automatic time step adjustment and bisection process in addition to stiffness matrix updates, BFGS updates, and line searches similar to those on the NLPARM Bulk Data entry. Since default values have been selected from numerous test runs, the analysis should be started with the default setting and changed if necessary. The TSTEPNL data format is shown below with default values:

### Format:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| TSTEPNL | ID | NDT | DT | NO | METHOD | KSTEP | MAXITER | CONV | +TNL1 |
| +TNL1 | EPSU | EPSP | EPSW | MAXDIV | MAXQN | MAXLS | FSTRESS | | +TNL2 |
| +TNL2 | MAXBIS | ADJUST | MSTEP | RB | MAXR | UTOL | RTOLB | | |

## Example with Defaults (x is Required Input):

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| TSTEPNL | x | x | x | 1 | ADAPT | 2 | 10 | PW | +TNL1 |
| +TNL1 | 1.E-2 | 1.E-3 | 1.E-6 | 2 | 10 | 2 | 0.2 | | +TNL2 |
| +TNL2 | 5 | 5 | 0 | 0.75 | 16.0 | 0.1 | 20. | | |

The TSTEPNL Bulk Data entry is selected using ID via the Case Control command TSTEPNL. Each subcase (residual superelement solutions only) requires a TSTEPNL entry. Multiple subcases are assumed to occur sequentially in time. Therefore, the initial conditions of each subcase are defined by the end conditions of the previous subcase.

The NDT field specifies the number of time steps with DT as the size of each time step. The total duration for the subcase can be assessed by multiplying NDT and DT (i.e., NDT*DT). The time increment ($\Delta t$) remains constant during the analysis in AUTO and TSTEP options, and is equal to DT. However, the time increment ($\Delta t$) changes during the analysis in the ADAPT option and the actual number of time steps will not be equal to NDT. In the ADAPT option, DT is used as an initial value for $\Delta t$.

The NO field specifies the time step interval for output; i.e., every NO-th step solution is saved for output. The data will be output at steps 0, NO, 2NO, ..., etc., and the last converged step for printing and plotting purposes. The Case Control command OTIME may also be used to control the output points.

The METHOD field selects an option for direct time integration and the stiffness matrix update strategies among ADAPT, AUTO and TSTEP.

- If you select the AUTO option, Simcenter Nastran automatically updates the stiffness matrix to improve convergence while the KSTEP value is ignored.
- If you select the TSTEP option, Simcenter Nastran updates the stiffness matrix every KSTEP-th increment of time.
- If you select the ADAPT option, Simcenter Nastran automatically adjusts the incremental time and uses the bisection algorithm in case of divergence.

During the bisection process in the ADAPT option, stiffness is updated at every KSTEP-th successful bisection. The ADAPT method allows linear transient analysis, but AUTO or TSTEP will abort the run if the model does not have any data representing nonlinearity. The stiffness matrix is always updated for a new subcase or restart, irrespective of the option selected.

The number of iterations for a time step is limited to MAXITER. If the solution does not converge in MAXITER iterations, the process is treated as a divergent process; i.e., either a bisection or stiffness matrix update takes place based on the value of MAXBIS. The sign of MAXITER provides a control over reiteration in case of failure in convergence or bisection. If MAXITER is negative, the analysis is terminated when the divergence condition is encountered twice during the same time step or the solution diverges for five consecutive time steps. If MAXITER is positive, Simcenter Nastran computes the best attainable solution and continues the analysis.

The convergence test is controlled by convergence test flags (U for displacement error test, P for load equilibrium error test, W for work error test) and the error tolerances (EPSU, EPSP and EPSW) which define the convergence criteria. All requested criteria (a combination of U, P, and/or W) are satisfied upon convergence. It should be noted that at least two iterations are necessary to check the displacement convergence criterion.

The MAXDIV field provides control over diverging solutions. Depending on the rate of divergence, the number of diverging solutions (NDIV) is incremented by 1 or 2. The solution is assumed to be divergent when NDIV reaches MAXDIV during the iteration. If the bisection option is used with the ADAPT method, the time step is bisected upon divergence. Otherwise, the solution for the time step is repeated with a new stiffness based on the converged state at the beginning of the time step. If NDIV reaches MAXDIV twice within the same time step, the analysis is terminated with a fatal message.

In transient analysis, the BFGS quasi-Newton updates and the line search process work in the same way as in static analysis (except for the default settings). The MAXQN field defines the maximum number of quasi-Newton vectors to be saved on the database and the MAXLS defines the number of line searches allowed per iteration. Nonzero values of MAXQN and MAXLS activate the quasi-Newton update and the line search process, respectively.

The FSTRESS field defines a fraction of the effective stress ($\bar{\sigma}$) which is used to limit the subincrement size in the material routine. The number of subincrements in the material routines is determined such that the subincrement size is approximately *FSTRESS* $\cdot \bar{\sigma}$ (equivalent stress). FSTRESS is also used to establish a tolerance for error correction in the elasto-plastic material, i.e.,

$$Error \text{ in yield function} < FSTRESS \cdot \bar{\sigma}$$

If the limit is exceeded at the converging state, Simcenter Nastran exits with a fatal error message. Otherwise, the stress state is adjusted to the current yield surface.

In the ADAPT method, MAXBIS is the maximum number of bisections allowed for each time step ($-9 \le$ MAXBIS $\le 9$). The bisection process is activated when divergence occurs and MAXBIS $\neq 0$. The number of bisections for a time increment is limited to MAXBIS. If MAXBIS is positive and the solution does not converge after MAXBIS bisections, the best solution is computed and the analysis is continued to the next time step. If MAXBIS is negative and the solution does not converge in MAXBIS bisections, the analysis is terminated.

The parameter ADJUST allows the user to control the automatic time stepping in the ADAPT option. A value of zero for ADJUST turns off the automatic adjustment completely. If ADJUST is positive, the time increment is continually adjusted for the first few steps until a good value of $\Delta t$is obtained. After this initial adjustment, the time increment is adjusted every ADJUST time steps only. A value of ADJUST an order greater than NDT will turn off adjustments after the initial adjustment. Since the automatic time step adjustment is based on the mode of response and not on the loading pattern, it may be necessary to limit the adjustable step size when the period of the forcing function is much shorter than the period of dominant response frequency of the structure. It is the user's responsibility to ensure that the loading history is properly traced with the ADJUST option. The ADJUST option should be suppressed for the duration of a short pulse loading. If unsure, the user should start with a value of DT that is much smaller than the pulse duration in order to properly represent the loading pattern.

MSTEP defines the desired number of time steps to obtain the dominant period response accurately (10 ≤ Integer ≤ 200). RB defines bounds for maintaining the same time step for the stepping function in the automatic time step adjustment method (0.0 ≤ Real ≤ 1.0). Parameters MSTEP and RB are used to adjust the time increment during the analysis in the ADAPT option. The adjustment is based on the number of time steps desired to capture the dominant frequency response accurately. The time increment is adjusted as

$$\Delta t_{n+1} = f(r)\Delta t_n$$

**Equation 5-10.**

where:

$$r = r = \frac{1}{MSTEP}\left(\frac{2\pi}{\omega_n}\right)\left(\frac{1}{\Delta t_n}\right)$$

with:

$f$ = 0.25 for $r < 0.5 \cdot RB$

$f$ = 0.5 for $0.5 \cdot RB \le r < 2$

$f$ = 1.0 for $RB \le r < 2$

$f$ = 2.0 for $2. \le r < 3. / RB$

$f$ = 4.0 for $r \le 3. / RB$

The recommended value of MSTEP for nearly linear problems is 20. A larger value (e.g., 40) is required for highly nonlinear problems. In the default options, Simcenter Nastran automatically computes the value of MSTEP based on the changes in the stiffness.

The MAXR field defines the maximum ratio for the adjusted incremental time relative to DT allowed for time step adjustment (1.0 ≤ Real ≤ 32.0). MAXR is used to define the upper and lower bounds for adjusted time step size, i.e.,

$$MIN\left(\frac{DT}{2^{MAXBIS}}, \frac{DT}{MAXR}\right) \le \Delta t \le MAXR \cdot DT$$

**Equation 5-11.**

The UTOL field defines the tolerance on displacement increment below which there is no time step adjustment (0.001 < Real ≤ 1.0). UTOL is used to filter undesirable time step adjustment; i.e., no time step adjustment is performed if

$$\frac{\|\dot{U}_n\|}{\|\dot{U}\|_{max}} < UTOL$$

**Equation 5-12.**

The RTOLB field defines the maximum value of incremental rotation (in degrees) allowed per iteration to activate bisection (Real > 2.0). The bisection is activated if the incremental rotation for any degree-of-freedom ($\Delta\theta_x$, $\Delta\theta_y$, $\Delta\theta_z$), or exceeds the value specified for RTOLB. This bisection strategy based on the incremental rotation is controlled by the MAXBIS field.

## Iteration Related Output Data

During the incremental and iterative computation, the process information consisting of iteration data is printed at the end of each iteration or time step. The data is printed under the following headings:

| | |
|---|---|
| **TIME** | Cumulative time for the duration of the analysis |
| **ITER** | Iteration count for each time step |
| **DISP** | Relative error in terms of displacements |
| **LOAD** | Relative error in terms of load vectors |
| **WORK** | Relative error in terms of work |
| **LAMBDA(I)** | Rate of convergence in iteration |
| **LAMBDA(T)** | Ratio of the load error for two consecutive time steps |
| **LAMBDA-BAR** | Average of LAMBDA(T) over the last three steps, computed only for AUTO or TSTEP method |
| **DLMAG** | Absolute norm of the residual error vector, $\|R\|$. The absolute convergence is defined using DLMAG by $\|R\| < 10^{-12}$ |
| **FACTOR** | Final value of the line search parameter |
| **E-FIRST** | Divergence rate error before line search |
| **E-FINAL** | Error at the end of line search |
| **NQNV** | Number of quasi-Newton vectors appended |
| **NLS** | Number of line searches performed during the iteration |

| | |
|---|---|
| **ITR DIV** | Number of occurrences of divergence detected during the adaptive iteration by the iteration module NLTRD2 |
| **MAT DIV** | Number of occurrences of bisection conditions in the material routine (excessive stress increment) or in the rotation angle (excessive rotation) during the iteration using the ADAPT method |
| **NO. BIS** | Number of bisections executed for the current time interval |
| **ADJUST** | Ratio of time step adjustment relative to DT within a subcase |

## Three-Point Method (NLTRD Module)

The three-point method is chosen to be compatible with the Simcenter Nastran linear transient integration method. The Newmark Beta method based on the three-step integration is combined with the modifications to Newton's method for nonlinear solutions. The additional iteration steps provide equilibrium solutions at each time step, thereby guaranteeing stability and accuracy for arbitrary time step size. This method can be used by selecting the AUTO or TSTEP method in the TSTEPNL Bulk Data entry.

### Basic Equations

We may calculate the load equilibrium error vector, $\{R_n\}$ at time step $n$ by the equation

$$\{R_n\} = \{\bar{P}_n - M\ddot{u}_n - B\dot{u}_n - \bar{F}_n\}$$

**Equation 5-13.**

where:

$\{\bar{P}_n\}$ = Average load over the time period ($t_{n-1} < t_n < t_{n+1}$).

$\{\ddot{u}_n\}$, $\{\dot{u}_n\}$ = Corresponding acceleration and velocity vectors. $\dot{u}_n$

$\{\bar{F}_n\}$ = Average elasto-plastic element total force vector.($\bar{F}_n = K\bar{u}$ for linear problems.)

The above equation is solved at the reduced ($u_d$) displacement vector size. The approximation errors due to dynamic reduction methods are not included in the error vector $\{R_n\}$. Applying Newmark's method over a finite time period, $t_{n-1} < t < t_{n+1}$, the average static forces are

$$\{\bar{F}\} = \{\beta F(u_{n+1}) + (1-2\beta)F(u_n) + \beta F(u_{n-1})\}$$

**Equation 5-14.**

where β is the Newmark Beta operator and $F(u_n)$ is the nonlinear force due to a generalized displacement vector $\{u_n$. An identical definition occurs for $\{\bar{P}\}$ from the applied loads at each time step.

In summary, at each time step Simcenter Nastran iterates the displacements and forces until $\{R^i_n\}$ passes the convergence tests or the number of passes reaches an iteration limit. With a single step, i = i, calculating only $R_0$, the results will be identical to the existing NOLINi results in Simcenter Nastran. For faster convergence the iterations may continue, the matrices may be updated, and/or the time step size may be reduced. For more details on this algorithm, refer to Chapter 9 of the *Simcenter Nastran Nonlinear Handbook*.

## Two-Point Method (NLTRD2 Module)

The multistep implicit integration method has difficulties when changing time step size. Therefore, to allow self-adapting time step sizes, the two-point integration method is introduced with module NLTRD2. This method can be selected by specifying the ADAPT method in the TSTEPNL Bulk Data entry.

### Newmark Integration

For the adaptive scheme, Newmark's method is employed with the two-point recurrence formula for one-step integration, i.e.,

$$\{U_{n+1}\} = \{U_n\} + \Delta t\{\dot{U}_n\} + \frac{1}{2}\Delta t^2\{\ddot{U}_n\} + \beta\Delta t^2\{\ddot{U}_{n+1} - \ddot{U}_n\}$$

**Equation 5-15.**

and

$$\{\dot{U}_{n+1}\} = \{\dot{U}_n\} + \Delta t\{\ddot{U}_n\} + \gamma\Delta t\{\ddot{U}_{n+1} - \ddot{U}_n\}$$

**Equation 5-16.**

where $\{U\}$, $\{\dot{U}\}$, $\{\ddot{U}\}$ and $\Delta t$ denote displacement, velocity, acceleration and the time step increment, respectively. The subscript $n$ designates the time step and the parameters (β and γ) are to be selected for the best solution. An equilibrium equation to be satisfied at time step ($n$ + 1) is

$$M\{\ddot{U}_{n+1}\} + C\{\dot{U}_{n+1}\} + \{F(U_{n+1})\} = \{P_{n+1}\}$$

**Equation 5-17.**

where $M$ and $C$ denote mass and damping matrices, and $\{F\}$ and $\{P_{n+1}\}$ denote internal and external forces, respectively.

An alternative expression for the load vector can be derived for $\gamma = 0.5$ by introducing

$$M\{\ddot{U}_n\} + C\{\dot{U}_n\} = \{P_n - F_n\}$$

**Equation 5-18.**

By virtue of **Eq. 5-18** the residual load error at each time step is effectively carried over to the next step and the error propagation is reduced, while the computation is significantly simplified.

$$\{R^i_{n+1}\} = \{P_{n+1} - F^i_{n+1}\} + \frac{4}{\Delta t}M\{\dot{U}_n\} + \{P_n - F_n\}$$
$$- \left[\frac{4}{\Delta t^2}M + \frac{2}{\Delta t}C\right]\{U^i_{n+1} - U_n\}$$

**Equation 5-19.**

The iteration method calculates new values of $R^i_{n+1}$ until it is sufficiently small. The results then become the starting point for the next time step.

## Adaptive Time Stepping

It is desirable to have a fully automated method of time integration that renders effective and accurate solutions of nonlinear problems. The dynamic response characteristics of the structure may change due to the nonlinearity (geometric or material) or the mode of excitation. When the type of nonlinear behavior changes, the time step size should be adjusted. The drawback is that in implicit methods, a change in the time step size requires a decomposition of the tangent matrix. After the solution is achieved for the original time step, the usual procedure will be resumed for the next time step.

In the present implementation of the automatic time stepping for nonlinear applications, the adjustment is designed to be adaptive to the severity of the nonlinearity in the problem by monitoring the changes in the stiffness. However, difficulties in the automatic time stepping have been discovered when plasticity or GAP elements are involved. In such cases an undesired time step adjustment can be caused by a drastic change in the stiffness, which may only be a temporary difficulty. A filtering scheme has been devised to suppress the effects of the spurious mass or stiffness in the automatic time step adjustments. In addition, the bisection method is chosen to be activated only at the time when

divergence occurs. The bisection process is coordinated with the stiffness matrix update so that changes in nonlinearities are properly reflected while the bisection is in progress.

After the time step is completely converged, the integration proceeds to the next time step with the same increment (*Atk*). Further bisection may be required after achieving converged solution at the intermediate time steps.

If no further bisection is required and the solutions converge without any difficulty, some recovery process such as doubling time step may be activated. However, any adjustment to the time step would require additional matrix decomposition that offsets the advantages of adjusting time step size. Therefore, it has been determined to continue stepping through the specified time interval to completion. The normal process should be resumed when the user-specified time step at which bisection is activated is fully processed.

When the automatic time stepping is combined with the bisection method and if the bisection is activated, the automatic adjustment procedure will be deferred until the divergence is trapped and the solution process is stabilized. The solution process is considered stabilized when two successive solutions have converged without requiring further bisection. This is the time when the automatic time step adjustment is resumed.

The maximum number of bisections is limited by a user-specified parameter MAXBIS (defaulted to 5). The bisection process is activated on an as needed basis. Users also have an option to suppress bisection by specifying MAXBIS = 0. If the bisection is required more than MAXBIS times, the solution process will continue without bisection by activating the reiteration procedure, by which the same iteration process is repeated to find the best attainable solution. The best attainable solution is a solution obtained from the iteration that yields the least average error. The average error is defined by averaging relative errors in displacements, loads and energy. If the i-th iteration yields the least average error, the reiteration procedure will end at the i-th iteration and the normal time stepping procedure will be resumed.

## Quasi-Newton and Line Searches

The BFGS update and the line search processes are fully operational during the iteration. Quasi-Newton (QN) vectors are continuously accumulated up to MAXQN pairs (MAXQN is a user-specified parameter) until the new stiffness matrix is evaluated. Once the number of QN vector pairs reaches MAXQN (defaulted to 10), the QN vectors will be updated selectively based on the condition number of the QN update matrix. As a requirement for the reiteration procedure, the QN vectors to be accumulated are stored on a temporary basis until a converged solution is obtained. All the accumulated QN vectors are purged if the stiffness matrix update or the decomposition (due to the change in $\Delta t$) is scheduled.

The iterative process for time step ($n$ + 1) can be summarized as follows:

1. Given at the beginning of the time step are $\Delta t_n$, $\omega_{ref}$, [$M$], [$C$], {$P_n$}, {$\Delta F_n$}, {$F_n$}, {$U_n$}, {$\dot{U}_n$}, and the triangularized matrices ($LDL^T$) for $[A] = \left[\frac{4}{\Delta t^2}M + \frac{2}{\Delta t}C + K\right]$.

2. Adjust the time step size, if applicable.

3. Initialize $\{U^0_{n+1}\}$ , $\{F^0_{n+1}\}$, and $\{P^0_{n+1}\}$.

4. Decompose, $[A] = LKL^T$, if $\Delta t_{n+1} \neq \Delta t_n$. Then, purge the QN vector file.

5. Compute $\{R^i_{n+1}\}$.

6. Solve for $\{\Delta U^{i+1}\}$ by forward and backward substitution, using the BFGS update if applicable.

7. Compute the line search error.

8. If divergence occurs, go to step 12.

9. Proceed to the next step if the line search error is less than the tolerance. Otherwise, perform the line search process and go back to step 7.

10. Compute the global error function and check convergence.

11. If the solution has converged, go to step 16. Otherwise, save the QN vector on a temporary basis (if applicable) and go to the next iteration in step 5.

12. If it is the first divergence or the divergence after the KSTEP-th converged bisection step, update the stiffness matrix, and go back to step 4. Otherwise, proceed to the next step.

13. Bisect $\Delta t$ and go back to step 3. However, if the maximum number of bisections allowed for the time step is reached, proceed to the next step.

14. Go to step 5 to continue iteration. However, if the maximum number of iterations allowed for an increment is reached, proceed to the next step.

15. Perform the reiteration procedure to find the best attainable solution.

16. Accept the solution ($U_n + 1$ and ($F_n + 1$) and append temporary QN vectors to the permanent QN vector file.

17. Compute the velocity $\dot{U}_{n+1}$.

18. Advance to the next time step with $n = n + 1$.

## Restarts

Since SOL 106 and SOL 129 share the same database storage formats for nonlinear tables and matrices, the restart system for transient analysis can use either a previous static or transient nonlinear analysis as its initial conditions.

## Restarting From SOL 106 Into SOL 129

The options for a restart from SOL 106 into SOL 129 are static to static and static to dynamic analysis. For a restart from a previous static analysis, only the first subcase is affected. Simply provide a database created in SOL 106 and specify the parameter

```
PARAM,SLOOPID,N
```

where N is the printed value of LOOPID for the desired static solution. The initial transient load should be identical to static loads at the restart state. Constraint sets, direct input matrices, mass, and damping may be changed.

## Restarting Within SOL 129

Restarting within SOL 129 allows static to static, static to dynamic, and dynamic to dynamic analysis. Restarts from a previous nonlinear transient execution are available for a number of cases. If the same model is to be reexecuted, only the residual superelement needs to be reassembled. If the final results from the previous transient run are to be used as the initial conditions at t = 0, add dummy SUBCASE commands, corresponding to the previous run, to start the residual Case Control execution and set the parameter, STIME = 0.

The normal restart for a transient run is to be continued from the last step of a previous subcase with different loads and/or TSTEPNL data. For the normal restart provide the following parameters:

```
LOOPID = N:  Start from the N-th subcase
STIME = t:  Start from time t
```

Note that to avoid incompatible matrix sizes, constraint sets must not be changed. The values of LOOPID and STIME, which are printed with the iteration information for each subcase, can be directly read from the printout of the previous run. See the *Simcenter Nastran Nonlinear Handbook* for more details and some examples.

$$t = \sum_{i=1}^{N} NDT_i \cdot DT_i$$

**Equation 5-20.**

where $NDT_i$ and $DT_i$ are the number of time steps and the time increment of the i-th subcase, respectively.

If a SOL 129 run is terminated abnormally in the middle of a subcase, it may or may not be restartable depending upon the cause of the abnormal exit. If the job is stopped due to a diverging solution, it can be restarted either from the end of a previous subcase or from the last saved solution step. The restart

procedure for the former is identical to that for the normal restart as described in the preceding paragraph. The latter case also requires parameters LOOPID and STIME; however, the input value for STIME differs depending on the value of METHOD specified on the TSTEPNL entry.

If METHOD = AUTO or TSTEP, STIME is the time corresponding to the last output step which may be calculated based on the output skip factor (i.e., the NO on the TSTEPNL entry). If METHOD = ADAPT, the last converged solution is always treated as an output step and is always saved for the restart so that STIME can be the time of the last converged step. The values of STIME and LOOPID can also be found in the printout, if the ADAPT method is used.

Once STIME and LOOPID are known, determine the number of remaining time steps in the subcase and create a new TSTEPNL entry for the remaining time. Insert a new subcase that references the new TSTEPNL entry prior to the remaining subcases in the Case Control Section.

A solution may be terminated in the middle of a subcase due to insufficient CPU time: (1) the CPU time specified in the Executive Control Section is insufficient so the run is forced to exit by Simcenter Nastran, or (2) the CPU time exceeds the limit specified in the computer system which leads to a sudden job abortion by the system. In the first case, Simcenter Nastran is able to detect the specified CPU time in the Executive Control Section and automatically activate the exit procedure before time expiration. When completed, the solution can be restarted from the termination point as in the solution diverging case. In the second case, the solution can only be restarted from the end of a subcase.

Restarts may also be performed solely for data recovery by providing the following parameters:

```
SDATA
= -1 : Recover data without running the solution module LOOPID = N : from
the 1st through the N-th subcases
```

Note that solution sets DISP, VELO, ACCE, OLOAD, SPCF (printout and plotting) and NLLOAD (plotting only) are recoverable while STRESS, SDISP, SVELO and SACCE sets are not.

## Expedient Iteration Strategies

The dynamic tangent matrix may change for the following reasons:

1. Stiffness (K) changes due to geometric, material, and/or kinematic nonlinearity.

2. Bisection or time-step adjustment causes changes in the effective stiffness.

While the second case only requires decomposition, the first case requires reformulation of the new stiffness matrix as well. The adaptive method is based on the following matrix update strategies:

1. Stiffness is updated at the previously converged position when the divergence is detected for the first time at a given time step.

2. Stiffness matrix update is allowed only once within the same time step.

3. if the bisection is in progress, additional K-updates are allowed within a given $\Delta T$ at every KSTEP-th converged bisection, where KSTEP is a user-specified parameter in the TSTEPNL entry.

4. Decomposition is performed within the module at every bisection or time step adjustment.

5. Whenever the decomposition is performed, the iteration starts from the previously converged position.

6. The stiffness matrix is updated at the beginning of each subcase.

## 5.5 Fourier Transform

The Fourier transform capability in Simcenter Nastran allows a transient analysis to be performed using a frequency response solution. Time-dependent applied loads are transformed to the frequency domain and all frequency dependent matrix calculations are completed. The frequency response solution variables are then transformed back into the time domain.

Fourier transform methods have been implemented in Simcenter Nastran to integrate the equations of motion in order to obtain the aeroelastic response of fixed wing aircraft. This capability is especially important for this type of analysis since the unsteady aerodynamic matrices are known only in the frequency domain. The Fourier transform method may also be used to solve for the transient response of conventional structural models (no aerodynamic effects) subjected to periodic loads.

This capability is available in SOLs 108 and 111 for frequency response output data. For transient type output, SOL 146 must be used. The transformation is performed when the requested load is the TLOADi form.

### Theory

Two forms of the Fourier transform are available: the Fourier series and the Fourier integral. Both methods require necessary numerical compromises and hence produce numerical approximations. The inverse transform includes an infinite sum, for which only a finite number of terms are numerically evaluated. The inverse Fourier integral must be numerically integrated, which may result in integration errors. The number of frequencies at which the integral is evaluated is limited by the cost of calculations.

In the Fourier series, the basic time interval is $0 < t < T$, with the function periodic. The circular frequencies are given by

$$\begin{cases} \omega_n = 2\pi n \Delta f \\ \Delta f = \dfrac{1}{T} \end{cases}$$

where $T$ is a large time equal to the period of the lowest forcing frequency.

The load transformation for a load at point *a* is given by

$$\tilde{P}_a(\omega_n) = \int_0^T P_a(t) e^{-i\omega_n t} dt$$

**Equation 5-21.**

The response at point *j* is given by

$$\tilde{u}_j(\omega_n) = H_{ja}(\omega_n)\tilde{P}_a(\omega_n)$$

**Equation 5-22.**

where $H_{ja}(\omega_n)$ is the frequency response of any physical variable due to unit load. The response in the time domain is given by

$$u_j(t) = \frac{\Delta\omega}{\pi}\left[\left(\frac{1}{2}\right)\tilde{u}_j(0) + \sum_{n=1}^{\infty} Re(\tilde{u}_j(\omega_n)e^{i\omega_n t})\right]$$

**Equation 5-23.**

In the Fourier Integral, the time interval is the limit as $T \to \infty$, $\Delta f \to 0$, and $2\pi n\Delta f \to \omega$ of the Fourier series. Here, $\omega$ is a continuous variable. **Eq. 5-21**, **Eq. 5-22**, and **Eq. 5-23** take the form

$$\tilde{P}_a(\omega) = \int_0^{\infty} P_a(t) e^{-i\omega t} dt$$

**Equation 5-24.**

$$\tilde{u}_j(\omega) = H_{ja}(\omega)\tilde{P}_a(\omega)$$

**Equation 5-25.**

$$u_j(t) = \left(\frac{1}{\pi}\right)\int_0^{\infty} Re(\tilde{u}_j(\omega)e^{i\omega t})d\omega$$

**Equation 5-26.**

## Transformation of Loads to the Frequency Domain

The transformation of the user-specified time dependent loads into the frequency domain is given in **Eq. 5-27**.

$$\tilde{P}_a(\omega_n) = \int_0^{T} P_a(t)e^{-i\omega_n t}dt$$

**Equation 5-27.**

For more information, see **Theory**.

With this transformation, the user must define a function that vanishes for $t > T$.

For piecewise linear tabular functions (TLOAD1), a table of pairs $(x_i, Y_i)(i = 1,N)$ prescribes N – 1 time intervals. If an X1 shift and an X2 scale factor are included, the time-dependent load at point $a$ is given by

$$P_a(t) = A_a Y_T\left(\frac{t - \tau_a - X1}{X2}\right)$$

**Equation 5-28.**

where $A_a$ is an amplitude factor and $\tau_a$ is a delay factor that may depend upon the loading point. Applying finite step-by-step integration to **Eq. 5-27**, the transformed load, $P_a(\omega)$, is obtained for each requested frequency.

Likewise, the general function (TLOAD2) is defined by

$$P_a(T) = \begin{cases} A_a\tilde{t}^n e^{\alpha\tilde{t}}\cos(2\pi f\tilde{t} + \phi) & 0 < t < T_2 - T_1 \\ 0 & \text{Otherwise} \end{cases}$$

**Equation 5-29.**

where $t = t - T_1 - \tau_a$

The value of $n$ must be an integer for transient analysis by the Fourier method. The transformation to the frequency domain is also obtained by numerical integration.

These loads, which appear in the form required for frequency response, are transformed to the modal coordinates exactly as in the modal frequency response method.

One other source of loads for aeroelastic problems is a one-dimensional gust. The same time dependencies are allowed as defined above; however, the amplitude ($A_a$) and delays ($\tau_a$) for the aerodynamic elements are computed from areas, dihedrals, and coordinates in the flow direction.

## Calculation of Frequency Response

Frequency responses are computed in the frequency domain by conventional Simcenter Nastran methods for coupled equations.

## Inverse Transformation of the Response to the Time Domain

The response in the time domain is found either from the Fourier integral approximation of **Eq. 5-30** or from the Fourier series result of **Eq. 5-31** (which can be thought of as a special form of the integral approximation).

$$u_j(t) = \left(\frac{1}{\pi}\right) \int_0^\infty Re(\tilde{u}_j(\omega) e^{i\omega t}) d\omega$$

**Equation 5-30.**

$$u_j(t) = \frac{\Delta\omega}{\pi}\left[\left(\frac{1}{2}\right)\tilde{u}_j(0) + \sum_{n=1}^{\infty} Re(\tilde{u}_j(\omega_n) e^{i\omega_n t})\right]$$

**Equation 5-31.**

For more information, see **Theory**.

Three approximation methods are available to evaluate the inverse transform which may be selected by the user via the parameter IFTM. In all cases, the quantity $\tilde{u}(\omega)$ is first calculated at a set of frequencies, $\omega_i$, by the frequency response module where the $\omega_i$'s do not need to be equally spaced. For all methods, $\tilde{u}(\omega)$ is set equal to zero outside the range of $\omega$'s computed. These methods are:

| | |
|---|---|
| **Method 0** | Approximate $\tilde{u}(\omega)e^{i\omega t}$ as a constant in each interval (the default method). For equal frequency intervals, this method reduces to the Fourier series approximation of Eq. 3. |
| **Method 1** | Fit $\tilde{u}(\omega)$ with a piecewise linear function, and do not approximate $e^{i\omega t}$. |
| **Method 2** | Fit $\tilde{u}(\omega)$ with a cubic spline function, and do not approximate $e^{i\omega t}$. |

Consider Method 2. Solving the three-moment equations, the second derivatives $\tilde{u}(\omega)$, can be found for each $\omega$ for which a frequency response has been computed. Then, in any interval $\omega_i < \omega < \omega_{i+1}$,

$$\tilde{u}(\omega) = [\tilde{u}(\omega_i) \cdot s + \tilde{u}(\omega_{i+1}) \cdot r] - \left[\frac{(\omega_{i+1} - \omega_i)^2}{6}\right] \cdot [\tilde{u}(\omega_i)s - s^3 + u(\omega_{i+1})(r - r^3)]$$

**Equation 5-32.**

where:

$$r = \frac{(\omega - \omega_i)}{(\omega_{i+1} - \omega_i)}$$

$$s = 1 - r$$

Integrate **Eq. 5-30** using **Eq. 5-32** for $\tilde{u}(\omega)$, and sum over the integrals. Then collect the terms for each $\omega_n$ with the result,

$$u(t) = \left(\frac{1}{\pi}\right) \sum_{n=1}^{N} \mathrm{Re}\left\{[C_n(t)\tilde{u}(\omega_n) + D_n(t)\tilde{u}(\omega_n)]e^{i\omega_n t}\right\}$$

**Equation 5-33.**

$$C_n(t) = \frac{\omega_n - \omega_{n-1}}{2} E_2(-it(\omega_n - \omega_{n-1})) + \frac{\omega_{n+1} - \omega_n}{2} E_2(it(\omega_{n+1} - \omega_n))$$

**Equation 5-34.**

$$D_n(t) = \frac{(\omega_n - \omega_{n-1})^3}{24} G(it(\omega_n - \omega_{n-1})) - \frac{(\omega_{n+1} - \omega_n)^3}{24} G(it(\omega_{n+1} - \omega_n))$$

**Equation 5-35.**

For the first terms in Eq. 5-33 ($n = 1$), use only the second terms on the right side of Eq. 5-34 and Eq. 5-35. For the last term in Eq. 5-33 ($n = N$), use only the first terms on the right side of Eq. 5-34 and Eq. 5-35.

$$G(z) = 2E_2(z) - E_4(z)$$

**Equation 5-36.**

$$E_K(z) = \frac{K!}{z^K}\left[e^z - \sum_{k=0}^{k=K} \frac{z^k}{k!}\right]$$
$$1 + \frac{z}{K+1} + \frac{z^2}{(K+1)(K+2)} + \frac{z^3}{(K+1)(K+2)(K+3)}$$

**Equation 5-37.**

The above form of the inverse transform has two advantages. First, numerical problems for small values of $t\Delta\omega$ are efficiently evaluated by choosing the series form of Eq. 5-37. Also, the other two methods are easily derived as subcases. If $u^n$ terms are removed from Eq. 5-32, a piecewise linear fit occurs. Thus, Method 1 results by deleting $\tilde{u}''$ from Eq. 5-33, i.e.,

$$u(t) = \left(\frac{1}{\pi}\right) \sum_{n=1}^{N} \mathrm{Re}[C_n(t) u(\omega_n) e^{i\omega_n t}]$$

**Equation 5-38.**

with $C_n$ defined by Eq. 5-35. Method 0 results if we replace $E_2$ by 1.0 in Eq. 5-35.

The above procedure for Method 0 always multiplies the first and last terms in the series by one-half. In order to force agreement with the Fourier series, which is the limiting case of equal frequency intervals, the first term in the series is multiplied by one-half only if the value of the first frequency is zero.

Some special considerations are given in the equal frequency interval case. When all $\Delta f$'s are equal and the first frequency is an integer multiple of $\Delta f$, the time step $\delta t$ is adjusted to make $\delta f \cdot \Delta t = 1$ / integer,

reducing the number of distinct values of $\sin\omega^t_n$ and $\cos\omega^t_n$ used in **Eq. 5-38**. Also, $C_n(t)$ and $D_n(t)$ (**Eq. 5-34** and **Eq. 5-35**) become independent of $n$ and do not need to be computed at every frequency.

Other important practical considerations must be observed to use these methods successfully. to illustrate one problem, consider the response of a simple damped oscillator to a pulse (**Figure 5-3**). The upper three curves show the pulse and the response of the system if it is very stable and slightly stable. Using the Fourier method, the pulse is replaced by a series of pulses, with period $1 / \Delta f$.

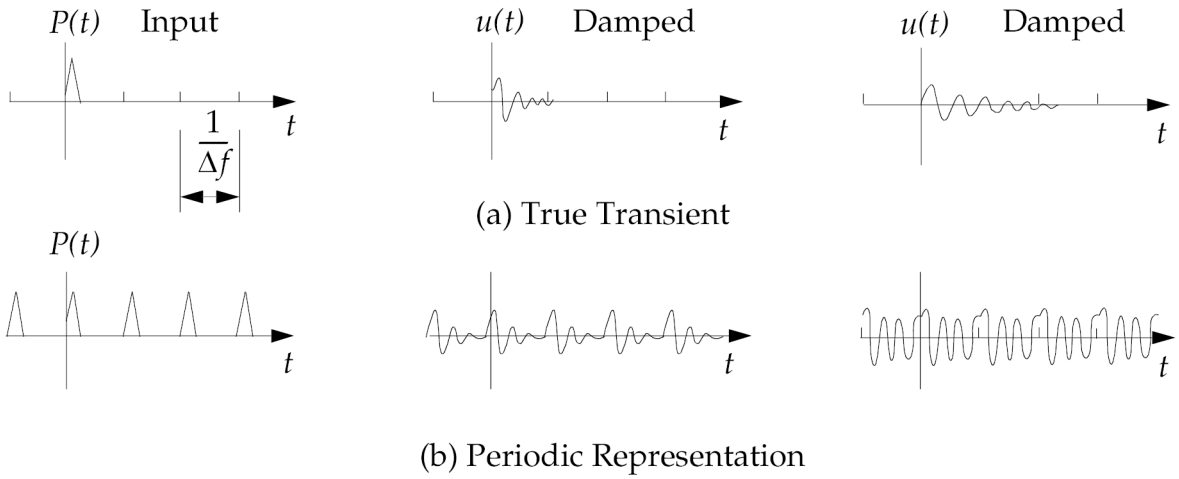



**Figure 5-3. Response of a Single Degree-of-Freedom for Three Types of Periodic Loads**

### Guidelines for Use

As can be seen, this method gives good results if the system is damped, but an incorrect impression if the system is lightly damped. Thus, in order for the results of the Fourier method to be valid:

1. The system should be reasonably well damped.

2. The forcing function should be zero for some time interval to allow decay.

3. The frequency interval $\Delta f \leq 1 (T_{pulse} + T_{decay})$.

If the system has unstable modes, these will appear as a precursor before the pulse as a stable mode in the reverse time. In general, Methods 1 and 2 are more accurate than Method 0. However, these methods introduce positive artificial damping into the result that may lead to erroneous conclusions in stability studies. To see this, consider the function $C_n(t)$ in **Eq. 5-39**. If equal frequency intervals are chosen, then

$$C_n(t) = \begin{cases} \Delta\omega \cdot \dfrac{1 - \cos(t\Delta\omega)}{1/2(t\Delta\omega)^2} = \Delta\omega\left(1 - \dfrac{1}{6}(t\Delta\omega)^2 + \ldots\right) & \text{Method 0} \\ \Delta\omega & \text{Method 1} \end{cases}$$

**Equation 5-39.**

Thus, Method 1 (and also Method 2) produces a decaying envelope that the user may incorrectly interpret as additional damping.

The use of equal frequency intervals versus unequal intervals has been studied and results are shown in the *Simcenter Nastran Aeroelastic Analysis User's Guide*.

## 5.6 Viscoelastic Material Properties

The mechanical behavior of many glasses, rubbers, and high polymers under stress is described in terms of a combination of elastic and viscous phenomena that may be approximated by linear viscoelastic theory. The dynamic behavior of such material is characterized by a strong dependence on frequency; and, under sinusoidal excitation, these materials exhibit a steady-state response in which the stress lags the associated strain. The general three-dimensional state of stress can then be described in terms of a complex frequency-dependent shear modulus and a real constant value for Poisson's ratio.

This method was developed for the analysis of rubber-like materials such as solid rocket propellants. In that case, the dominant strain energy terms were the shear strains.

Simcenter Nastran provides the ability to represent a single complex frequency-dependent scalar material modulus of the form

$$G(f) = G'(f) + iG''(f)$$

**Equation 5-40.**

where:

| | | |
|---|---|---|
| $G'$ | = | Shear storage modulus |
| $G''$ | = | Shear loss modulus |

The ratio

$$\frac{G''(f)}{G'(f)} = \tan\phi$$

is denoted as the shear loss tangent.

The above formulation of viscoelastic (frequency-dependent) material properties may be used in direct frequency analysis (SOL 108).

The stiffness and damping components of the dynamic matrices for direct frequency response analysis are documented in the *Simcenter Nastran Basic Dynamic Analysis User's Guide* in the following form:

$$[K_{dd}] = (1 + ig)[K^1_{dd}] + [K^2_{dd}] + [K^4_{dd}]$$

**Equation 5-41.**

$$[B_{dd}] = [B^1_{dd}] + [B^2_{dd}]$$

**Equation 5-42.**

where:

| | | |
|---|---|---|
| $g$ | = | overall structural damping specified through the PARAM,G Bulk Data entry |
| $[K^1_{dd}]$ | = | stiffness matrix for structural elements |
| $[K^2_{dd}]$ | = | stiffness terms generated through direct matrix input, e.g., DMIG Bulk Data entries |
| $[K^4_{dd}]$ | = | element damping matrix generated by the multiplication of individual element stiffness matrices by an element damping, $g_e$, entered on the MATi Bulk Data entry associated with the element or elements in question |
| $[B^1_{dd}]$ | = | damping matrix generated through CVISC and CDAMP1 Bulk Data entries |
| $[B^2_{dd}]$ | = | damping terms generated through direct matrix input, e.g., DMIG Bulk Data entries |

Eq. 5-41 is of particular interest in the current discussion of viscoelastic material properties because the presence of these properties will be reflected in terms of this equation. For discussion purposes, frequency-dependent material properties will be denoted as viscoelastic materials and those material properties that are independent of frequency will be denoted as elastic materials. Thus, if the stiffness properties for the viscoelastic elements are initially computed on the basis of a representative reference modulus, $G_{REF}$, the stiffness matrix for the viscoelastic elements (denoted by the subscript $V$) may be written in the form

$$[K_{dd}(f)]_V = \left[\frac{G'(f) + iG''(f)}{G_{REF}}\right][K^1_{dd}]_V$$

**Equation 5-43.**

## Input Description

To use the viscoelastic capability, the following conditions are necessary:

1. Assume the $[K^1_{dd}]$ matrix will be restricted only to the viscoelastic elements. This restriction implies that elastic elements will have a blank or zero entry in the GE field on their associated MATi Bulk Data entries. Conversely, all viscoelastic elements must have the reference element damping ($g_{REF}$) defined in the GE field, and the reference shear modulus ($G_{REF}$) defined in the G field on their associated MATi Bulk Data entries. Then, by definition,

$$[K^4_{dd}]_V = g_{REF}[K^1_{dd}]_V$$

2. The TABLEDi tabular functions *TR* (*f*) and *TI* (*f*) are defined to represent the complex moduli of all viscoelastic materials.

These two conditions may be combined in

$$[K_{dd}] = (1 + ig)[K^1_{dd}] + [K^2_{dd}] + [K^4_{dd}]$$

(for more information, see **Viscoelastic Material Properties** ) to provide the following expression:

$$\begin{aligned}[K_{dd}]_V &= (1 + ig)[K^1_{dd}]_V + \{TR(f) + iTI(f)\}[K^4_{dd}]_V \\ &= \{(1 + g_{REF}TR(f)) + i[g + g_{REF}TI(f)]\}[K^1_{dd}]_V\end{aligned}$$

**Equation 5-44.**

A comparison of

$$[K_{dd}(f)]_V = \left[\frac{G'(f) + iG''(f)}{G_{REF}}\right][K^1_{dd}]_V$$

(for more information, see **Viscoelastic Material Properties** ) and **Eq. 5-44** yields the form of the tabular functions *TR* (*f*) and *TI*(*f*):

$$TR(f) = \frac{1}{g_{REF}}\left[\frac{G'(f)}{G_{REF}} - 1\right]$$

**Equation 5-45.**

$$TI(f) = \frac{1}{g_{REF}}\left[\frac{G''(f)}{G_{REF}} - g\right]$$

**Equation 5-46.**

Note that the direct input matrix, $[K^2_{dd}]$, from

$$[K_{dd}] = (1 + ig)[K^1_{dd}] + [K^2_{dd}] + [K^4_{dd}]$$

(for more information, see **Viscoelastic Material Properties** ) is still available but not involved in the definition of viscoelasticity.

Direct frequency response analyses that involve viscoelastic materials require some special input data relative to analyses that involve only elastic materials. These special input requirements are given below:

1. Case Control Section:
   SDAMPING = *n* reference TABLEDi Bulk Data entry that defines the alternate tabular form of *TR*(*f*)

2. Bulk Data Section:

   a. MATi Bulk Data entry

      - $G = G_{REF}$, the reference modulus
      - *NU* = Poisson's ratio for the viscoelastic material
      - $GE = g_{REF}$, the reference element damping
      - All other entries on the MATi Bulk Data entry are utilized in the standard manner.

   b. TABLEDi Bulk Data entries:

      - A TABLEDi Bulk Data entry with an *ID* = *n* is used to define the function *TR*(*f*) of **Eq. 5-45**.
      - A TABLEDi Bulk Data entry with an *ID* = *n* + 1 is used to define the function *TI*(*f*) of **Eq. 5-46**.

All other input requirements to the Simcenter Nastran Bulk Data are typical of direct frequency response analysis. Note that the overall structural damping, *g*, entered through the PARAM Bulk Data entry (PARAM,G,XX) applies to all elastic materials.

**Compare With Theory**

The functional form of **Eq. 5-45** and **Eq. 5-46** requires the user to perform some modest calculations that involve $G_{REF}$ and $g_{REF}$. You must also input these two terms through the MATi Bulk Data entry for the

viscoelastic materials. In general, you should use representative values for these parameters. However, in those cases where you do not use the Simcenter Nastran OMIT feature, the calculation of *TR*(*f*) and *TI*(*f*) can be simplified. In this case, define

$$G_{REF} \ll G'$$

$$G_{REF} \ll \frac{G''}{g}$$

and select $g_{REF}$ so that

$$G_{REF} g_{REF} = 1$$

Then **Eq. 5-45** and **Eq. 5-46** reduce to the following form:

$$TR(f) = G'(f)$$

**Equation 5-47.**

$$TI(f) = G''(f)$$

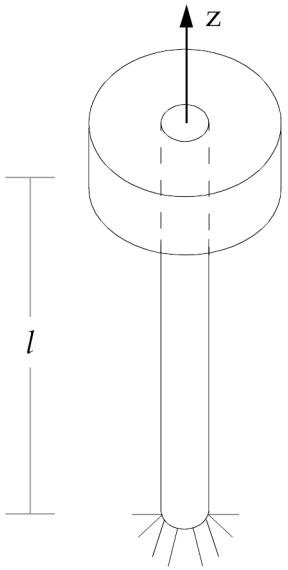
**Equation 5-48.**

The above simplifications should not be used if OMIT calculations are involved in the analysis in order to avoid possible matrix ill-conditioning.

Note that stress and force data recovery calculations are performed with the reference moduli irrespective of frequency.

## Example

To illustrate the representation of viscoelastic material properties in Simcenter Nastran, consider the following structure that may undergo both axial extension along the z-axis and torsion about the z-axis:




where:

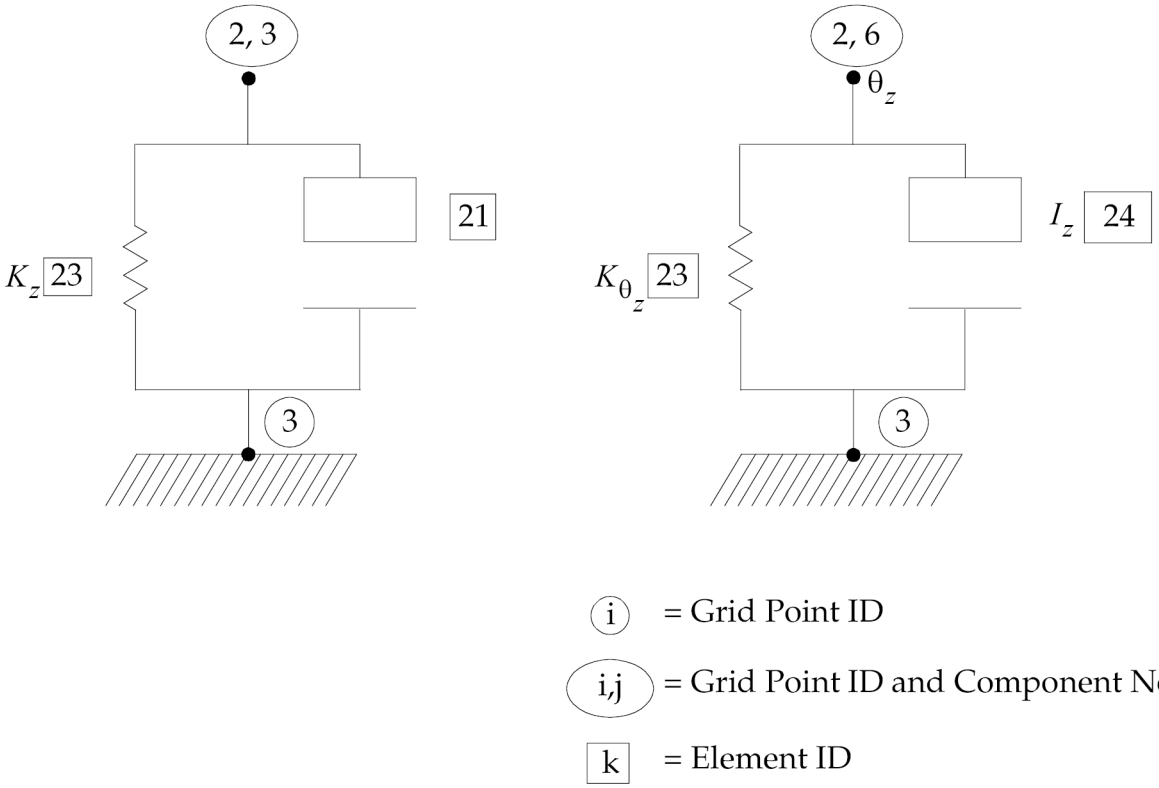
| | | |
|---|---|---|
| flywheel mass, $M_z$ | = | 2.0 |
| flywheel inertia, $I_z$ | = | 10.0 |
| axial stiffness, $K_z$ | = | $\frac{E(f)A}{l}$ |
| torsional stiffness, $K\theta_z$ | = | $\frac{G(f)J}{l}$ |
| length, $l$ | = | 2.0 |
| area, $A$ | = | .9 |
| area moment, $J$ | = | 2.0 |

The symbols $E(f)$ and $G(f)$ imply that the extensional and torsional moduli are functions of frequency, i.e., viscoelastic. For simplicity, it is assumed that $E(f) = G(f)$ and that these quantities have the following frequency-dependent characteristics:

| $f$, hz | $G'(f)$ | $G''(f)$ |
|---|---|---|
| .8 | 1800. | 180. |
| 1.1 | 1850. | 185. |
| 1.4 | 1910. | 191. |
| 1.7 | 1970. | 197. |
| 2.0 | 2030. | 203. |
| 2.3 | 2070. | 207. |

| f, hz | G ´(f) | G″(f) |
|---|---|---|
| 2.6 | 2140. | 214. |
| 2.9 | 2210. | 221. |

A model for the system is shown in the following schematic:




You can generate this model with the following Bulk Data entries:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| GRID | ID | CP | X1 | X2 | X3 | CD | PS | | |
| GRID | 2 | | | | 2. | | 1245 | | |
| GRID | 3 | | | | | | 123456 | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| CMASS2 | EID | M | G1 | C1 | G2 | C2 | | | |
| CMASS2 | 21 | 2. | 2 | 3 | 3 | 3 | | | |
| CMASS2 | 24 | 10. | 2 | 6 | 3 | 6 | | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| CROD | EID | PID | G1 | G2 | | | | | |
| CROD | 23 | 1 | 2 | 3 | | | | | |

| PROD | PID | MID | A | J | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| PROD | 1 | 1 | .9 | 2. | | | | | |

| MAT1 | MID | E | G | NU | RHO | A | TREF | GE | |
|---|---|---|---|---|---|---|---|---|---|
| MAT1 | 1 | 2000. | 2000. | | | | | .09 | |

Note that the reference values of 2000. for both *E* and *G* are specified on the MAT1 Bulk Data entry. The reference value for structural damping, $g_{REF}$, is set to .09 under the GE field of the MAT1 Bulk Data entry. Once the reference values $G_{REF}$ and $g_{REF}$ have been assigned, one can evaluate **Eq. 5-45** and **Eq. 5-46** for the values to be assigned to *TR*(*f*) and (*TI*(*f*). The values for these functions are entered on TABLEDi Bulk Data entries. As elastic portions of the structure may exist in addition to viscoelastic portions, assume that a value of overall structural damping, g, is to be utilized for these elastic portions of the model. The overall structural damping value of .06 is assigned through the following PARAM Bulk Data entry.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| PARAM | N | V1 | V2 | | | | | | |
| PARAM | G | .06 | | | | | | | |

This value of *g* must be considered in **Eq. 5-46**. The evaluation of **Eq. 5-45** and **Eq. 5-46** will result in the values shown in the following TABLED1 Bulk Data entries:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| TABLED1 | ID | | | | | | | | |
| TABLED1 | 10 | | | | | | | | +ABR1 |

| | $X_1$ | $Y_1$ | $X_2$ | $Y_2$ | $X_3$ | $Y_3$ | $X_4$ | $Y_4$ | |
|---|---|---|---|---|---|---|---|---|---|
| +ABR1 | .0 | .0 | .8 | −1.11111 | 1.1 | −.833333 | 1.4 | −.5 | +ABR2 |

| | $X_5$ | $Y_5$ | $X_6$ | $Y_6$ | $X_7$ | $Y_7$ | $X_8$ | $Y_8$ | |
|---|---|---|---|---|---|---|---|---|---|
| +ABR2 | 1.7 | −.166667 | 2. | .1666667 | 2.3 | .3888889 | 2.6 | .7777777 | +ABR3 |

| | $X_9$ | $Y_9$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| +ABR3 | 2.9 | .5611111 | ENDT | | | | | | |

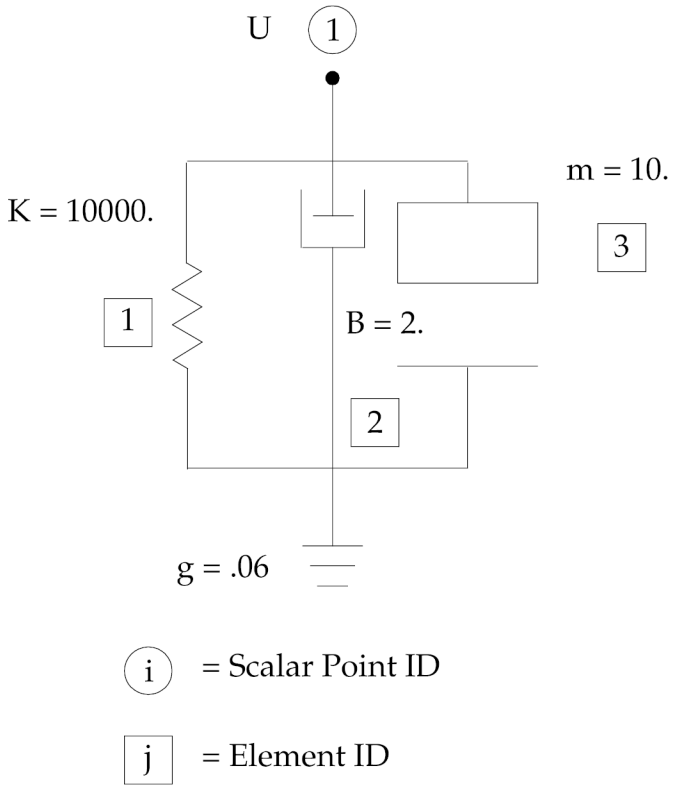
| TABLED1 | ID | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| TABLED1 | 11 | | | | | | | | +ABI1 |

| | $X_1$ | $Y_1$ | $X_2$ | $Y_2$ | $X_3$ | $Y_3$ | $X_4$ | $Y_4$ | |
|---|---|---|---|---|---|---|---|---|---|
| +ABI1 | 0. | 0. | .8 | .3333333 | 1.1 | .3611111 | 1.4 | .3944444 | +ABI2 |

| | $X_5$ | $Y_5$ | $X_6$ | $Y_6$ | $X_7$ | $Y_7$ | $X_8$ | $Y_8$ | |
|---|---|---|---|---|---|---|---|---|---|
| +ABI2 | 1.7 | .4277778 | 2. | .4611111 | 2.3 | .4833333 | 2.6 | .5222222 | +ABI3 |

| | $X_9$ | $Y_9$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| +ABI3 | 2.9 | .5611111 | ENDT | | | | | | |

To demonstrate that elastic as well as viscoelastic elements may be included in the same analysis, the following single degree-of-freedom is added to the Bulk Data Section:




The following Bulk Data entries are required to represent the foregoing single degree-of-freedom oscillator:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| CELAS2 | EID | K | G1 | C1 | G2 | C2 | GE | S | |
| CELAS2 | 1 | 1000. | 1 | | | | | | |

| CDAMP2 | EID | B | G1 | C1 | G2 | C2 | | | |
|---|---|---|---|---|---|---|---|---|---|
| CDAMP2 | 2 | 2. | 1 | | | | | | |

| CMASS2 | EID | M | G1 | C1 | G2 | C2 | | | |
|---|---|---|---|---|---|---|---|---|---|
| CMASS2 | 3 | 10. | 1 | | | | | | |

The excitation for both disjoint models is a force (moment for the torsional system) with a magnitude of cos$\omega t$. This function can be generated with the following Bulk Data entries:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| RLOAD2 | SID | L | M | N | TB | TP | | | |
| RLOAD2 | 1 | 1 | 0 | 0 | 1 | 0 | | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| TABLED1 | ID | | | | | | | | |
| TABLED1 | 1 | | | | | | | | +ABC |

| | $X_1$ | $Y_1$ | $X_2$ | $Y_2$ | $X_3$ | $Y_3$ | | | |
|---|---|---|---|---|---|---|---|---|---|
| +ABC | 0. | 1. | 1. | 1. | 100. | 1. | ENDT | | |

| DAREA | SID | P | C | A | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| DAREA | 1 | 2 | 3 | 1. | | | | | |

| DAREA | 1 | 2 | 6 | 1. | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| DAREA | 1 | 1 | | 1. | | | | | |

To perform a frequency response analysis is necessary to provide a list of frequencies (Hz) at which solutions are desired. The following FREQ1 Bulk Data entry is used for this purpose.

| FREQ1 | SID | F1 | DF | NDF | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| FREQ1 | 1 | .5 | .3 | 8 | | | | | |

The complete Simcenter Nastran input file is listed below:

```
ID TEST,DAMPING
SOL 108
CEND
TITLE=FREQUENCY RESPONSE
SUBTITLE=VISCOELASTIC AND ELASTIC MATERIALS
DLOAD=1
SDAMP=10
FREQ=1
SPC=1
SET 1=1,2
DISP=1
VELO=1
ACCE=1
SET 7=23
STRESS=7
BEGIN BULK
$ INCLUDE ALL BULK DATA ENTRIES FOR BOTH
$ DISJOINT MODELS DISCUSSED IN THE PRECEDING
$ REMARKS
$
```

```
$********
$
$RID     ID     CP     X1     X2     X3     CD     PS
GRID     2                2.         1245
GRID     3                          123456
$
$********
$
$MASS2     EID     M     G1     C1     G2     C2
CMASS2     21     2.     2     3     3     3
CMASS2     24     10.     2     6     3     6
$ROD     EID     PID     G1     G2
CROD     23     1     2     3
$ROD     PID     MID     A     J
PROD     1     1     .9     2.
$AT1     MID     E     G     NU     RHO     A     TREF     GE
MAT1     1     2000.     2000.                     .09
$
$********
$
$ARAM     N     V1     V2
PARAM     G     .06
$
$********
$
$ABLED1 ID
TABLED1 10                                   +ABR1
$    X1     Y1     X2     Y2     X3     Y3     X4     Y4
+ABR1     .0     .0     .8     -1.1111 1.1     -.83333 1.4     -.5
+ABR2
$    X5     Y5     X6     Y6     X7     Y7     X8     Y8
+ABR2     1.7     -.16667 2.     .166667 2.3     .388889 2.6     .777777
+ABR3
$    X9     Y9
+ABR3     2.9     .561111 ENDT
$
$********
$
$ABLED1 ID
TABLED1 11                                   +ABI1
$    X1     Y1     X2     Y2     X3     Y3     X4     Y4
+ABI1     0.     0.     .8     .333333 1.1     .361111 1.4     .394444
+ABI2
$    X5     Y5     X6     Y6     X7     Y7     X8     Y8
+ABI2     1.7     .427778 2.     .461111 2.3     .483333 2.6     .522222
+ABI3
$    X9     Y9     X10     Y10     X11     Y11     X12     Y12
+ABI3     2.9     .561111     ENDT
$
```

```
$********
$
$ELAS2     EID      K      G1      C1      G2      C2      GE      S
CELAS2     1      1000.      1
$DAMP2     EID      B      G1      C1     G2      C2
CDAMP2     2      2.      1
$MASS2     EID      M      G1      C1      G2      C2
CMASS2     3      10.      1
$
$********
$
$LOAD2  SID      L      M      N      TB      TP
RLOAD2     1      1      0      0      1      0
$ABLED1 ID
TABLED1 1                                    +ABC
$    X1      Y1      X2      Y2      X3      Y3
+ABC      0.      1.      1.      1.      100.      1.      ENDT
$
$AREA      SID      P      C      A
DAREA      1      2      3      1.
$
$REQ1      SID      F1      DF      NDF
FREQ1      1      .5      .3      8
$
$********
$
ENDDATA
```

## 5.7 Free Body Techniques

Free body motion in a structure occurs when a structure may move freely without applied forces. Although the stiffness matrix will have one or more singularities, the combined dynamic system, with mass and damping, may not be singular. Examples are flying objects such as aircraft or rockets, and structures with kinematic mechanisms such as a gyroscope or a pendulum.

In most cases, the dynamic response solutions in Simcenter Nastran do not require any special attention for free body motions. In general, Simcenter Nastran decomposes matrix combinations that are not singular. For example, an unbalanced load applied to a free body has a unique dynamic solution, namely a constant acceleration. The only exception is that of a frequency response analysis at a frequency of zero. In this case it is recommended that a small frequency be used instead of zero for free bodies.

In some cases, it may be necessary to use the SUPORT Bulk Data to provide a method of defining the free body modes of a structure. The primary use for this process is in inertia relief option for static analysis.

See Also

- "Inertia Relief in Linear Static Analysis" in the *Simcenter Nastran User's Guide*

In other cases, the basic geometry is used for the definition of the rigid body displacements. The parameter GRDPNT specifies a grid point or a location for the six rigid body displacement vectors—three translations and three rotations. This method is used in the weight and CG calculations and is used for inertia relief with superelements.

## The SUPORT Option

The SUPORT data defines a set of degrees-of-freedom that are used to restrain free motion of the structure *temporarily* for the purposes of calculating normal modes, providing a moving frame of reference, or measuring unbalanced loads. The basic theory and usage are given in SUPORT Entry in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*. Other bulk data variations that perform this function are CYSUP Bulk Data (cyclic symmetry) and SESUP Bulk Data (superelements).

In dynamic analysis, some specialized areas in which it is useful to apply free body SUPORTs include:

1. Calculation of pure zero-frequency eigenvalues and, if necessary, their eigenvectors. The result depends on the chosen eigenvalue extraction method. In the Inverse Power method, the zero frequency eigenvectors are obtained from the stiffness matrix. In the other real eigenvalue methods, the number of SUPORT degrees-of-freedom indicate the number of zero-frequency modes to be calculated.

2. Use of the large mass technique to enforce motion in a modal formulation. Here the SUPORTs are essential to obtain frequencies that are exactly zero. (See **"Enforced Motion with Loads"** .)

3. Response Spectra analysis with participation factors. Here the responses are calculated relative to a moving base, defined by SUPORTs (see **"Shock and Response Spectrum Analysis"** ).

4. Verification of new FE models with a free body test will find several types of errors. It is useful to remove the constraints for a fixed structure to perform extra check runs. For instance, the calculations for SUPORT inputs will indicate if additional mechanism modes exist.

5. SUPORT data is required for mode acceleration data recovery of free bodies. In this calculation it is necessary to solve a quasi static equation using the stiffness matrix. The use of SUPORTs will eliminate singularities without over-constraining the structure.

6. SESUP data are used for modal analysis of individual superelements. This is used for Component Mode Synthesis when SECSET boundaries are used to obtain free body modes.

Each of these topics is described elsewhere in this guide. However, they all have the same limitations and requirements on the selection of the SUPORT set.

The SUPORT inputs define a set of degrees-of-freedom, $u_r$, which belong to the partitions

$$\{u_a\} = \begin{Bmatrix} q \\ u_l \\ u_r \end{Bmatrix}$$

**Equation 5-49.**

where $q$ are the dynamic QSET or component modal coordinates, and $u_l$ are the physical degrees-of-freedom for the restrained structure. The free body motions of the structure may be obtained by solving a problem with zero forces. The partitioned static matrix solution is

$$\begin{bmatrix} K_{ll} & K_{lr} \\ K_{rl} & K_{rr} \end{bmatrix} \begin{bmatrix} D_l \\ I \end{bmatrix} = \{P\} = \{0\}$$

**Equation 5-50.**

where $[D_l]$ is called the rigid body transformation matrix. It is obtained by solving the equation:

$$[K_{ll}][K_l] = -[K_{lr}]$$

**Equation 5-51.**

Depending on the type of solution, the $[D_l]$ matrix may be used for extracting zero-frequency mode shapes, for finding relative deformation vectors and for measuring resultants for load and force vectors. The number of columns of $[D_l]$ equals the number of user-selected SUPORT points. This number may be less than six if other constraints are applied, or may be greater than six if mechanisms are present. (Usually six degrees-of-freedom are chosen at a hard point near the center of gravity.)

The requirements for the $u_r$ set are that they provide a nonredundant set of supports, namely that:

1. If the structure were to be constrained to zero displacement at the SUPORT DOF, no singularities will occur in the resulting stiffness matrix partition; i.e.,

$$[K_{ll}]^{-1} \text{ exists}$$

**Equation 5-52.**

2. If the unloaded structure were to be given enforced displacements at each of the SUPORT DOF (with the others constrained), no stresses or forces should occur in the structure; i.e.,

$$[K_{rl}][D_{lr}] + [K_{rr}] = [0]$$

**Equation 5-53.**

3. The SUPORT DOF are connected to structural elements which provide real supporting stiffness. (Points near the CG are preferred.)

4. In superelement analysis the final SUPORT set is restricted to the residual structure. (However, SESUP data may be used to define separate $u_r$ sets for modal analysis of upstream superelements.)

If any of these criteria are violated, the table output with User Information Message 3035 will contain large values for the epsilon and strain energy. The equations defining these parameters are given in Theoretical Considerations for Using SUPORT in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*.

## Geometric Shape Methods

In these cases, the rigid body shapes are defined geometrically by the locations of the grid points and orientation of their displacement coordinate systems. The structure may or may not be a free body. Also, mechanisms that produce free body differences in displacements or rotations are not included.

The parameter GRDPNT is used normally to calculate the total weight, inertia, and CG properties of a structure. It is also used to generate geometric rigid body shapes. The matrix $[D_l]$ as defined in **Eq. 5-54** may be calculated directly from the geometry. The VECPLOT Module generates a 6 × 6 matrix, $[D_l]$, for every grid point, *g*, from basic kinematics.

$$[K_{rl}][D_{lr}] + [K_{rr}] = [0]$$

**Equation 5-54.**

For more information, see **The SUPORT Option**.

The definition is

$$\{u_{gi}\} = [D_i]\{u_o\}$$

**Equation 5-55.**

where $\{u_{gi}\}$ is the vector of three displacements and three rotations in the local coordinate system. $\{u_o\}$ represents three displacements and three rotations at GRDPNT (or the origin) in basic coordinates. These 6 × 6 partitions are merged into a full-size matrix, $[D_{go}]$ with six rows for each grid point and six columns total. Note that the rows of $[D_{go}]$ are null for the scalar points and is a unit matrix for the six rows corresponding to the GRDPNT parameter.

If SUPORT data is present, the matrix is transformed to define motion relative to the supported degrees-of-freedom, $\{u_r\}$. The internal calculations are given below

Partition:

$$[D_{go}] \Rightarrow \begin{bmatrix} D_{\bar{g}o} \\ D_{ro} \end{bmatrix}$$

**Equation 5-56.**

Invert the support partition, and multiply:

$$[D_{\bar{g}r}] = [D_{\bar{g}o}][D_{ro}]^{-1}$$

**Equation 5-57.**

and merge:

$$[D_{gr}] \Leftarrow \begin{bmatrix} D_{\bar{g}r} \\ I \end{bmatrix}$$

**Equation 5-58.**

These matrices may be calculated at the superelement level as opposed to assembled residual SE level required by the SUPORT partition method.

The applications include:

1. Static inertia relief solutions for SOL 101. The requirement to combine and reduce mass matrices is eliminated with this method, saving nearly 50% of the execution cost. For inertia relief effects, these solutions also require the input PARAM,INREL,–1 and a set of six SUPORT degrees-of-freedom.

2. Component Mode Synthesis for superelements provides an option for inertia relief vectors. The advantage is improved accuracy in the final results with an equal or lesser number of component modes. These are deformation shapes that correspond to the six uniform acceleration loads on the constrained structure. The deformation shape vectors are appended to the mode shapes and are included as generalized degrees-of-freedom. The INRLM parameter controls this option.

Note that in addition to the limitation of six free body motions, the geometric method will not be correct for scalar points and fluid analysis methods.

## 5.8 Aeroelastic Solutions

The *Simcenter Nastran Aeroelastic Analysis User's Guide* describes the theoretical aspects and the numerical techniques used to perform aeroelastic analyses. As described in **"Overview of Aeroelastic Analysis"** , the system is used for flutter, frequency response, gust response, and static analysis of aerodynamically loaded structures. An outline of the capability is given here.

The aeroelastic analyses use the following features:

**Structural Model**

Any of the existing Simcenter Nastran structural finite elements (except axisymmetric elements) can be used to build the structural model. The structural stiffness, mass, and damping matrices required by the aeroelastic analyses are generated by Simcenter Nastran from geometric, structural, inertial, and damping data that you input for subsequent use in the various aeroelastic analyses.

**Fluid/Structure Connections**

Matrices of aerodynamic influence coefficients are computed only from the data describing the geometry of the aerodynamic finite elements. The choice of aerodynamic grid points for the aerodynamic model is independent of the location of the structural grid points. An automated interpolation procedure is provided to relate the aerodynamic to the structural degrees-of-freedom. Splining techniques for both lines and surfaces are used to generate the transformation matrix from structural grid point deflections to aerodynamic grid point deflections where local streamwise slopes are also computed. The transpose of this matrix transfers the aerodynamic forces and moments at aerodynamic boxes to structural grid points.

**Aerodynamic Theories**

One subsonic and two supersonic lifting surface aerodynamic theories are available in Simcenter Nastran, as well as Strip Theory. The subsonic theory is the Doublet-Lattice method, which can account for interference among multiple lifting surfaces and bodies. The supersonic theories are the Mach Box method and Piston Theory.

**Static Aeroelastic Analysis**

The structural load distribution on an elastic vehicle in trimmed flight is determined by solving the equations for static equilibrium. The SOL 144 and SOL 200 processes will calculate aerodynamic stability derivatives (e.g., lift and moment curve slopes and lift and moment coefficients due to control surface rotation) and trim variables (e.g., angle of attack and control surface setting) as well as aerodynamic and structural loads, structural deflections, and element stresses.

**Modal Formulation**

Dynamic aeroelastic solutions provide for modal reduction of the system matrices. The number of degrees-of-freedom required for accurate solutions to dynamic aeroelastic problems is generally far less than the number of physical degrees-of-freedom used in the finite element structural model. The

number of independent degrees-of-freedom can be greatly reduced by using the (complex) amplitudes of a series of vibration modes as generalized coordinates, e.g., by Galerkin's method. Simcenter Nastran can compute the vibration modes and frequencies and make the transformation to modal coordinates. The matrices of aerodynamic influence coefficients are also transformed to generalized aerodynamic forces by use of the vibration eigenvectors.

**Flutter Analysis**

The dynamic aeroelastic stability problem, flutter, is solved in SOL 145, by any of four methods. The traditional American flutter method developed by the Air Materiel Command (AMC) in 1942 is available in the first two methods.

- The K-method is a variation of the AMC method.
- The KE-method is a more efficient adaptation of the K-method from the point of view of tracking roots, but is limited in input (no viscous damping) and output (no eigenvectors).

The other two methods are based on the British flutter method, which was developed by the Royal Aircraft Establishment.

- The PK-method is similar to the British flutter method.
- The PKNL-method is the PK-method without looping over Mach number, density ratio, and velocity.

**Frequency Response**

The coupling with aerodynamic loads has also been added to the existing Simcenter Nastran structural modal frequency response capability, SOL 146. Analyses of frequency response to arbitrarily specified forcing functions can be carried out using the oscillatory aerodynamic loads from any of the available aerodynamic theories. Frequency response to a harmonic gust field can be calculated at subsonic speeds using the Doublet-Lattice method for wing/body interference.

**Transient Response**

Because unsteady aerodynamic loads are obtained only for steady-state harmonic motion, they are known only in the frequency- and not the time-domain. In SOL 146, Inverse Fourier Transform techniques provide the appropriate methods by which transient response is obtained from the frequency response. Both forward and inverse Fourier transforms are provided so that the time-varying forcing function or the gust profile can be transformed into the frequency domain. Then, after convolution with the system frequency response, the inverse transform leads to the transient response of the system to the specified forcing function or gust profile.

**Random Response**

Stationary random response of the system, is available in SOL 146 from specified loadings and the power spectral densities of loads. Loads may be either specified force distributions or harmonic gust fields. The statistical quantities of interest in the response are $\bar{A}$, the ratio of standard deviations (rms

values) of the response to that of the input loading, and $N_o$, the mean frequency of zero crossings (with a positive slope) of the response. The capability to compute these quantities was added to Simcenter Nastran by modifying the existing random response module to include options to generate various atmospheric turbulence power spectra and to perform the calculation of $N_o$.

**Design Sensitivities**

The sensitivities of response parameters to changes in design variables are calculated by the perturbation techniques developed for structural optimization in Simcenter Nastran and extended to include static aeroelasticity and flutter in SOL 200. The basic aeroelastic sensitivities that can be obtained include stability derivatives, trim variables, and flutter system dampings. The synthetic response technique of Simcenter Nastran optimization also permits the calculation of sensitivities of user-specified functions of those standard response quantities.

**Aeroelastic Optimization**

Optimization of aeroelastic characteristics can be combined with the other optimization features of Simcenter Nastran in SOL 200, and vehicles can now be designed optimally for aeroelastic loads, flying qualities, and flutter, as well as for strength, vibration frequencies, and buckling characteristics.

**Post-processing**

Aeroelastic results and model geometry are written to the OP2 file where they can be accessed by post-processing software.

- For SOLs 144 and 145, aeroelastic beam geometry defined as AEROBEAM elements and aeroelastic panel geometry defined as Q4AERO and T3AERO elements are written to the GEOM2 data block. AEROBEAM, Q4AERO, and T3AERO elements are elements that the software generates during the aeroelastic analysis solve.
- For SOL 145, normal modes and the corresponding mode shapes defined relative to the aeroelastic mesh are written to an OUGV1 data block.
- For SOL 145, complex modes and the corresponding complex mode shapes defined relative to the aeroelastic mesh are written to an OUGV1 data block.
- For SOL 145, data for XY plotting is written to the OVG data block.
- For SOLs 144 and 145, aeroelastic forces and moments are written to the OAEROF data block.
- For SOLs 144 and 145, aeroelastic pressures are written to the OAEROP data block.
- For SOL 144, aeroelastic control surface position and hinge moment are written to the OAERCSHM data block.
- For SOL 144, aeroelastic hinge moment derivative coefficients are written to the OAEROHMD data block.

- For SOL 144, aeroelastic stability and control derivative coefficients are written to the OAEROSCD data block.
- For SOL 144, aeroelastic trim variables are written to the OAEROTV data block.

## Introduction to Aeroelastic Analysis and Design

Aeroelastic analysis and design solution sequences extend the range of capabilities in Simcenter Nastran beyond basic static and dynamic structural analysis.

**Aeroelastic Modules**

Options are available to:

- Generate aerodynamic grid points.
- Compute aerodynamic matrices.
- Provide connection (interpolation) between the structural and aerodynamic grid points.
- Solve the equations for static aeroelasticity.
- Solve the equations for flutter.
- Solve the equations for dynamic aeroelastic response.
- Calculate aeroelastic design sensitivities.
- Optimize aeroelastic and related structural characteristics.

**Aeroelastic DMAP Sequences**

Four solution sequences are available:

1. SOL 144 for static aeroelastic analyses.
2. SOL 145 for modal flutter analyses by the K-, KE-, PK-, or PKNL-method.
3. SOL 146 for modal dynamic aeroelastic response analyses due to gusts or control surface deflections.
4. SOL 200 for design sensitivity and optimization including aeroelastic effects. Since this fourth sequence has applications to many areas other than aeroelasticity, the reader is referred to the *Simcenter Nastran Design Sensitivity and Optimization User's Guide* for a more comprehensive treatment. Aeroelastic optimization is beyond the scope of this Guide and is not discussed further in this section.

## Aerodynamic Theories

Simcenter Nastran has implemented five aerodynamic theories:

1. Doublet-Lattice subsonic lifting surface theory (DLM)
2. Subsonic wing-body interference theory (DLM with slender bodies)
3. Mach Box method
4. Strip Theory
5. Piston Theory

Each of these methods is described in the *Simcenter Nastran Aeroelastic Analysis User's Guide*. They all share a common matrix structure.

Three matrix equations summarize the relationships required to define a set of aerodynamic influence coefficients [see Rodden and Revell (1962)]. These are the basic relationships between the lifting pressure and the dimensionless vertical or normal velocity induced by the inclination of the surface to the airstream; i.e., the downwash (or normalwash),

$$\{w_j\} = [A_{jj}]\{f_j / \bar{q}\}$$

**Equation 5-59.**

the substantial differentiation matrix of the deflections to obtain downwash,

$$\{w_j\} = [D^1_{jk} + ik\ D^2_{jk}]\{u_k\} + \{w^g_j\}$$

**Equation 5-60.**

and the integration of the pressure to obtain forces and moments,

$$\{P_k\} = [S_{kj}]\{f_j\}$$

**Equation 5-61.**

where:

$w_j$ = downwash (dimensionless)

$w^g_j$ = static aerodynamic downwash; it includes, primarily, the static incidence distribution that may arise from an initial angle of attack, camber, or twist

$f_j$ = pressure on lifting element $j$

$\bar{q}$ = flight dynamic pressure

$k$ = reduced frequency, $k = \omega b/V$ where $\omega$ is the angular frequency, $b$ is a reference semichord, and $V$ is the free-stream velocity

$A_{jj}(m, k)$ = aerodynamic influence coefficient matrix, a function of Mach number ($m$), and reduced frequency ($k$)

$u_k$, $P_k$ = displacements and forces at aerodynamic grid points

$D^1_{jk}$, $D^2_{jk}$ = real and imaginary parts of substantial differentiation matrix, respectively (dimensionless)

$S_{kj}$ = integration matrix

### The Aerodynamic Influence Coefficient Matrix

The three matrices of **Eq. 5-60** and **Eq. 5-61** can be combined to give an aerodynamic influence coefficient matrix:

$$[Q_{kk}] = [S_{kj}][A_{jj}]^{-1}[D^1_{jk} + ik\ D^2_{jk}]$$

**Equation 5-62.**

All aerodynamic methods compute the $S$, $D^1$ and $D^2$ matrices at user-supplied Mach numbers and reduced frequencies. The Doublet-Lattice theory computes the $A$ matrix. Then, matrix decomposition and forward and backward substitution are used in the computation of the $Q$ matrix. The remaining methods compute $A^{-1}$ directly and use matrix multiplications to form $Q$. Details of the various methods are described in the *Simcenter Nastran Aeroelastic Analysis User's Guide*.

## Generation of Aerodynamic Matrices

The aerodynamic equations described above form the basis of the aerodynamic computations required for static aeroelastic analysis with some special purpose modifications made for the Simcenter Nastran implementation.

For static aeroelasticity, the downwash relation of **Eq. 5-63**

$$\{w_j\} = [D^1_{jk} + ik\ D^2_{jk}]\{u_k\} + \{w^g_j\}$$

**Equation 5-63.**

becomes

$$\{w_j\} = [D_{jk}]\{u_k\} + [D_{jx}]\{u_x\} + \{w_j^g\}$$

**Equation 5-64.**

where:

- $\{w_j\}$ = a vector of aerodynamic degrees-of-freedom (e.g., angles of attack)
- $\{u_k\}$ = a vector of aerodynamic displacements (deformations)
- $\{u_x\}$ = a vector of extra aerodynamic points used to describe, e.g., aerodynamic control surface deflections and overall rigid body motions
- $\{w^g_j\}$ = represents an initial static aerodynamic downwash. It includes, primarily, the static incidence distribution that may arise from an initial angle of attack, camber, or washout (twist)
- $[D_{jk}]$ = a substantial derivative matrix for the aerodynamic displacements. This is the $D_{1jk}$ term of **Eq. 5-63**. The $D_{2jk}$ term is not used for this quasi-steady analysis.
- $[D_{jx}]$ = a substantial derivative matrix for the extra aerodynamic points

For more information, see **Aerodynamic Theories**.

## Static Aeroelastic Equations of Motion

The aerodynamic forces are transferred to the structure reduced to the a-set to form an aerodynamic influence coefficient matrix, $Q_{aa}$, which provides the forces at the structural grid points due to structural deformations, i.e.,

$$\{F_a\} = [Q_{aa}]\{u_a\}$$

and a second matrix, $Q_{ax}$, which provides forces at the structural grid points due to unit deflections of the aerodynamic extra points, $\{u_x\}$:

$$\{F_x\} = [Q_{ax}]\{u_x\}$$

The complete equations of motion in the a-set degrees-of-freedom require

| | |
|---|---|
| $[K_{aa}]$ | Structural stiffness matrix |
| $[M_{aa}]$ | Structural mass matrix |
| $\{P_a\}$ | Vector of applied loads (e.g., mechanical, thermal, and gravity loads plus aerodynamic terms due to user input pressures and/or downwash velocities) |

The a-set equations are then:

$$[K_{aa} - \bar{q}\ Q_{aa}]\{u_a\} + [M_{aa}]\{\ddot{U}_a\} = \bar{q}[Q_{ax}]\{u_x\} + \{P_a\}$$

**Equation 5-65.**

This is the basic set of equations used for static aeroelastic analysis. In the general case, rigid body motions are included in the equations to represent the free-flying characteristic of an air vehicle. This is addressed in Simcenter Nastran by a requirement that you identify reference degrees-of-freedom equal in number to the number of rigid body motions using the SUPORT Bulk Data entry. **Eq. 5-65** is then partitioned into r-set (supported) and *l*-set (left over) degrees-of-freedom, yielding

$$\begin{bmatrix} K^a_{ll} & K^a_{lr} \\ K^a_{rl} & K^a_{rr} \end{bmatrix} \begin{Bmatrix} u_l \\ u_r \end{Bmatrix} + \begin{bmatrix} M_{ll} & M_{lr} \\ M_{rl} & M_{rr} \end{bmatrix} \begin{Bmatrix} \ddot{u}_l \\ \ddot{u}_r \end{Bmatrix} = - \begin{bmatrix} K^a_{lx} \\ K^a_{rx} \end{bmatrix} \{u_x\} + \begin{Bmatrix} P_l \\ P_r \end{Bmatrix}$$

**Equation 5-66.**

where the notation

$$[K^a_{aa}] = [K_{aa} - \bar{q}\ Q_{aa}]$$
$$[K^a_{ax}] = -\bar{q}[Q_{ax}]$$

has been introduced.

At this point the Simcenter Nastran implementation of aeroelastic analysis introduces a mathematical technique that is based on the Simcenter Nastran inertia relief analysis without aeroelastic effects. The technique entails multiplying the first row of **Eq. 5-66** by $D^T$ and adding the result to the second row. Assuming that a steady-state condition exists, the accelerations may be constrained and the system may be solved for steady-state loads or divergence. The stability derivatives and static control system coefficients may also be obtained.

## Flutter Solution Techniques

Flutter is the dynamic aeroelastic stability problem. It can be solved in any speed regime simply by selecting the appropriate aerodynamic theory. In the linear case assumed throughout this guide, the solution involves a series of complex eigenvalue solutions; the eigenvalue problem to be solved depends on the way in which the aerodynamic loads are included in the equations of motion or whether certain damping terms are included.

The manner in which the aerodynamic loads are included depends on how the dimensionless oscillatory aerodynamic coefficients are defined. When Theodorsen (1935) first developed the American method

(K-method) of flutter analysis, he introduced the aerodynamics into a vibration analysis as complex inertial terms and the flutter analysis became a vibration analysis requiring complex arithmetic. At the same time, he introduced an artificial complex structural damping, proportional to the stiffness, to sustain the assumed harmonic motion. Flutter analysis is then a double eigenvalue problem in frequency and velocity, and an iterative solution, using the reduced frequency of the assumed harmonic motion as the iteration parameter, leads to the neutrally stable conditions (flutter frequencies and velocities) at which no artificial damping is required. The artificial damping is therefore seen not to be physically meaningful, other than, perhaps, at speeds near flutter speeds.

## Generalized Aerodynamic Matrices

**Eq. 5-67** defines an aerodynamic influence coefficient matrix $Q_{kk}$ that is computed based on the aerodynamic model.

$$[Q_{kk}] = [S_{kj}][A_{jj}]^{-1}[D^1_{jk} + ik\ D^2_{jk}]$$

**Equation 5-67.**

For more information, see **Aerodynamic Theories**.

In order for this matrix to be useful in a flutter analysis, two transformations must take place:

1. The matrices must be applied to the structural model using the spline techniques.

2. A modal reduction must be applied to obtain the matrices in generalized form.

Mathematically, those transformations can be expressed as

$$[Q_{ii}] = [\phi_{ai}]^T[G_{ka}]^T[WTFACT][Q_{kk}][G_{ka}][\phi_{ai}]$$

**Equation 5-68.**

where:

| | | |
|---|---|---|
| $Q_{ii}$ | = | the generalized aerodynamic matrix |
| $\varphi_{ai}$ | = | a matrix of i-set normal mode vectors in the physical a-set |
| $G_{ka}$ | = | the spline matrix reduced to the a-set |
| *WTFACT* | = | a weighting factor matrix $W_{kk}$ defined by the user |

A level of complexity is added if the flutter analysis includes the use of extra points. Extra points are used for the representation of control systems and are therefore required in aeroservoelastic analyses. The flutter analysis then uses a merged matrix

$$[Q_{hh}] = \begin{bmatrix} Q_{ii} & Q_{ie} \\ 0 & 0 \end{bmatrix}$$

**Equation 5-69.**

in which the h-set is a combination of the i-set normal modes and the e-set extra points. It is seen that the lower e-set rows in the matrix are null. Physically, this indicates that the normal mode deflections do not produce aerodynamic forces on the extra points ($Q_{ei} = 0$) and that the extra point deflections do not produce aerodynamic loads on the extra points ($Q_{ei} = 0$).

## The K-Method of Flutter Solution

The basic equation for modal flutter analysis by the K-method is

$$\left[-M_{hh}\omega^2 + iB_{hh}\omega + (1 + ig)K_{hh} - \left(\frac{1}{2}\rho V^2\right)Q_{hh}(m,k)\right]\{u_h\} = 0$$

**Equation 5-70.**

where:

| | | |
|---|---|---|
| $M_{hh}$ | = | modal mass matrix, usually (but not necessarily) diagonal |
| $B_{hh}$ | = | modal damping matrix |
| $K_{hh}$ | = | modal stiffness matrix, usually (but not necessarily) diagonal; may be complex (with actual structural damping); will be singular if there are rigid body modes |
| $m$ | = | Mach number |
| $k$ | = | reduced frequency = $.\omega\bar{c}/2V$ |
| $\bar{c}$ | = | reference length |
| $Q_{hh}$ ($m$, $k$) | = | aerodynamic force matrix, which is a function of parameters $m$ and $k$ |
| $\omega$ | = | circular frequency = $2\pi f$ |
| $g$ | = | artificial structural damping |
| $\rho$ | = | fluid density |
| $V$ | = | overall forward velocity |
| $u_h$ | = | modal amplitude vector, sometimes called modal participation factors |

Note that $k$, $V$, and $\omega$ are not independent.

For the K-method of solution, the aerodynamic term is converted to an equivalent aerodynamic mass

$$\left[-\left[M_{hh}+\frac{\rho}{2}\left(\frac{\bar{c}}{2k}\right)^2 Q_{hh}(m,k)\right]\frac{\omega^2}{1+ig}+B_{hh}\frac{i\omega}{\sqrt{1+ig}}+K_{hh}\right]\{u_h\} = 0$$

**Equation 5-71.**

The term involving $B_{hh}$ in **Eq. 5-71** has been multiplied by $\sqrt{1+ig}$ for mathematical convenience, and is valid only at flutter, i.e., when $g = 0$. **Eq. 5-71** is solved as an eigenvalue problem for a series of values for parameters $m$, $k$, and $\rho$. The complex eigenvalue is $\omega^2 / (1 + ig)$, which can be interpreted as real values of $\omega$ and $g$. The velocity, $V$, is recovered from $V = \omega\bar{c}/2k$. Flutter occurs for values of $m$, $k$, and $\rho$ for which $g = 0$. The solutions are not valid except when $g = 0$, since the aerodynamic force terms are valid only for sinusoidal motion and $g$ is not a physical damping.

The K-method of flutter analysis is a looping procedure. The values of $V$, $g$, and $f = \omega/2\pi$ are solved for various values of $m$, $k$, and $\rho$. Plots of $V$ versus $g$ can be used to determine the flutter speed(s) (where $g$ goes through zero to positive values). The KE-method and the PK-method are the other major flutter options. These are discussed in the *Simcenter Nastran Aeroelastic Analysis User's Guide*. Typical flutter plots are shown in **Figure 5-4** and **Figure 5-5** for the output from the PK-method.

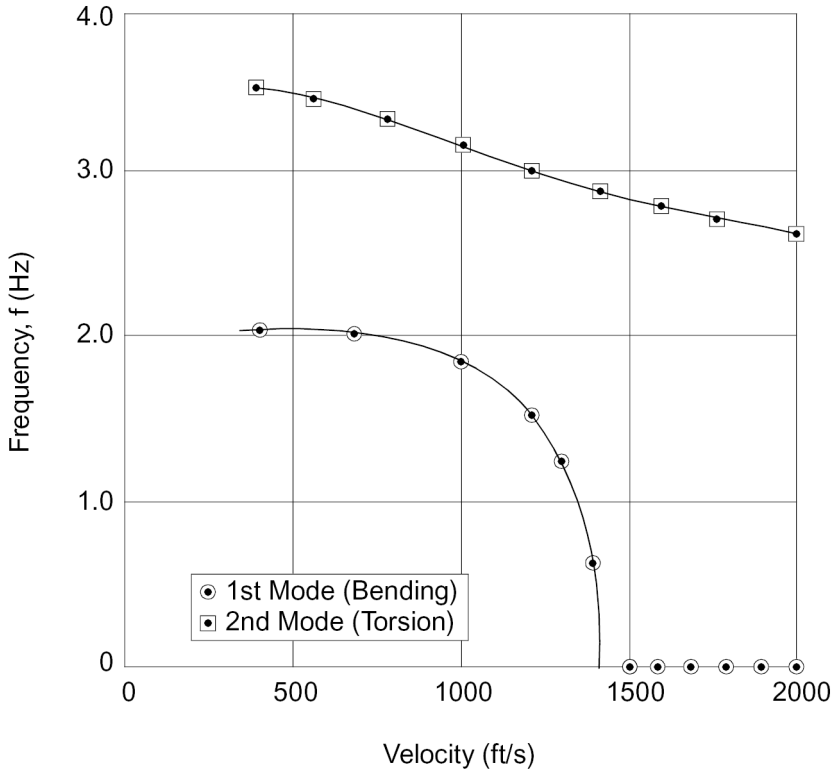



**Figure 5-4. V-f Curve for BAH Wing**

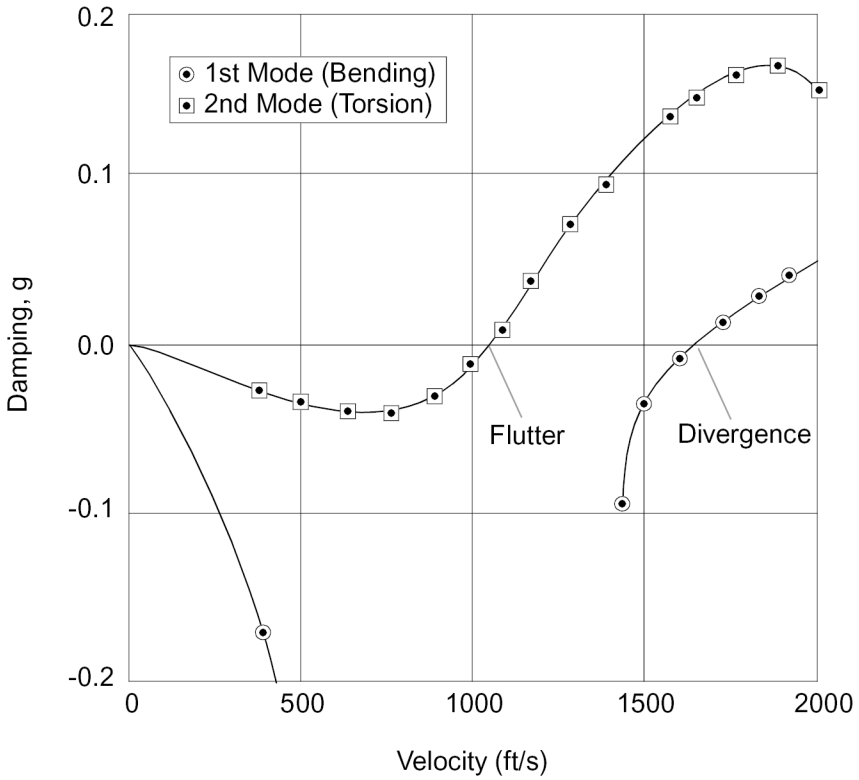



**Figure 5-5. V-g Curve for BAH Wing**

Note that the bending branch goes to zero frequency before the mode goes unstable.

## Dynamic Aeroelastic Analysis

Dynamic aeroelasticity differs from the flutter analysis described in the previous section in that the right-hand side of **Eq. 5-72** is no longer zero.

$$\left[-M_{hh}\omega^2 + iB_{hh}\omega + (1+ig)K_{hh} - \left(\frac{1}{2}\rho V^2\right)Q_{hh}(m,k)\right]\{u_h\} = 0$$

**Equation 5-72.**

For more information, see **The K-Method of Flutter Solution**.

Instead, loading, which can be in either the frequency or the time domain, is applied. For both types of loading, Simcenter Nastran performs the primary analyses in the frequency domain. If the user has supplied loadings in the time domain, Fourier Transform techniques are used to convert the loadings into the frequency domain, a frequency response analysis is performed, and the computed quantities are transformed back to the time domain using Inverse Fourier Transform techniques.

This section first describes the frequency response analysis that is the basis of all Simcenter Nastran dynamic aeroelastic analysis and then discusses the special topics of transient response analysis and random response analysis.

Simcenter Nastran performs aeroelastic frequency response analysis in modal coordinates with a basic equation of the form:

$$\left[-M_{hh}\omega^2 + iB_{hh}\omega + (1+ig)K_{hh} - \frac{1}{2}\rho V^2 Q_{hh}(m,k)\right]\{u_h\} = \{P(\omega)\}$$

**Equation 5-73.**

where all terms on the left-hand side are identical to those of **Eq. 5-72** and are defined with that equation. The right-hand side provides the loading in modal coordinates, which can be aerodynamic or nonaerodynamic in nature and is a function of the analysis frequency. Nonaerodynamic generalized loads, designated PHF(ω), are obtained in the standard fashion from the loadings applied to physical coordinates.

The solution of **Eq. 5-73** entails solving for the generalized displacements by decomposition/forward-backward substitution techniques applied to the coupled set of complex equations. Because modal reduction techniques have been applied, the solution costs are typically modest. Once the generalized displacements have been computed, standard data recovery techniques can be used to determine physical displacements, velocities, stress, etc.

## Aeroelastic Transient Response Analysis

As discussed in the introduction to this section, Aeroelastic Transient Analysis relies on Fourier transform techniques. Transient analysis by a Fourier transformation is separated into three phases. First, the loads (defined as a function of time) are transformed into the frequency domain. Second, the responses are computed in the frequency domain using the algorithm of the preceding subsection. Third, these responses (in the frequency domain) are transformed back to the time domain.

### Transformation of Loads

The user specifies loads in the same manner as given in "**Modal Versus Direct Transient Response** " in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*. The two general forms are the tabular, piecewise linear function and the general purpose function.

These loads, which are in the form required for frequency response, are transformed to the modal coordinates exactly as in the modal frequency response method.

### Inverse Transformation of the Response

The response is found from a numerical approximation to the inverse Fourier integral or from the Fourier series, the Fourier series result, which can be regarded as a special form of approximation to the integral. The quantity $\tilde{u}(\omega)$ is first calculated at a set of frequencies, $\omega_i$, by the frequency response

analysis. The $\omega_i$ do not need to be equally spaced and the integral is evaluated only over the frequency range for which the frequency response has been performed. This option is described in **"Fourier Transform"** .

### Random Response Analysis

The major loads to which an aerospace vehicle is subjected can be predicted for the most part from its design mission and maneuvering requirements. However, the total environment cannot be predicted exactly and statistical methods based on the theory of random processes must be employed to complete the description. Examples of random processes in aeroelasticity include response to atmospheric gusts and to aerodynamic buffeting.

The random process theory considered in Simcenter Nastran is based on generalized harmonic analysis, i.e., frequency response techniques, and assumes that the system is linear and that both the excitation and response are stationary with respect to time. See **"Random Analysis with Coupled Excitations"** for a description of the methods.

## 5.9 Dynamic Optimization and Design Sensitivity

Design optimization is a major area of innovation in the development of Simcenter Nastran. It provides comprehensive structural optimization capabilities that allow you to automatically design a structure while considering limits on the structure's static, modal and buckling responses. It also includes advanced capabilities, such as superelement optimization, aeroelastic optimization, shape optimization, and dynamic optimization. All of these capabilities can be made to couple with the same design task. You can use a mixture of design variables (i.e., shape and sizing) while constraining or minimizing responses from a number of analysis disciplines, such as statics, normal modes, flutter and modal frequency response.

The specific dynamic analyses that are supported are:

- direct frequency response (i.e., the analysis takes place in physical coordinates)
- modal frequency response
- modal transient response (i.e., the analysis model is reduced using modal approximations)

The basis for performing the dynamic optimization is the dynamic sensitivity analysis described in the *Simcenter Nastran Design Sensitivity and Optimization User's Guide*.

Note: Eigenvector sensitivities are not computed as part of the sensitivity analysis. Instead, the response sensitivities are computed directly from existing mode shapes. As a rule of thumb, it is recommended that sensitivity and optimization tasks retain at least twice as many modes as are felt to be adequate for the analysis.

## User Interface

This section briefly describes how to use the design optimization capability and documents the types of responses that are available for the dynamic process. Again, see the *Simcenter Nastran Design Sensitivity and Optimization User's Guide* for complete details on the use of the capability.

## Case Control Commands

- Subcases are used for multiple excitations and environments.
- The objective function is a constraint set selected by the user.
- The user selects the active design constraint set for a particular subcase.

## Bulk Data Inputs

### Example Response Definition

The DRESP1 Bulk Data entry defines a set of structural responses that is used in the design either as constraints or as an objective. As an example, the full Bulk Data description is included below.

### Format:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| DRESP1 | ID | LABEL | RTYPE | PTYPE | REGION | ATTA | ATTB | ATT1 | |
| | ATT2 | -etc.- | | | | | | | |

### Example:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| DRESP1 | 1 | DX1 | STRESS | PROD | 2 | 3 | 15 | 102 | |
| | 103 | | | | | | | | |

| Field | Contents |
|---|---|
| ID | Unique entry identifier. (Integer > 0) |
| LABEL | User-defined label. (Character) |
| RTYPE | Response type. See table below. (Character) |
| PTYPE | Element flag (PTYPE = "ELEM") or property entry name. Used with element type responses (stress, strain, force, etc.) to identify the property type, since property entry IDs are not unique across property types. (Character: "ELEM", "PBAR", "PSHELL", etc.) |
| REGION | Region identifier for constraint screening. (Integer > 0) |
| ATTA, ATTB, ATTi | Response attributes. (Integer > 0 or Real or blank) |

The DRESP1 entry shown above is used to identify the responses that are to be used in the design task. Responses are used for constraints on the design or as objective functions either directly or through the

use of user-supplied equations. The RTYPE field on this entry selects the particular response type involved. There are ten optional RTYPE's available with dynamic analysis:

| | |
|---|---|
| **[FRDISP]** | Displacement in a frequency response analysis. |
| **[FRVELO]** | Velocity in a frequency response analysis. |
| **[FRACCL]** | Acceleration in a frequency response analysis. |
| **[FRSTRE]** | Stress in a frequency response analysis. |
| **[FRFORC]** | Force in a frequency response analysis. |
| **[TDISP]** | Displacement in a transient response analysis. |
| **[TVELO]** | Velocity in a transient response analysis. |
| **[TACCL]** | Acceleration in a transient response analysis. |
| **[TSTRE]** | Stress in a transient response analysis. |
| **[TFORC]** | Force in a transient response analysis. |

A response may be associated with a specific time step, frequency, or subcase—or it may be defined for all available solutions. In the latter case, an individual constraint is generated for every available time step, etc. The specification of location of the responses (e.g., grid ID or stress component) is performed using the ATTi fields on the entry. Note that no distinction is made between modal and direct frequency analysis. Instead, the user specifies the type of analysis by using Case Control requests.

## The DRESP2, DEQATN, DCONSTR and DESOBJ Entries

Three additional Bulk Data entries are used for defining the design objective and constraints:

1. The DRESP2 entry, coupled with the DEQATN entry, can be used to synthesize a response based on DRESP1 responses and a user supplied equation. An example of the use of this feature is to specify a response that is the difference of two FRDISP responses. A synthesized response will be calculated for all time steps or frequencies defined on the DRESP1 data.

2. The DCONSTR entry is used to place lower and upper bounds on the DRESP1 and/or DRESP2 responses.

3. The DESOBJ entry identifies the objective function that is to be minimized by referring to a DRESP1 or DRESP2 entry. The objective must be a scalar quantity such as mass, error, or cost.

On the other hand, a single DRESPi entry may generate many responses at each frequency or time step the user specifies. Therefore, if the user invokes one of the dynamic response quantities as the objective, it is necessary to identify which of the frequencies or time steps is to be used for the objective.

## Example: A Clamped-Free Plate

This example is a plate that is acted upon by a harmonically varying pressure loading (see the figure **The Clamped-Free Plate**). The structural model is the same as the one used in Reference 3 to validate the sensitivity analysis in the presence of dynamic loads. The plate is rectangular with a length of 20.0 inches and a width of 10.0 inches. It is clamped at three edges and free on the fourth. Since the

pressure loading is uniform, only a one-half model with a symmetry boundary condition is required. The material has an elastic modulus of $1.0 \times 10^7$ and a density of 0.1 lbm/in$^3$. The uniform pressure load has a magnitude of 1.0 lbf/in$^2$ and is applied at 500 Hz.

## Finite Element Model

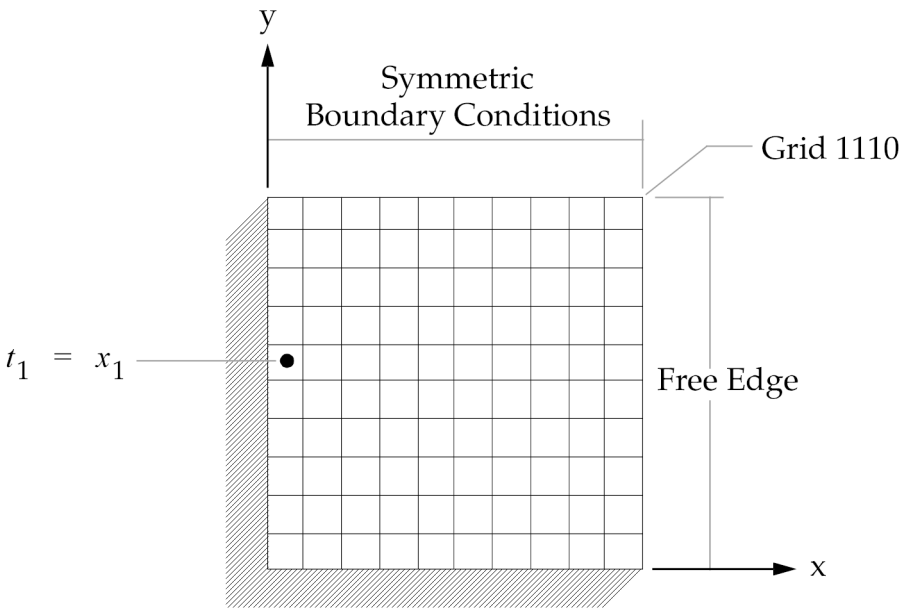



**Figure 5-6. The Clamped-Free Plate**

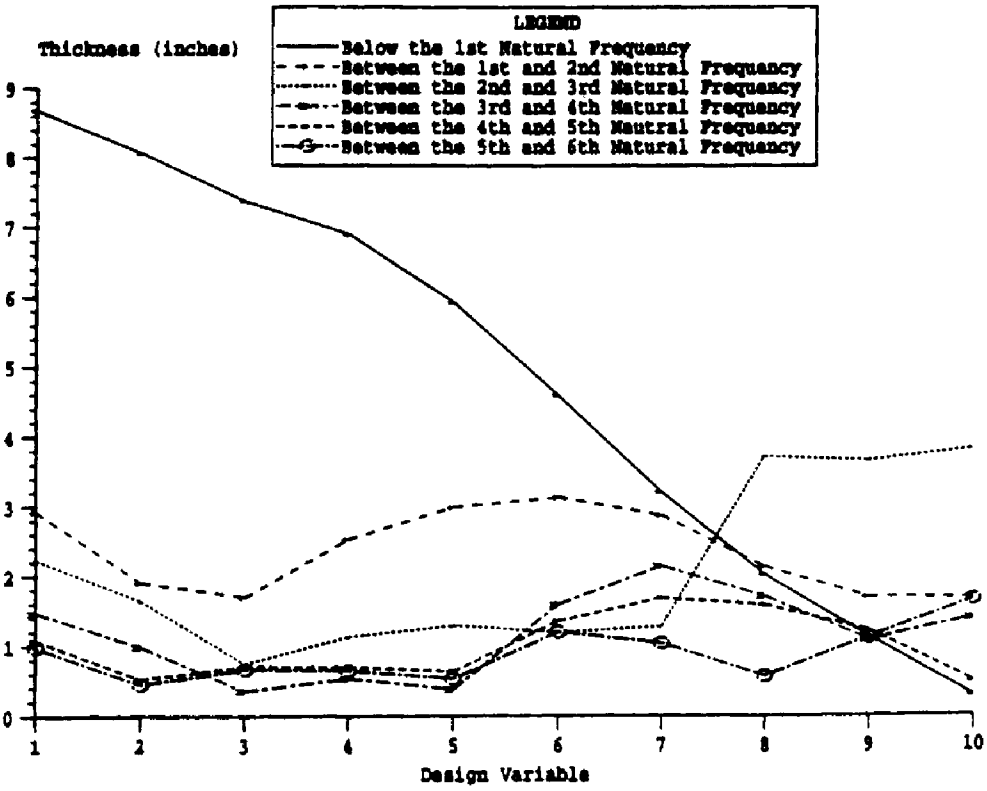



**Figure 5-7. Final Thickness Distributions for the Clamped-Free Plate as a Function of the Imposed Frequency Constraints**

The half model of the plate is modeled by a 10 × 10 array of four-noded plate elements. The elements are linked so that a single design variable controls a band of elements. That is, all 19 elements that are situated along the edge of the plate are controlled by the first design variable, the 17 elements that are one element in from the edge are controlled by the second design variable and, finally, the single element at the center is controlled by the tenth design variable. The magnitude of the displacement at plate center is constrained to be less than 2.5 inches and the plate thickness is constrained to be greater than 0.01 inches.

Six different cases were run with the only difference being a constraint on the relation between the natural frequencies and the excitation frequencies. Each test case was a separate Simcenter Nastran execution, which resulted in six different optimal designs. In the first case, the first natural frequency was constrained to be greater than the 500.0 Hz excitation frequency. For the remaining five cases:

$$\omega_i < 500.0 \text{ Hz} < \omega_{i+1} \qquad i = 1, 5$$

**Equation 5-74.**

This frequency constraint was imposed to remove the disjoint design space phenomenon that occurs when a local minima exists near the original modal frequencies. In other words, we are forcing a major change in the modal frequency.

### Results

The figure **Final Thickness Distributions for the Clamped-Free Plate as a Function of the Imposed Frequency Constraints** shows the thickness distributions along either edge of the plate for the six cases. In this example, only in the first test, where the first natural frequency is greater than the excitation frequency, is the result intuitive. **Table 5-45** shows the final weights of the six designs. It can be seen that the weight is monotonically decreasing as the structure is made more flexible (i.e., more modes are below the fixed excitation frequency).

**Table 5-45. Final Weights for the Clamped-Free Plate**

| Test | Constraint | Weight |
|---|---|---|
| 1 | 500.0 Hz. $< \omega_1$ | 5.163 |
| 2 | $\omega_1 <$ 500.0 Hz. $< \omega_2$ | 1.963 |
| 3 | $\omega_2 <$ 500.0 Hz. $< \omega_3$ | 1.831 |
| 4 | $\omega_3 <$ 500.0 Hz. $< \omega_4$ | 1.310 |
| 5 | $\omega_4 <$ 500.0 Hz. $< \omega_5$ | 0.740 |
| 6 | $\omega_5 <$ 500.0 Hz. $< \omega_6$ | 0.613 |

## Example: A Portal Frame with Transient Loading

The second example demonstrates the application of the optimization capabilities to transient analyses. The example is a simple portal frame acted upon by an enforced displacement at its base (**Figure 5-8**). This is a simple representation of a civil engineering structure subjected to seismic loading. The problem was obtained from **Reference 9** , which demonstrated the possibility of disjoint design spaces when working in the time domain in an analogous fashion to the disjoint design space referred to earlier in the frequency domain. The portal has the following parameters:

**Table 5-46. Parameters for the Portal Frame Example**

| PARAM | Meaning | Value |
|---|---|---|
| E | Young' Modulus | $30.0 \times 10^6$ psi |
| $\rho$ | Material density | 0.28 lbm/in$^3$ |
| $I_{min}$ | Minimum allowable inertia | 290.04 in$^4$ |
| NSM | Nonstructural mass per unit length | 10.0 lbm/in |
| $\omega_{min}$ | Minimum allowable fundamental frequency | 60 rad/sec |
| $\delta_{max}$ | Maximum allowable displacement | ±3.0 in |
| $\omega_{max}$ | Maximum allowable stress | ±3.0 ksi |
| L | Height and width of the portal | 180 in |

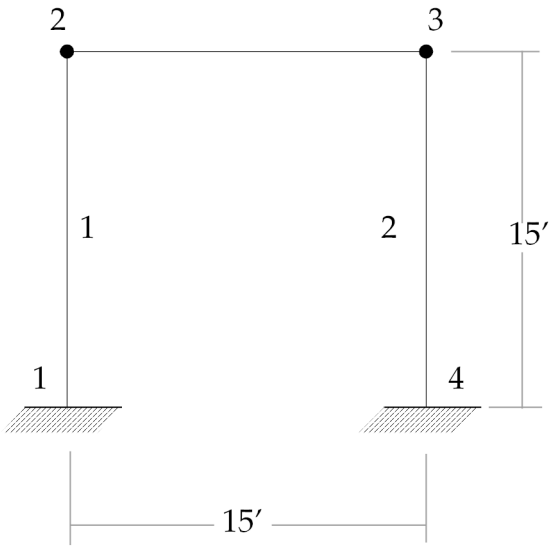



**Figure 5-8. The Portal Frame**

The frame is subjected to horizontal ground motion in the form of a half sine pulse:

$$u_g(t) = 1.0 \sin(30t), \quad t \le \pi / 30.0$$

**Equation 5-75.**

The design task is to minimize the weight of the structure while imposing the limits on structural displacement, stress and natural frequency listed in **Table 5-46**.

## Finite Element Model

The frame was modeled using three finite element bars (the symmetry of the problem could have been used to limit this to two elements). The stress is computed as the ratio of the bending moment in the beam divided by the section modulus.

The weight of the beam is simply its volume times the structural density. One design variable controls the bending moment of inertia of the vertical columns, while a second variable controls the inertia of the horizontal beam. Equations for the section modulus, S, and area, A, in terms of the moment of inertia of the beam, I, are taken from **Reference 9** in **"Selected Bibliography"** :

$$S = \sqrt{60.6I + 84000} - 290$$

**Equation 5-76.**

$$A = 0.465\sqrt{I}$$

**Equation 5-77.**

The transient response was performed over a period of 250 msecs at a time step of 1 msec. Responses were retained at every 5 msecs and the stress and displacement constraints were applied at each of the retained time steps. For the initial design, moment of inertia values of 600 in $^4$ were used for both design variables, making the initial objective value 1722.2 lbs (this ignores the weight of the nonstructural mass).

## Results

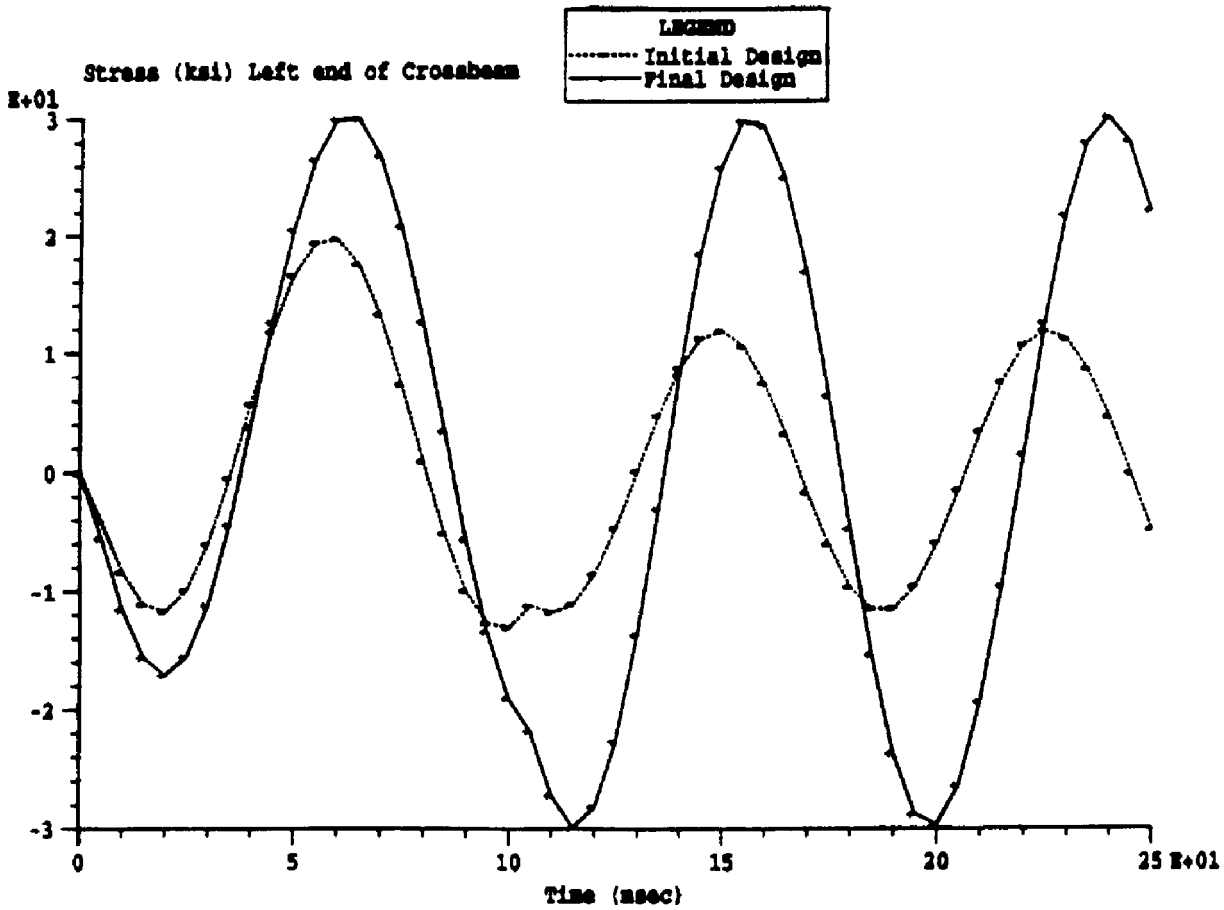



**Figure 5-9. Transient Stress Response for the Initial and Final Designs of the Portal Frame**

Figure 5-9 shows the transient stress response of the end of the horizontal beam for the initial and final designs. It can be observed that the stresses for the initial design are well below the 30 ksi limits, while the response is right at this limit for the final design. Since there is no damping in the structural representation, the structure would continue to sway if a longer record were obtained, but the imposed limits should not be exceeded.

Table 5-47 compares the results from this analysis with those given in Reference 9 in "Selected Bibliography" . Note that the Simcenter Nastran results are somewhat higher than those given in the reference, which can be attributed to differences in the finite elements used in the two analyses.

**Table 5-47. Comparison of Results for the Portal Frame**

| PARAM | Simcenter Nastran | Reference |
|---|---|---|
| X1 (columns) | 578.6 $in^4$ | 498.0 $in^4$ |
| X2 (crossbeam) | 301.7 $in^4$ | 330.0 $in^4$ |
| Objective | 1534. lbs | 1463. lbs |

Although the examples shown here are elementary, it should be apparent that the capabilities they demonstrate are quite powerful. The capabilities for dynamic response optimization can be coupled with design tasks that have requirements for static response, natural frequency, and buckling and aeroelastic responses. Note, too, that the models can include superelements. Furthermore, although all the examples here have properties for the design variables, the capability also interacts with the shape optimization capability.

One additional area in which dynamic response optimization can be applied is the system identification of structures to match modal test results. A modal test can be considered a forced excitation. Therefore, the capability described here could be used modify structural parameters in a Simcenter Nastran analysis so that the responses obtained from a Simcenter Nastran frequency analysis could match test results.

## 5.10 DDAM

The Dynamic Design Analysis Method (DDAM) is a procedure that is used to determine the shock response of equipment mounted on-board a ship to underwater explosions. The DDAM procedure consists of three distinct phases:

- Phase 1: Simcenter Nastran performs a modal analysis of the equipment structure and calculates the corresponding modal participation factors and modal effective masses for each mode.
- Phase 2: The Naval Shock Analysis (NAVSHOCK) FORTRAN program uses the modal effective masses and natural frequencies calculated by Simcenter Nastran in Phase 1 and user-supplied inputs from either a DDAM control file or command line entry to compute shock design accelerations.
- Phase 3: Simcenter Nastran uses the shock design accelerations computed by NAVSHOCK in Phase 2, and the modal participation factors and natural frequencies calculated by Simcenter Nastran in Phase 1 to compute the shock response of the structure.

Prior to NX Nastran 5, you were required to use dmap alter files to complete a DDAM procedure. Beginning with NX Nastran 5, the DDAM procedure has been implemented as a new solution sequence, SOL 187, thus eliminating the alter requirement. When you use SOL 187, you can perform a DDAM procedure:

- Automatically from a single Simcenter Nastran input file and a DDAM control file.
- Interactively from two Simcenter Nastran input files (one for Phase 1 and one for Phase 3) and either a DDAM control file or command line entry.

For the automated procedure, NAVSHOCK executes automatically as an ISHELL program after Simcenter Nastran has completed Phase 1. Once NAVSHOCK has completed Phase 2, Simcenter Nastran automatically resumes execution to complete Phase 3.

For the interactive procedure, Simcenter Nastran terminates execution once Phase 1 has completed. To execute NAVSHOCK, you must manually start it from the command line. Once NAVSHOCK has completed Phase 2, you must restart Simcenter Nastran to complete Phase 3.

## NAVSHOCK Overview

During Phase 2 of the DDAM procedure, NAVSHOCK calculates shock design accelerations for each mode found in Phase 1 of the procedure. To do so, NAVSHOCK uses the modal effective masses and natural frequencies calculated by Simcenter Nastran in Phase 1, user-supplied inputs from either a DDAM contol file or command line entry, and coefficients that are either built-in or user-supplied from an external coefficient file. NAVSHOCK converts the modal effective masses to modal effective weights using:

$$W_a = \frac{386 c_a M_a}{1000 c_f}$$

where $W_a$ is the modal effective weight (in kips) and $M_a$ is the modal effective mass for the $a^{th}$ mode. $c_f$ and $c_a$ are user-supplied force and acceleration conversion factors, respectively.

The modal effective weights are subsequently used by NAVSHOCK to calculate the corresponding reference velocities and reference accelerations. The reference velocity is calculated using:

$$V_{oa} = VF \frac{VA(VB + W_a)}{(VC + W_a)}$$

where $V_{oa}$ is the reference velocity for the $a^{th}$ mode and *VA*, *VB*, *VC*, and *VF* are user-specified or built-in coefficients.

The reference acceleration is calculated using:

$$A_{oa} = AF \frac{AA(AB + W_a)(AC + W_a)}{(AD + W_a)^2}$$

where $A_{oa}$ is the reference acceleration for the $a^{th}$ mode and *AA*, *AB*, *AC*, *AD*, and *AF* are user-specified or built-in coefficients. An alternative form of the reference acceleration equation that is sometimes used in the DDAM procedure is:

$$A_{oa} = AF\frac{AA(AB + W_a)}{(AC + W_a)}$$

To obtain this form of the reference acceleration equation, make the numerical values for the *AC* and *AD* coefficients equal.

Using the reference velocities and reference accelerations, NAVSHOCK calculates the shock design accelerations for each mode using:

$$D_a = c_a \max[386g_{\min}, \min(386A_{oa}, \omega_a V_{oa})]$$

For purposes of clarity, the shock design acceleration equation can also be expressed as:

$$D_a = \begin{cases} 386c_a A_{oa} & \omega_a > \dfrac{386A_{oa}}{V_{oa}} \\ c_a V_{oa}\omega_a & \dfrac{386g_{\min}}{V_{oa}} \le \omega_a \le \dfrac{386A_{oa}}{V_{oa}} \\ 386c_a g_{\min} & \omega_a < \dfrac{386g_{\min}}{V_{oa}} \end{cases}$$

where for the $a^{th}$ mode, $D_a$ is the shock design acceleration, $V_{oa}$ is the reference velocity, $A_{oa}$ is the reference acceleration, and $\omega_a$ is the natural frequency (in rad/sec). $g_{min}$ is the user-specified minimum acceleration value (in G's).

For additional information regarding the force and acceleration conversion factors, see **Unit Conversion Factors**.

The NAVSHOCK program is included in the Simcenter Nastran installation at /scnas/*arch*/ddam.exe.

## DDAM Automated Procedure

When you use SOL 187 to perform the DDAM procedure automatically from a single Simcenter Nastran input file:

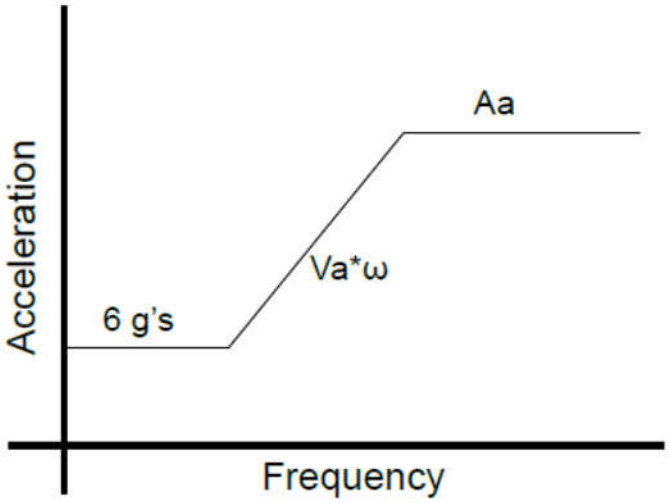
1. Simcenter Nastran performs a modal analysis of the equipment structure and calculates modal participation factors and modal effective masses for each mode. Simcenter Nastran then writes the modes, modal participation factors, and modal effective masses to an ASCII OUTPUT4 file.

2. The NAVSHOCK program automatically executes as an ISHELL program and reads the OUTPUT4 file created by Simcenter Nastran and a user-supplied DDAM control file that contains various runtime options. Depending on the contents of the DDAM control file, NAVSHOCK may also read a user-supplied external coefficient file. NAVSHOCK then computes the shock design accelerations and writes them to an INPUTT4 file.
   For more information regarding the Simcenter Nastran OUTPUT4 file, see **OUTPUT4 File Contents**.

3. Simcenter Nastran automatically resumes and reads the INPUTT4 file created by NAVSHOCK. Then Simcenter Nastran calculates the shock response for the structure and outputs the results for post-processing.

**Source Criterion Table output in .f06 file**

During Phase 2 of the DDAM procedure, NAVSHOCK calculates the following acceleration versus frequency curve.




Then for each mode found in Phase 1, NAVSHOCK uses the curve to select the appropriate shock design criteria:

$D_a$=*min (6g, acceleration, velocity)*.

The following **Source Criterion Table** is written to the .f06 file during Phase 2 when you perform the "DDAM Automated Procedure" from a single Simcenter Nastran input file. The table shows the source criterion NAVSHOCK selected for each mode.

SOURCE CRITERION TABLE (SORTED BY WEIGHT)

| MODE | FREQ(HZ) | PARTICPN FAC | ACCEL (G) | SOURCE CRITERION |
|---|---|---|---|---|
| 4 | 9.988 | 2.70E+00 | 6.00 | MINIMUM-G |
| 22 | 31.684 | 9.02E-01 | 12.06 | VELOCITY |
| 2 | 5.368 | 9.02E-01 | 6.00 | MINIMUM-G |
| 49 | 72.441 | -7.65E-01 | 27.76 | VELOCITY |
| 13 | 20.942 | -6.33E-01 | 8.07 | VELOCITY |
| 92 | 120.061 | 4.69E-01 | 46.53 | VELOCITY |
| 137 | 165.724 | -4.04E-01 | 49.30 | ACCELERATION |
| 208 | 231.819 | 3.60E-01 | 49.45 | ACCELERATION |
| 20 | 29.617 | 2.37E-01 | 11.54 | VELOCITY |
| 312 | 291.206 | -2.16E-01 | 49.80 | ACCELERATION |
| 142 | 169.473 | -2.10E-01 | 49.81 | ACCELERATION |
| 16 | 25.691 | 2.06E-01 | 10.01 | VELOCITY |
| 202 | 226.968 | -2.00E-01 | 49.83 | ACCELERATION |
| 25 | 33.813 | 1.98E-01 | 13.18 | VELOCITY |

## DDAM Interactive Procedure

For security reasons, you may be required to run NAVSHOCK on a separate, secure computer or enter runtime options manually. In such situations, you can run SOL 187 interactively using two Simcenter Nastran input files – one for the initial modal analysis run and another for the restart run. When you use SOL 187 to perform the DDAM procedure interactively:

1. Simcenter Nastran performs a modal analysis of the equipment structure and calculates modal participation factors and modal effective masses for each mode. Simcenter Nastran then writes the natural frequencies, modal participation factors, and modal effective masses to an ASCII OUTPUT4 file. At this point, execution of Simcenter Nastran terminates.

2. You then must initiate execution of NAVSHOCK by entering either `ddam.exe 'filename.inp'` or `ddam.exe` on the command line.

   If you enter `ddam.exe 'filename.inp'`, NAVSHOCK reads the DDAM control file named `'filename.inp'` and then locates and reads the OUTPUT4 file created by Simcenter Nastran during the modal analysis run. Depending on the contents of the DDAM control file, NAVSHOCK may also read a user-supplied external coefficient file. NAVSHOCK then computes the shock design accelerations and writes them to an INPUTT4 file.

   If you enter `ddam.exe`, NAVSHOCK prompts you to enter runtime options at the command prompt. The runtime options you are prompted to enter would otherwise be included in the DDAM control file. Depending on the runtime options you enter, NAVSHOCK may read a user-supplied external coefficient file. NAVSHOCK then reads the OUTPUT4 file created by Simcenter Nastran during the modal analysis run, computes the shock design accelerations, and writes them to an INPUTT4 file.

3. After NAVSHOCK has finished executing, you must manually initiate execution of Simcenter Nastran using the restart run input file. Simcenter Nastran then reads the INPUTT4 file created by

NAVSHOCK, calculates the shock response for the structure, and outputs the results for post-processing.

For more information regarding the Simcenter Nastran OUTPUT4 file, see **OUTPUT4 File Contents**.

## DDAM Input Requirements

The exact input requirements to perform the DDAM procedure depend largely on whether you are using the automated procedure or the interactive procedure. When you use the automated procedure, the entire DDAM procedure is performed from a single Simcenter Nastran input file and a DDAM control file. When you use the interactive procedure, two Simcenter Nastran input files are required (one for the initial modal analysis and another for the restart run), and the control data for NAVSHOCK is entered from either a DDAM control file or the command line.

### Simcenter Nastran Input File Requirements for Automated Procedure

In general, a Simcenter Nastran input file for an automated DDAM procedure is identical to that for SOL 103 modal analysis except as follows:

- You must include three ASSIGN statements in the file management section.

  Assign Unit 11 to the OUTPUT4 file to which Simcenter Nastran writes modal data during the initial phase of the DDAM procedure. The OUTPUT4 file is later read by the NAVSHOCK program. The following ASSIGN statement names the OUTPUT4 file `'filename.f11'`. The DELETE describer tells Simcenter Nastran to delete any existing version of `'filename.f11'` and replace it with the one generated in the current run.

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

  Assign Unit 13 to the INPUTT4 file that stores the results from the NAVSHOCK run. During the final phase of the DDAM procedure, Simcenter Nastran reads this file. The following ASSIGN statement names this INPUTT4 file `'filename.f13'`. The DEFER describer tells Simcenter Nastran not to check for the existence of this file at the start of the Simcenter Nastran job. This keyword is needed because this INPUTT4 file is not created until the NAVSHOCK run has completed during the second phase of the DDAM procedure.

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

  Assign Unit 21 to a second INPUTT4 file that is the DDAM control file. The DDAM control file contains various runtime options that are used by the NAVSHOCK program. The following ASSIGN statement assigns Unit 21 to the DDAM control file named `'filename.inp'`. The STATUS=OLD describer indicates that `'filename.inp'` has been created prior to the run.

```
ASSIGN INPUTT4='filename.inp', UNIT=21, FORM=FORMATTED STATUS=OLD
```

- You must specify SOL 187 in the executive control section rather than SOL 103.

- You must include a SUPORT bulk entry to define a single fixed base point where the excitation specified by the shock design accelerations is applied to the equipment structure being analyzed. Because the excitation is only translational, the rotational degrees-of-freedom and any unexcited translational degrees-of-freedom should be constrained using an SPC bulk entry. RBE2 rigid elements can be used to apply the excitation over a region rather than at a single point. When using an RBE2, the SUPORT bulk entry should reference the independent grid point of the RBE2. If a massless degree-of-freedom is included on the SUPORT entry, the following error message occurs: "MR MATRIX has NULL column".

### Example: Simcenter Nastran Input File for Automated DDAM Procedure

The following input file contains the instructions necessary to complete the entire DDAM procedure from a single job submittal. For conciseness, the bulk entries for grid points, elements, properties, and so on are supplied in an external file named `'filename.blk'`.

```
ASSIGN OUTPUT4='filename.f11', UNIT=11, FORM=FORMATTED, DELETE
ASSIGN INPUTT4='filename.f13', UNIT=13, FORM=FORMATTED, DEFER
ASSIGN INPUTT4='filename.inp', UNIT=21, FORM=FORMATTED, STATUS=OLD
$
SOL 187
$
CEND
$
METHOD = 1
SPC = 10
DISPLACEMENT = ALL
STRESS = ALL
$
BEGIN BULK
$
INCLUDE 'filename.blk'
$
EIGRL,1,10.,300.,10
$
SUPORT,145,123
SPC,10,145,456
$
ENDDATA
```

## Simcenter Nastran Input File Requirements for Interactive Procedure

Two Simcenter Nastran input files are required for the interactive DDAM procedure. One input file is used for the initial Simcenter Nastran run that calculates modal data for the equipment structure being analyzed. The second input file is used for the restart run when results are recovered.

For the initial run, you are required to enter the command line keyword SCRATCH = NO. Doing so ensures that the database files created during the initial run are kept for later use in the restart run. Like the input file for the automated DDAM procedure, the input file for the initial run of the interactive procedure is identical to that for SOL 103 modal analysis except as follows:

- You must include an ASSIGN statement in the file management section that assigns Unit 11 to the OUTPUT4 file to which Simcenter Nastran writes modal data during the initial phase of the DDAM procedure. The OUTPUT4 file is later read by the NAVSHOCK program. The following ASSIGN statement names the OUTPUT4 file `'filename.f11'`. The DELETE describer tells Simcenter Nastran to delete any existing version of `'filename.f11'` and replace it with the one generated in the current submittal.

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

- If database files already exist, you must include two additional ASSIGN statements in the file management section. The following ASSIGN statements delete any existing version of `'filename.master'` and `'filename.dball'` and replace them with files generated during the current submittal. The ASSIGN statements also place the files generated during the current submittal in a known location for the restart run.

```
ASSIGN MASTER='filename.master', DELETE
ASSIGN DBALL='filename.dball', DELETE
```

- You must specify SOL 187 in the executive control section rather than SOL 103.

- You must include a SUPORT bulk entry to define a single fixed base point where the excitation specified by the shock design accelerations is applied to the equipment structure being analyzed. Because the excitation is only translational, the rotational degrees-of-freedom and any unexcited translational degrees-of-freedom should be constrained using an SPC bulk entry. RBE2 rigid elements can be used to apply the excitation over a region rather than at a single point. When using an RBE2, the SUPORT bulk entry should reference the independent grid point of the RBE2. If a massless degree-of-freedom is included on the SUPORT entry, the following error message occurs: “MR MATRIX has NULL column”.

- You must include PARAM,DPREONLY,YES in the bulk data section to stop execution of the SOL 187 solution sequence once the modal data for the equipment structure has been computed and before the NAVSHOCK program executes.

A second input file is used for a restart run of Simcenter Nastran after NAVSHOCK has finished executing. The restart input file has the following requirements:

- You must include an ASSIGN statement in the file management section that assigns Unit 13 to the INPUTT4 file that stores the results from the NAVSHOCK run. During the final phase of the DDAM procedure, Simcenter Nastran reads this file. The following ASSIGN statement identifies this INPUTT4

file as `'filename.f13'`. The STATUS=OLD describer indicates that `'filename.f13'` has been created prior to the restart run.

```
ASSIGN INPUTT4='filename.f13', UNIT=13, FORM=FORMATTED, STATUS=OLD
```

- You must include a second ASSIGN statement and a RESTART statement in the file management section that directs Simcenter Nastran to use the modal data from the initial Simcenter Nastran run during the current run. The following ASSIGN and RESTART statements accomplish this requirement.

```
ASSIGN RESULT='filename.master'
RESTART LOGICAL=RESULT
```

- You must specify SOL 187 in the executive control section rather than SOL 103.
- The bulk data in the database is used for the restart run if you leave the bulk section blank. Optionally, you can clear the bulk data from the database and reenter it in the restart input file. In either case, PARAM,DPREONLY,YES must not be included in the bulk data for the restart run.

  If you use the bulk data in the database for the restart run, include the `/,n` directive after BEGIN BULK in the restart input file where `n` is the sequence number of the DPREONLY parameter. You can obtain the sequence number of the DPREONLY parameter in the sorted bulk data, which is output by default. If you specify the ECHO case control command in the initial run, you should set it to ECHO=SORT.

  If you want to clear the bulk data from the database and reenter it in the restart input file, include the `/,1,9999999` directive after BEGIN BULK in the restart input file followed by the reentered bulk entries. This entry will clear all the bulk data (including the DPREONLY parameter) from the database. The reentered bulk data must not include PARAM,DPREONLY,YES.

For additional information regarding restart runs, see the Restarts chapter of the *Simcenter Nastran User's Guide*.

### Example: Simcenter Nastran Input Files for Interactive DDAM Procedure

The following input file contains the instructions necessary to complete the initial Simcenter Nastran run of the interactive procedure. For conciseness, the bulk entries for grid points, elements, properties, and so on are supplied in an external file named `'filename.blk'`.

```
ASSIGN OUTPUT4='filename.f11', UNIT=11, FORM=FORMATTED, DELETE
ASSIGN MASTER='filename.master', DELETE
ASSIGN DBALL='filename.dball', DELETE
$
SOL 187
```

```
$
CEND
$
METHOD = 1
SPC = 10
DISPLACEMENT = ALL
STRESS = ALL
$
BEGIN BULK
$
PARAM,DPREONLY,YES
$
INCLUDE 'filename.blk'
$
EIGRL,1,10.,300.,10
$
SUPORT,145,123
SPC,10,145,456
$
ENDDATA
```

The following restart input file complements the input file for the initial run. The bulk section of the restart file is empty except for the `/,36` entry. The database created during the initial run is used for the restart run and assuming the sequence number of the DPREONLY entry is 36, the DPREONLY entry is deleted.

```
ASSIGN INPUTT4='filename.f13', UNIT=13, FORM=FORMATTED, STATUS=OLD
ASSIGN RESULT='filename.master'
RESTART LOGICAL=RESULT
$
SOL 187
$
CEND
$
METHOD = 1
SPC = 10
DISPLACEMENT = ALL
STRESS = ALL
$
BEGIN BULK
$
/,36
$
ENDDATA
```

## Simcenter Nastran Input File Miscellaneous Considerations

- To obtain an accurate representation of the modal effective masses for the shock response calculations, you must specify a sufficiently large frequency range for the modal solution.
- To ensure that the correct modal effective masses and modal participation factors are calculated, you must mass normalize the eigenvectors. If you use either an EIGR or EIGRL bulk entry to specify the real eigenvalue extraction method, mass normalization is the default scaling method.
- The number of supported case control output requests in SOL 187 is limited as compared to SOL 103. For SOL 187, you can request STRESS, FORCE, DISPLACEMENT, VELOCITY and ACCELERATION. Eigenvectors are written by default in SOL 187.
- The output from standard case control requests are written to an .f06 file by default. You can optionally request that the output be written to an .op2 file by specifying PARAM,POST,n<0 in either the case control or bulk entry sections of the input file.
- If superelements are included in the model, you must specify the SUPORT bulk entry that defines the fixed base point as part of the residual structure and in the A-set.
- Superelement reduction procedures are supported. However, NRL summed responses are only calculated for the residual structure. For information regarding NRL summing, see **NRL Summed Response**.

## DDAM Control Files

You can enter the runtime options for the NAVSHOCK program from either an ASCII DDAM control file or the command line.

- NAVSHOCK reads the runtime options from the DDAM control file when you use the automated procedure or when you initiate execution of NAVSHOCK by entering `ddam.exe 'filename.inp'` on the command line.
- NAVSHOCK prompts you for the runtime options when you initiate execution of NAVSHOCK by entering `ddam.exe` on the command line.

The DDAM control file has the following format:

1st line - spectrum control - format a1,1x,a1

1st item - coefficients from external or built-in source

T = use coefficients from external coefficient file

F = use built-in coefficients

2nd item - DDAM or response spectrum (non-DDAM) run

T = response spectrum run

F = DDAM run

2nd line - file name (if needed) -format a80

The file name of the external coefficient file if the 1st item on line 1 is T.

The file name of the external shock spectrum file if the 2nd item on line 1 is T.

If neither are T, the line is not needed and should be omitted – do not leave a blank line.

3rd line - location flags (if needed) - format i1,1x,i1,1x,i1

1st item - ship type (denoted as nsurf)

nsurf = 1 for a surface ship

nsurf = 2 for a submarine

2nd item - equipment location (denoted as nstruc)

nstruc = 1 for deck mounted

nstruc = 2 for hull mounted

nstruc = 3 for shell mounted

3rd item - coefficient type (denoted as nplast)

nplast = 1 for elastic

nplast = 2 for elastic/plastic

If the 2nd item on line 1 is T, the line is not needed and should be omitted – do not leave a blank line.

4th Line - Weight cutoff percentage $W_{cutoff}$ - format F8.3 (0. To 100.)

5th Line - Axis orientation - format a1,1x,a1

1st item - fore/aft axis - specify either X, Y, or Z

2nd item - vertical axis - specify either X, Y, or Z

6th Line - Input file name (OUTPUT4 file created by Simcenter Nastran) - format a80

7th Line - Output file name (INPUTT4 file created by NAVSHOCK) - format a80

8th Line - Verification file name - format a80

9th Line - Minimum acceleration value $g_{min}$ (in units of G's) - format F8.3 (must be >0.0)

10th Line - Force conversion factor $c_f$ (converts lbf to force units in Simcenter Nastran model) - format F8.3 (must be >0.0)

11th Line - Acceleration conversion factor $c_a$ (converts in/sec$^2$ to acceleration units in Simcenter Nastran model) - format F8.3 (must be >0.0)

> Note:
>
> Formatting of the DDAM control file is important. The entries on the spectrum control line and the axis orientation line must be capitalized and separated by a space. The entries on the location flags line must be integers separated by a space.

To perform a DDAM procedure, you can specify either a default coefficient option DDAM control file or a user coefficient option DDAM control file.

- If you specify a default coefficient option DDAM control file, NAVSHOCK uses the built-in coefficients. A default coefficient option DDAM control file requires the following entries:

F F
nsurf nstruc nplast
$W_{cutoff}$
fore/aft_axis vertical_axis
filename.f11
filename.f13
filename.ver
$g_{min}$
$c_f$
$c_a$

> Note:
>
> Because a default coefficient option DDAM control file directs NAVSHOCK to use the built-in coefficients, the 2nd line is not needed and is omitted from the control file without leaving a blank line.

- If you specify a user coefficient option DDAM control file, NAVSHOCK uses coefficients from an external coefficient file. A user coefficient option DDAM control file requires the following entries:

  T F
  filename.dat (external coefficient file)
  nsurf nstruc nplast
  $W_{cutoff}$
  fore/aft_axis vertical_axis
  filename.f11
  filename.f13
  filename.ver
  $g_{min}$
  $c_f$
  $c_a$

You can also perform a response spectrum analysis using SOL 187 with a user spectrum option control file. However, as a best practice, it is recommended that you use SOL 103 to perform a response spectrum analysis.

- A user spectrum option control file requires the following entries:

  F T
  filename.dat file (external shock spectrum file)
  $W_{cutoff}$
  fore/aft_axis vertical_axis

filename.f11
filename.f13
filename.ver
$g_{min}$
$c_f$
$c_a$

> Note:
>
> Even though NAVSHOCK does not use coefficients during a response spectrum analysis, you must still specify either `F` or `T` for the 1st item in the 1st line of the control file. However, the 3rd line is not needed and is omitted from the control file without leaving a blank line.

### Unit Conversion Factors

Prior to NX Nastran 7.1, the NAVSHOCK program required that the Simcenter Nastran model units be lbf, in, and sec for force, length, and time, respectively. Now by appropriately specifying force and acceleration conversion factors, $c_f$ and $c_a$, respectively, you can use any consistent set of units in the Simcenter Nastran model.

The force conversion factor, $c_f$, is the numerical value for force in the units of the Simcenter Nastran model that is physically equivalent to 1 lbf. Similarly, the acceleration conversion factor, $c_a$, is the numerical value for acceleration in the units of the Simcenter Nastran model that is physically equivalent to 1 in/sec$^2$.

For example, if the Simcenter Nastran model units are N, mm, and sec, $c_f = 4.448$ because 4.448 N = 1 lbf and $c_a = 25.4$ because 25.4 mm/sec$^2$ = 1 in/sec$^2$.

### Coefficients

NAVSHOCK requires coefficients (also referred to as weighting factors and directional scaling factors) to calculate reference velocities and reference accelerations. NAVSHOCK can use built-in coefficients, user-supplied coefficients from an external coefficient file, or a combination of both. For example, NAVSHOCK uses:

- Built-in coefficients if you specify a default coefficient option DDAM control file.
- User-supplied coefficients if you specify a user coefficient option DDAM control file that references an external coefficient file.
- Both built-in and user-supplied coefficients if you specify a user coefficient option DDAM control file that references either a blank external coefficient file or an external coefficient file with some of the coefficients missing. When this occurs, NAVSHOCK uses built-in coefficients in place of the missing user-supplied coefficients.

Beginning with NX Nastran 7.1, the built-in coefficients are set to values specified in NRL-1396. The following tables list the values specified in NRL-1396. In both tables:

- 'nsurf' refers to the ship type. Allowable values are SUB (submarine) and SURF (surface).
- 'nstruc' refers to the mounting location. Allowable values are DECK, HULL, and SHELL.
- 'nplast' refers to elastic or elastic-plastic factors. Allowable values are ELASTIC and ELPL.

In the following table, (1), (2), and (3) after VF and AF refer to directions:

- (1) = fore/aft
- (2) = athwartship
- (3) = vertical

| nsurf/nstruc/nplast | VF(1) | VF(2) | VF(3) | AF(1) | AF(2) | AF(3) |
|---|---|---|---|---|---|---|
| SURF/DECK/ELASTIC | 0.4 | 0.4 | 1.0 | 0.4 | 0.4 | 1.0 |
| SURF/HULL/ELASTIC | 0.2 | 0.4 | 1.0 | 0.2 | 0.4 | 1.0 |
| SURF/SHELL/ELASTIC | 0.1 | 0.2 | 1.0 | 0.1 | 0.2 | 1.0 |
| SURF/DECK/ELPL | 0.2 | 0.2 | 0.5 | 0.4 | 0.4 | 1.0 |
| SURF/HULL/ELPL | 0.1 | 0.2 | 0.5 | 0.2 | 0.4 | 1.0 |
| SUB/DECK/ELASTIC | 0.8 | 2.0 | 1.0 | 0.8 | 2.0 | 1.0 |
| SUB/HULL/ELASTIC | 0.4 | 1.0 | 1.0 | 0.4 | 1.0 | 1.0 |
| SUB/SHELL/ELASTIC | 0.08 | 0.2 | 1.0 | 0.08 | 0.2 | 1.0 |
| SUB/DECK/ELPL | 0.4 | 1.0 | 0.5 | 0.8 | 2.0 | 1.0 |
| SUB/HULL/ELPL | 0.2 | 0.5 | 0.5 | 0.4 | 1.0 | 1.0 |

| nsurf/nstruc/nplast | VA | VB | VC | AA | AB | AC | AD |
|---|---|---|---|---|---|---|---|
| SURF/DECK/ELASTIC | 30.0 | 12.0 | 6.0 | 10.0 | 37.5 | 12.0 | 6.0 |
| SURF/HULL/ELASTIC | 60.0 | 12.0 | 6.0 | 20.0 | 37.5 | 12.0 | 6.0 |
| SURF/SHELL/ELASTIC | 120.0 | 12.0 | 6.0 | 40.0 | 37.5 | 12.0 | 6.0 |
| SURF/DECK/ELPL | 30.0 | 12.0 | 6.0 | 10.0 | 37.5 | 12.0 | 6.0 |
| SURF/HULL/ELPL | 60.0 | 12.0 | 6.0 | 20.0 | 37.5 | 12.0 | 6.0 |
| SUB/DECK/ELASTIC | 10.0 | 480.0 | 100.0 | 5.2 | 480.0 | 20.0 | 20.0 |
| SUB/HULL/ELASTIC | 20.0 | 480.0 | 100.0 | 10.4 | 480.0 | 20.0 | 20.0 |
| SUB/SHELL/ELASTIC | 100.0 | 480.0 | 100.0 | 52.0 | 480.0 | 20.0 | 20.0 |
| SUB/DECK/ELPL | 10.0 | 480.0 | 100.0 | 5.2 | 480.0 | 20.0 | 20.0 |
| SUB/HULL/ELPL | 20.0 | 480.0 | 100.0 | 10.4 | 480.0 | 20.0 | 20.0 |

### Weight Cutoff Percentage

You can optionally use weight cutoff percentage to truncate the number of modes used to compute the shock response of the structure during Phase 3 of the DDAM procedure. You can specify the weight cutoff percentage from a DDAM control file, command line entry, or an external coefficient file. In addition, a default value for the weight cutoff percentage of 80.0 is included with the built-in coefficients in NAVSHOCK. The weight cutoff percentage that NAVSHOCK uses depends on the weight cutoff percentage specified in the control file or, equivalently, entered at the command prompt:

- If the weight cutoff percentage specified in the control file or entered at the command prompt is greater than $1.0 \times 10^{-3}$, the value specified in the control file or entered at the command prompt is used.
- If the weight cutoff percentage specified in the control file or entered at the command prompt is less than or equal to $1.0 \times 10^{-3}$ and a value for the weight cutoff percentage is specified in an external coefficient file, the value specified in the external coefficient file is used.
- If the weight cutoff percentage specified in the control file or entered at the command prompt is less than or equal to $1.0 \times 10^{-3}$ and an external coefficient file either does not exist or does not include a value for the weight cutoff percentage, the built-in default value of 80.0 is used.

> Note:
>
> As a best practice, you should specify the weight cutoff percentage in the DDAM control file or at the command prompt.

For additional information on how weight cutoff percentage is used in the DDAM procedure, see **NRL Summed Response**.

### Example: DDAM Control Files

The following is an example of a default coefficient option control file:

```
F F
1 1 1
75.
X Z
'filename.f11'
'filename.f13'
'filename.ver'
6.
4.448
25.4
```

The following is an example of a user coefficient option control file. This control file instructs NAVSHOCK to use the coefficients contained in an external coefficient file named `'filename.dat'`.

```
T F
'filename.dat'
1 1 1
75.
X Z
'filename.f11'
'filename.f13'
'filename.ver'
6.
1.
0.08333
```

## External Coefficient File

NAVSHOCK accesses user-supplied coefficients from an ASCII external coefficient file. The external coefficient file can also optionally include a value for weight cutoff percentage. The external coefficient file is formatted similar to a Simcenter Nastran bulk entry with each field containing eight spaces. However, character entries in each field must be left-justified.

The coefficients are defined on a COEF entry.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| COEF | nsurf | nstruc | nplast | | | | | | |
| | VF(1) | VF(2) | VF(3) | AF(1) | AF(2) | AF(3) | | | |
| | VA | VB | VC | AA | AB | AC | AD | | |

where:

- 'nsurf' refers to the ship type. Allowable values are SUB (submarine) and SURF (surface).
- 'nstruc' refers to the mounting location. Allowable values are DECK, HULL, and SHELL.
- 'nplast' refers to elastic or elastic-plastic factors. Allowable values are ELASTIC and ELPL.

and (1), (2), and (3) after VF and AF refer to directions:

- (1) = fore/aft
- (2) = athwartship
- (3) = vertical

All numerical values on a COEF entry must be entered as real numbers.

The weight cutoff percentage is defined on a CUTOFF entry.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| CUTOFF | pref | | | | | | | | |

where pref is the weight cutoff percentage. Enter pref as a real percentage, not as a decimal fraction. For example, a weight cutoff percentage of 85% must be entered as 85.0 instead of 0.85.

The external coefficient file can contain a complete or partial listing of coefficients for all possible nsurf/nstruc/nplast combinations. If the applicable nsurf/nstruc/nplast combination is missing from the external coefficient file, the built-in coefficients are used by NAVSHOCK. If the applicable nsurf/nstruc/nplast combination is present in the external coefficient file, but some of the coefficient fields are either blank or contain an asterisk (*), NAVSHOCK uses built-in coefficients to replace the missing values.

> Note:
>
> As a best practice, you should specify a complete set of coefficients in the external coefficient file when using the user coefficient option for the DDAM control file.

**Example: External Coefficient File**

The following external coefficient file contains coefficients for three nsurf/nstruc/nplast combinations and a weight cutoff percentage value.

```
$ or # indicate comment lines
$
$234567812345678123456781234567812345678123456781234567812345678
$
$ Format of COEF entry
$
$COEF   nsurf   nstruc  nplast
$       VF(1)   VF(2)   VF(3)   AF(1)   AF(2)   AF(3)
$       VA      VB      VC      AA      AB      AC      AD
$
$ Hypothetical coefficients for SURF/DECK/ELPL combination
$
COEF    SURF    DECK    ELPL
        0.3     0.45    1.0     0.27    0.6     0.9
        10.0    25.0    40.0    11.0    30.0    6.0     *
$
$ Hypothetical coefficients for SURF/HULL/ELPL combination
$
COEF    SURF    HULL    ELPL
        0.4     0.4     0.9     0.25    0.35    1.0
        4.0     10.0    38.0    38.0    45.0    6.0     15.0
$
$ Hypothetical coefficients for SURF/SHELL/ELPL combination
$
COEF    SURF    SHELL   ELPL
```

```
        0.25    0.6     1.0     0.25    0.4     1.0
                                10.0    45.0    8.0     17.0
$
$ Specify a weight cutoff percentage of 90%
$
CUTOFF  90.0
```

Note:

NAVSHOCK uses built-in coefficients when values are missing from the external coefficient file. In the example, NAVSHOCK will use built-in values for the AD coefficient in the SURF/DECK/ELPL combination, and for the VA, VB, and VC coefficients in the SURF/SHELL/ELPL combination.

## External Shock Spectrum File

When you use SOL 187 to perform a response spectrum analysis, you must define an ASCII external shock spectrum file that contains data pairs of frequency versus displacement, velocity or acceleration. For data spanning several orders of magnitude, you can optionally use a logarithmic scale.

The format for an external shock spectrum file is as follows:

```
DATATYP type    data    dir     freq    interp
BEGIN DATA
f1,data1
f2,data2
fn,datan
[BEGIN DATA]
[f1,data1]
[f2,data2]
[fn,datan]
[BEGIN DATA]
[f1,data1]
[f2,data2]
[fn,datan]
END FILE
```

The fields in the DATATYP entry have eight spaces. The character entries must be left-justified in each field and cannot be comma separated. The fields of the DATATYP entry are as follows.

| | |
|---|---|
| type | Type of data:<br>DISP (displacement), VELO (velocity), or ACCE (acceleration) |
| data | Units of displacement, velocity or acceleration:<br>G (acceleration in G's), F (displacement in ft, velocity in ft/sec, acceleration in ft/sec$^2$), I (displacement in inches, velocity in in/sec, acceleration in in/sec$^2$), or M (displacement in m, velocity in m/sec, acceleration in m/sec$^2$) |

| | |
|---|---|
| dir | Number of spectra in the file:<br>A value of 1 indicates a single spectrum will be used for all three shock directions. A value of 3 indicates there are three spectra, one for each direction. |
| freq | Units of frequency:<br>RAD (rad/sec) or HERTZ (frequency in Hz) |
| interp | Interpolation scheme for the table values:<br>LOGLOG (both axes are logarithmic), LINLIN (both axes are linear), LOGLIN (the frequency axis is logarithmic, the other is linear), or LINLOG (the frequency range is linear, the other is logarithmic) |

A BEGIN DATA entry should precede each frequency/motion data section. If the dir entry is 1, there will be one BEGIN DATA entry. If the dir entry is 3, there will be three. The data section ends with an END FILE entry.

**Example: External Shock Spectrum File**

The following external shock spectrum file contains a single frequency spectrum:

```
$ or # indicate comment lines
$
$234567812345678123456781234567812345678123456781234567812345678
$
DATATYP ACCE    I       1       HERTZ   LOGLIN
$
$ Frequency vs acceleration data pairs - acceleration in in/sec**2
$
BEGIN DATA
1.,1.
10.,1.5
100.,60.
500.,400.
1000.,130.
10000.,80.
END FILE
```

## Verification File

NAVSHOCK automatically writes a verification (.ver) file that contains a partial listing of the runtime options it used and a summary of the results it calculated in each direction.

**Example: Verification File**

The following verification file is for a NAVSHOCK run that uses a default coefficient option DDAM control file to enter the runtime options. The accelerations listed in each Modal Effective Mass table are the shock design accelerations (in G's) that correspond to each mode.

```
 COEFFECIENTS FROM FILE: (DEFAULT)
        INPUT FROM FILE: 'filename.f11'

          Cf    :     1.0000
          Ca    :     1.0000
          GMIN  :     6.0000

           ..... SURFACE SHIP .....
           ..... HULL MOUNTED STRUCTURE .....
           ..... ELASTIC-PLASTIC COEFFICIENTS .....

           ..... VERTICALLY (X) DIRECTED SHOCK .....

          MASS AVAILABLE THIS DIRECTION    85.58 PERCENT OF      1.28 .
          MASS USED THIS DIRECTION    85.58 PERCENT


           MODAL EFFECTIVE MASS TABLE

                          MODAL WEIGHT       CUMULATIVE WEIGHT
   MODE   FREQ(HZ)     WEIGHT       %      WEIGHT       %      PARTIC.
ACCEL (G)

     1     32.761       0.00    0.00       0.00    0.00    -1.20E-17
31.96
     2    390.403       0.00    0.00       0.00    0.00     8.75E-18
250.00
     3    603.928       0.07    5.74       0.07    5.74     1.38E-02
250.00
     4    793.373       0.00    0.00       0.07    5.74     1.17E-16
250.00
     5   1656.107       0.00    0.00       0.07    5.74     4.35E-16
250.00
     6   2376.412       0.00    0.00       0.07    5.74    -1.47E-15
250.00
     7   2569.216       1.02   79.84       1.10   85.58    -5.15E-02
249.94
     8   3415.784       0.00    0.00       1.10   85.58     3.30E-15
250.00
     9   4803.542       0.00    0.00       1.10   85.58    -8.31E-13
250.00
    10   5176.407       0.00    0.00       1.10   85.58    -4.32E-12
250.00

 COEFFECIENTS FROM FILE: (DEFAULT)
        INPUT FROM FILE: 'filename.f11'

          Cf    :     1.0000
          Ca    :     1.0000
```

```
          GMIN :     6.0000

           ..... SURFACE SHIP .....
           ..... HULL MOUNTED STRUCTURE .....
           ..... ELASTIC-PLASTIC COEFFICIENTS .....

           ..... FORE-&-AFT (Y) DIRECTED SHOCK .....

          MASS AVAILABLE THIS DIRECTION    91.15 PERCENT OF      1.28 .
          MASS USED THIS DIRECTION    91.15 PERCENT


           MODAL EFFECTIVE MASS TABLE

                          MODAL WEIGHT       CUMULATIVE WEIGHT
   MODE  FREQ(HZ)    WEIGHT       %      WEIGHT       %      PARTIC.
ACCEL (G)

     1     32.761       0.00    0.00       0.00    0.00
-8.76E-18      6.39
     2    390.403       0.00    0.00       0.00    0.00     1.29E-17
50.00
     3    603.928       1.06   82.88       1.06   82.88     5.25E-02
49.99
     4    793.373       0.00    0.00       1.06   82.88     1.26E-16
50.00
     5   1656.107       0.00    0.00       1.06   82.88    -1.47E-16
50.00
     6   2376.412       0.00    0.00       1.06   82.88     4.09E-16
50.00
     7   2569.216       0.11    8.28       1.17   91.15     1.66E-02
50.00
     8   3415.784       0.00    0.00       1.17   91.15    -3.28E-15
50.00
     9   4803.542       0.00    0.00       1.17   91.15     6.21E-13
50.00
    10   5176.407       0.00    0.00       1.17   91.15     2.57E-12
50.00

 COEFFECIENTS FROM FILE: (DEFAULT)
        INPUT FROM FILE: 'filename.f11'

          Cf   :     1.0000
          Ca   :     1.0000
          GMIN :     6.0000

           ..... SURFACE SHIP .....
           ..... HULL MOUNTED STRUCTURE .....
           ..... ELASTIC-PLASTIC COEFFICIENTS .....
```

```
          ..... ATHWARTSHIP (Z) DIRECTED SHOCK .....

         MASS AVAILABLE THIS DIRECTION    98.00 PERCENT OF      1.28 .
         MASS USED THIS DIRECTION    98.00 PERCENT


          MODAL EFFECTIVE MASS TABLE

                            MODAL WEIGHT       CUMULATIVE WEIGHT
   MODE  FREQ(HZ)    WEIGHT       %      WEIGHT       %      PARTIC.
ACCEL (G)

     1     32.761       1.14   89.02       1.14   89.02    -5.44E-02
12.78
     2    390.403       0.04    3.06       1.18   92.08     1.01E-02
100.00
     3    603.928       0.00    0.00       1.18   92.08     9.99E-18
100.00
     4    793.373       0.04    3.26       1.22   95.34    -1.04E-02
100.00
     5   1656.107       0.01    0.58       1.23   95.92    -4.40E-03
100.00
     6   2376.412       0.01    1.16       1.25   97.08    -6.21E-03
100.00
     7   2569.216       0.00    0.00       1.25   97.08     1.04E-16
100.00
     8   3415.784       0.00    0.17       1.25   97.25    -2.41E-03
100.00
     9   4803.542       0.01    0.54       1.25   97.80     4.25E-03
100.00
    10   5176.407       0.00    0.20       1.26   98.00     2.57E-03
100.00
```

## NRL Summed Response

During Phase 3 of the DDAM procedure, Simcenter Nastran uses the shock design accelerations computed in Phase 2, and the modal participation factors and natural frequencies calculated in Phase 1 to compute the shock response of the equipment structure. The scaled response at the $i^{th}$ location (grid point or element) for the $a^{th}$ mode is given by:

$$\Gamma_{ia} = \zeta_{ia} P_a \frac{D_a}{\omega_a^2}$$

where $\Gamma_{ia}$ is the scaled response for either stress, force, displacement, velocity, or acceleration given by $\zeta_{ia}$. $P_a$ is the modal participation factor, $D_a$ is the shock design acceleration, and $\omega_a$ is the natural frequency (in rad/sec) for the $a^{th}$ mode.

To estimate the total response at the $i^{th}$ location resulting from the cumulative contribution of each mode, Simcenter Nastran uses the NRL summing formula in SOL 187. The NRL summing formula is:

$$(\Gamma_{NRL})_i = \sqrt{(\Gamma_{ia})^2_{max}} + \sqrt{\sum_a (\Gamma_{ia})^2 - (\Gamma_{ia})^2_{max}}$$

where $(\Gamma_{NRL})_i$ is the total response at the $i^{th}$ location and $(\Gamma_{ia})_{max}$ is the largest magnitude response attributable to any single mode at the $i^{th}$ location.

Simcenter Nastran computes the NRL sum with contributions from all the modes included in the INPUTT4 file. The modes that NAVSHOCK writes to the INPUTT4 file depend on the weight cutoff percentage. If the weight cutoff percentage is 100%, NAVSHOCK writes all the modes to the INPUTT4 file. If the weight cutoff percentage is less than 100%, NAVSHOCK writes modes to the INPUTT4 file beginning with the lowest frequency mode and including each successive mode until the percentage of modal effective weight of all the modes written to the INPUTT4 file exceeds the weight cutoff percentage. Optionally, you can exclude individual modes by using Direct Matrix Input (DMI). For more information on how to use DMI to exclude individual modes from the NRL sum, see **Excluding Individual Modes**.

> Note:
>
> Little computational benefit is derived by excluding modes from the NRL sum. As a best practice, specify the weight cutoff percentage as 100% to include the contributions from all the modes in the NRL sum.

## DDAM Mode Filtering

In phase 1 of the DDAM procedure, Simcenter Nastran performs a modal analysis of the structure and calculates the corresponding participation factors and effective masses for each mode.

You can request for Simcenter Nastran to filter the modes computed in phase 1 based on their modal weight. Then for phase 2, Simcenter Nastran will send NAVSHOCK only the modes that pass the filter.

Mode filtering is important for DDAM because unnecessary modes can yield in an overly conservative design. It can also improve the solution efficiency.

You use the SELMODE bulk entry to request the filtering. There are three options to filter the modes.

- CUTOFF: The software sorts the modes in decreasing order of modal weight ($W_i$) such that $W_1$ has the largest modal weight. Then the cumulative modal weight sum ($CW_i$) is computed for each mode such that $CW_1=W_1$, $CW_2=W_1+W_2$, $CW_3=W_1+W_2+W_3$, etc. The modes with ($CW_i$ X 100% / Total Modal Weight > CUTOFF) are ignored for response computations (Real > 0.0; Default=0.0).

- PERCENT: Modes with (Modal Weight / Total Modal Weight < PERCENT) are ignored for response computations (Real > 0.0; Default=0.0).

- TOP: The top n modes with the largest modal weight are included. All other modes are ignored for response computations (Integer > 0; Default=All modes are included).

For example, the output below demonstrates how the software sorts and filters the modes.

1 If you define CUTOFF=75.0, only modes 1, 14, and 44 would be used for the response computations.

2 If you define PERCENT=.2, only modes 1, 14, 44, 15, and 82 would be used for the response computations.

3 If you define TOP=3, only modes 1, 14, and 44 would be used for the response computations.

MODAL EFFECTIVE MASS TABLE (SORTED BY WEIGHT)

| MODE | FREQ(HZ) | MODAL WEIGHT WEIGHT | % | CUMULATIVE WEIGHT WEIGHT | % | PARTIC. | ACCEL (G) |
|---|---|---|---|---|---|---|---|
| 1 | 3.478 | 2719.56 | 54.38 | 2719.56 | 54.38 | -2.65E+00 | 6.00 |
| 14 | 23.295 | 622.58 | 12.45 | 3342.14 | 66.83 | 1.27E+00 | 17.33 |
| 44 (3) | 62.765 | 298.59 | 5.97 | 3640.73 | 72.81 (1) | -8.79E-01 | 47.83 |
| 15 | 25.383 | 175.41 | 3.51 | 3816.14 | 76.31 | -6.74E-01 | 19.53 |
| 82 | 109.401 | 105.15 | 2.10 (2) | 3921.29 | 78.42 | 5.22E-01 | 84.65 |
| 139 | 166.168 | 68.35 | 1.37 | 3989.65 | 79.78 | -4.21E-01 | 98.50 |
| 219 | 238.527 | 60.96 | 1.22 | 4050.61 | 81.00 | -3.97E-01 | 98.66 |
| 16 | 25.691 | 36.73 | 0.73 | 4087.34 | 81.74 | 3.08E-01 | 19.99 |
| 159 | 185.535 | 21.78 | 0.44 | 4109.12 | 82.17 | -2.37E-01 | 99.52 |
| 302 | 288.835 | 19.17 | 0.38 | 4128.29 | 82.56 | 2.23E-01 | 99.57 |

The software filters the modes for each translational DOF (X, Y, Z) independently. For example, a mode that has a significant modal weight in the X-DOF may be kept for the NAVSHOCK X-DOF computation, but the same mode may have a small modal mass in the Y-DOF and filtered out for the NAVSHOCK Y-DOF computation.

## Excluding Individual Modes

You can use Direct Matrix Input (DMI) to select which modes to include in the NRL summation. To do so, define a partitioning matrix named PARTNVEC by adding DMI bulk entries to the Simcenter Nastran input file. If you are using the interactive procedure, the DMI bulk entries must be present in the bulk section of the Simcenter Nastran input file for the restart run. The partitioning matrix breaks the eigenvector matrix into specified submatrices. The partition matrix is a 3-column matrix where the columns correspond to the shock directions and the rows correspond to the mode numbers.

The example demonstrates how to create a partitioning matrix that selects certain modes for inclusion in the NRL sums from a system that has 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 column 1 (keep all modes – 1 -12) (fore/aft shock) | | | | | | | | |
| DMI | PARTNVEC | 1 | 1 | 1. | THRU | 12 | | |
| $ Define column 2 (keep modes 1, 2 and 3) (athwartship shock) | | | | | | | | |
| DMI | PARTNVEC | 2 | 1 | 1. | 2 | 1. | 3 | 1. |
| $ Define column 3 (keep only mode 5) (vertical shock) | | | | | | | | |
| DMI | PARTNVEC | 3 | 5 | 1. | | | | |

Note:

If you use DMI to choose the modes to include in the NRL sum, use 100.0 as the weight cutoff percentage.

## OUTPUT4 File Contents

During Phase I of the DDAM procedure, Simcenter Nastran performs a modal analysis. Natural frequencies, modal participation factors, and modal effective masses are calculated for each mode and are written to an ASCII OUTPUT4 file. The contents of the OUTPUT4 file include the following matrices:

| | |
|---|---|
| OMEX | Natural frequencies |
| PAB | Modal participation factors |
| MTOT | Total mass in each direction |
| MFRACT | Percent of total mass from the modes used |

Simcenter Nastran prints the following data in addition to the four matrices written to the OUTPUT4 file for user verification:

| | |
|---|---|
| MTOTC | Total rigid body mass matrix |
| MEFFC | Total modal effective mass matrix |
| MEFF | Diagonal terms of MEFFC |
| PHBASE | Rigid body vector set referenced to R-set |

# 6. Data Recovery

## 6.1 Overview of Data Recovery

In dynamic analysis, the size of the calculated output data could become several orders of magnitude larger than the size of output for the equivalent statics analysis job. For instance, a typical transient run could produce thousands of solution vectors. To prevent this, you can use the SET logic in the Case Control Section of the input file to reduce the number of outputs.

The choice of modal methods versus the direct method for dynamic solutions is discussed in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*. Each method uses a different technique to recover output answers. This chapter extends this discussion by describing several options available in the modal method to improve accuracy while reducing computation costs.

This chapter describes two methods for calculating physical results from a solution using modal degrees-of-freedom. The default method is the "matrix" method, which is the most effective technique for the majority of dynamics problems. The alternative method uses the static approach to calculate results one vector at a time.

The mode acceleration method, also detailed in this chapter, is useful in some cases where accurate stress results are required for only a few peak time steps or frequencies. However, this option may be more expensive if many solution vectors exist.

In addition to the standard displacement, stress, and force outputs, dynamics engineers frequently use other types of output that combine the effects of random or spectral inputs. Shock and response spectral analysis provides a quick calculation of the probable peak values in a modal solution for base accelerations. Random analysis is a more formal method of determining power spectral densities, RMS averages, and other data for a structure in a random load environment. This chapter devotes a section to each of these topics.

## 6.2 Data Recovery for Modal Methods

Modal methods of analysis have the advantages of reducing the problem size, decoupling the equations, and providing an intuitive feeling for the results. However, they require additional calculation steps and may also result in a loss in accuracy. There are four different methods for calculating the detailed results in the data recovery stage of analysis using modal coordinates. Each of these options provide a unique trade-off of efficiency vs. accuracy for a given size of problem, number of solutions, and number of output requests. Described below are the normal mode displacement method, the matrix method, the mode acceleration method, and the use of superelements for improved modal results.

### Mode Displacement, Vector Method

This method is almost identical to the statics method of data recovery in Simcenter Nastran. It is efficient when only a few time steps or frequencies are present in the solution and many outputs are requested (such as for contour plotting). Starting with the modal displacements, $[\xi]$, and the

eigenvectors, [φ], we obtain a set of displacement vectors that are expanded as with statics or normal mode analysis. The steps are summarized below.

Analysis Set

$$\{u_a\} = [\phi_a]\{\xi\}$$

**Equation 6-1.**

Free Set

$$\{u_f\} = \begin{bmatrix} G_0 \\ I \end{bmatrix}\{u_a\}$$

**Equation 6-2.**

SPC Forces

$$\{Q_s\} = [K_{sf}]\{u_f\} - \{P_s\}$$

**Equation 6-3.**

Independents

$$\{u_n\} = \begin{Bmatrix} u_f \\ 0 \end{Bmatrix}$$

**Equation 6-4.**

Grid Set

$$\{u_g\} = \begin{bmatrix} G_m \\ I \end{bmatrix}\{u_n\}$$

**Equation 6-5.**

In the actual calculations, each vector at a particular frequency or time step becomes a column of the solution matrix $[u_g]$. Note that for all modal methods, no provisions are made for thermal loads or enforced displacements (which are only calculated in statics analysis.)

The element stresses and forces are calculated directly from the displacement vectors, one element and one vector at a time. Note that if a large number of solutions exist (for instance 10000 time steps), the size of the matrices defined above may be quite large and the calculation speed will be very slow.

## Matrix Method for Modal Response

This method is very efficient when the number of solutions is much larger than the number of modes and when a limited number of output requests is made. The matrix method for modal response is the default for the modern modal solutions. In this method, the same operations as described above are performed on the eigenvectors and the results are saved. A matrix is calculated for every output request

using the eigenvector matrix instead of a solution vector. The intermediate matrices are calculated from the output data recovery of the modal solution. These are summarized below.

Displacements $$[u_g^m] = [\phi_g]$$

**Equation 6-6.**

SPC Forces $$[Q_s^m] = [K_{sf}][\phi_f]$$

**Equation 6-7.**

Stresses $$[S^m] = [\sigma(\phi)]$$

**Equation 6-8.**

Forces $$[F^m] = [f(\phi)]$$

**Equation 6-9.**

Note that the number of rows of these matrices will correspond only to the requested output sets (DISP =, STRESS =, etc.). The number of columns is equal to the number of modes (which is usually smaller than the number of time steps of frequencies). For the final output printing or xy-plotting, the equations for calculating the transient solution outputs are as follows:

Displacements $$[u_g^o(t)] = [u_g^m][\xi(t)]$$

**Equation 6-10.**

Forces of SPC $$[Q_s(t)] = [Q_s^m][\xi(t)]$$

**Equation 6-11.**

Similar equations are used for SPC forces, stresses, and element forces, as well as for frequency response, by simply combining the modal responses with the modal factors as indicated above.

## 6.3 Sparse Data Recovery

Sparse data recovery is a performance capability that you can use in conjunction with other modal data recovery methods. Sparse data recovery reduces the number of matrix-multiplications necessary to recover large amounts of data from a modal analysis. Sparse data recovery can significantly reduce the time required to recover large amounts of data.

Sparse data recovery is supported and used by default for the following solutions:

- Direct frequency response (SOL 108)
- Modal frequency response (SOL 111)(1)
- Modal transient response (SOL 112)(1)
- Optimization (SOL 200, ANALYSIS = MFREQ or MTRAN)(1)

(1) Sparse data recovery is only supported for the constraint mode method of enforced motion. If the loading is not enforced motion, or if the absolute method is used, sparse data recovery is not invoked.

If you plan to run a mini restart (scr=mini) in order to recover additional results later from a constraint mode enforced motion analysis, you must deactivate sparse data recovery. Otherwise, only results requested in the original run will be output; any additional result requests in the mini restart run will be zero. As a best practice, because the sparse data recovery method is so much more efficient, avoid using the mini restart method. Either request all the necessary output in the initial run or use a regular restart.

In the event you choose to deactivate sparse data recovery, you can do so by:

- Setting system cell 421 to zero.

```
NASTRAN SYSTEM(421) = 0
```

- Using the SPARSEDR keyword.

```
NASTRAN SPARSEDR = 0
```

Sparse data recovery supports various output requests including SPCFORCE, MPCFORCE, GPFORCE, GPSTRAIN, GPSTRESS, among others. However, the extensive amount of data required to generate MPRES results requires that sparse data recovery not be performed.

## 6.4 Improving the Results

### Mode Acceleration Method

The mode acceleration method of data recovery is used to reduce the modal truncation errors that occur in the methods above. For good accuracy, a general recommendation is to use many more modes than in the frequency range of the excitation loads. For most applications this is adequate and the mode displacement method produces acceptable results. However, some cases that may not produce accurate results are the following:

1. Errors may occur in the deformations and stresses in the area of a concentrated load on a free point.

2. Concentrated damper elements or direct input matrices may change the characteristic response displacements to be much different from simple modal combinations.

3. Models with complex shapes may require an excessive number of modes to represent the stress distributions. Accuracy of the modes becomes an issue when more than a few hundred are necessary.

4. Massless points with external loads will have deformations and internal loads that cannot be represented by finite frequency modes.

Using the direct method may eliminate these errors but may excessively increase the run costs and file storage. The mode acceleration method is a useful compromise that eliminates the errors described above yet is cheaper to run than the direct method. However, it will cost more than the matrix method if a large number of solution vectors are to be processed.

The mode acceleration method derivation is given below. The derivation of the mode acceleration method specific to the constraint mode method of enforced motion is also included at the end of the section. SeeOverview of Enforced Motion in the *Simcenter Nastran Basic Dynamic Analysis User's Guide* for information on enforced motion methods.

**Mode Acceleration Method Derivation**

For convenience, we will show the equations for a frequency response solution. However, the results are similar with a transient analysis. The ideal solution desired is that for the direct matrix solution at frequency $\omega$, which is

$$[-\omega^2 M + j\omega B + K]\{u(\omega)\} = \{P(\omega)\}$$

**Equation 6-12.**

where *j* is the imaginary unit value.

We may obtain an exact solution if we use all of the modes in the system; namely, if N is the order of the problem, then:

$$\{u\} = \sum_{i=1}^{N} \{\phi_i\}\xi_i$$

**Equation 6-13.**

Note that this set of modes also includes modes with infinite frequencies. These are present with singular mass matrices. The modal identities are

$$\omega_i^2[M]\{\phi_i\} = [K]\{\phi_i\}$$

**Equation 6-14.**

and for constant structural damping parameters, $g$ and $\omega_3$

$$[B]\{\phi_i\} = \left(\frac{g}{\omega_3}\right)[K]\{\phi_i\}$$

**Equation 6-15.**

### Eigenvalue of Modal and Direct Solutions

Substituting **Eq. 6-13**, **Eq. 6-14**, and **Eq. 6-15** into the terms defined in **Eq. 6-12**, we obtain the effect of each mode on the exact solution. These terms are

$$-\omega^2[M]\{u\} = \sum_{i=1}^{N}\left(\frac{\omega}{\omega_i}\right)^2[K]\{\phi_i\}\xi_i$$

**Equation 6-16.**

$$j\omega[B]\{u\} = \sum_{i=1}^{N}\left(\frac{jg\omega}{\omega_3}\right)[K]\{\phi_i\}\xi_i$$

**Equation 6-17.**

and

$$[K]\{u\} = \sum_{i=1}^{N}[K]\{\phi_i\}\xi_i$$

**Equation 6-18.**

If all modes were included, we know that the results would be identical to a direct solution. For the high frequency modes ($\omega_i > \omega_m$), it is apparent that their contribution will be much smaller for mass and damping effects than from their effect on the static solution of **Eq. 6-18**. Therefore, if we truncate the

modes such that the primary error will be due to the forces generated by the missing modes in Eq. 6-18. In the frequency domain the approximations to Eq. 6-16 through Eq. 6-18 become

$$-\omega^2[M]\{u\} \approx -\sum_{i=1}^{M} \omega^2[M]\{\phi_i\}\xi_i$$

**Equation 6-19.**

$$j\omega[B]\{u\} \approx \sum_{i=1}^{M} j\omega[B]\{\phi_i\}\xi_i$$

**Equation 6-20.**

$$[K]\{u\} = \sum_{i=1}^{M} [K]\{\phi_i\}\xi_i + [K]\{\Delta u\}$$

**Equation 6-21.**

The error can be expressed as a residual displacement vector $\{\Delta u(\omega)\}$.

Substituting Eq. 6-19 and Eq. 6-20 into Eq. 6-15, we obtain a corrected solution:

$$[K]\{u(\omega)\} = \{P(\omega)\} - \{R(\omega)\}$$

**Equation 6-22.**

where:

$$\{R(\omega)\} = j\omega \sum_{i=1}^{M} \{\phi_i\}\xi_i - \omega^2 \sum_{i=1}^{M} \{\phi_i\}\xi_i$$

An alternate method would be to solve the static part of Eq. 6-15 using Eq. 6-21. Since the residual vector is orthogonal to the retained modes, then:

$$[\phi]^T[K]\{\Delta u\} = \{0\}$$

**Equation 6-23.**

We can assume that {Δ*u* contains static response, if [*K*] is nonsingular, plus modal response in the form:

$$\{\Delta u\} = -[K]^{-1}\{P\} + [\phi]\{x\}$$

**Equation 6-24.**

Premultiplying **Eq. 6-24** by [$Q^T$*K* and using **Eq. 6-23**, we obtain

$$x_i = -(m_i\omega_i^2)^{-1}[\phi_i]^T\{P\}$$

**Equation 6-25.**

and, then from **Eq. 6-24** and **Eq. 6-25**, we obtain a decoupled residual solution

$$\{\Delta u(\omega)\} = [\Delta Z]\{P(\omega)\}$$

**Equation 6-26.**

where:

$$[\Delta Z] = [K]^{-1} - [\phi]\begin{bmatrix} \cdot & & \\ & (m_i\omega_i^2) & \\ & & \cdot \end{bmatrix}[\phi]^T$$

The matrix, [Δ*Z*], is known as the residual flexibility matrix and has also been used for modal synthesis modeling. Here it could be used as a data recovery step.

In the Simcenter Nastran design, **Eq. 6-22** has been chosen over **Eq. 6-26** for the calculation of the improved solution of large problems because the [Δ*Z*] matrix could be very dense and the singular free body case was easier to process.

Note that in the actual mode acceleration process other nonstructural effects, such as direct input matrices or transfer functions, are not included in the matrix [*K*], but are treated as terms on the right-hand side and added to {*P*}. For free bodies, the right hand loads are converted to equilibrium loads to permit a decomposition of the singular stiffness matrix, identically to the inertia relief solution. In fact,

the entire process may be viewed as if all the dynamic modal solutions were converted into equivalent static loads, and linear static solutions were generated using the symmetric structural stiffness matrix.

**Mode Acceleration Method with the Constraint Mode Method of Enforced Motion**

The dynamic equation of motion with the constraint mode method is

$$M_{ff}\ddot{v}_f + B_{ff}\dot{v}_f + K_{ff}v_f + i\tilde{K}_{ff}v_f = P_f - (M_{fs} + M_{ff}G_{fs})\ddot{u}_s - (B_{fs} + B_{ff}G_{fs})\dot{u}_s - i(\tilde{K}_{fs} + \tilde{K}_{ff}G_{fs})u_s$$

**Equation 6-27.**

where the total response, $u_f$, is computed as the sum of a static enforced motion component, $-K_{ff}^{-1}K_{fs}u_s$, and a dynamic relative displacement component, $v_f$, as

$$u_f = v_f + G_{fs}u_s \qquad where \qquad G_{fs} = -K_{ff}^{-1}K_{fs}$$

**Equation 6-28.**

The solution to **Eq. 6-27** can be obtained efficiently in modal coordinates, $\gamma$, by using the modal transformation

$$v_f = \Phi\gamma$$

**Equation 6-29.**

Here $\Phi$ is the mode shape matrix of normal modes.

Substituting into equation **Eq. 6-27** gives

$$m\ddot{\gamma} + b\dot{\gamma} + k\gamma + i\tilde{k}\gamma = \Phi^T\left[P_f - (M_{fs} + M_{ff}G_{fs})\ddot{u}_s - (B_{fs} + B_{ff}G_{fs})\dot{u}_s - i(\tilde{K}_{fs} + \tilde{K}_{ff}G_{fs})u_s\right]$$

**Equation 6-30.**

where

$$m = \Phi^T M_{ff}\Phi$$
$$k = \Phi^T K_{ff}\Phi$$
$$\tilde{k} = \Phi^T \tilde{K}_{ff}\Phi$$
$$b = \Phi^T B_{ff}\Phi$$

**Equation 6-31.**

The matrices *m* and *k* are modal diagonal matrices. The modal viscous damping matrix, *b*, and modal structural damping matrix, $\tilde{k}$, are generally not diagonal.

Standard modal analysis involves solving **Eq. 6-30** for $\gamma$, and substituting back into equations **6-28** and **6-29** to get the physical displacement $u_f$.

In the mode acceleration method, the same approach is used again to compute the modal displacement. But a more accurate physical displacement response is computed by solving for the displacement $v_f$ from a direct solution of **Eq. 6-27** with the inertia and damping terms taken to the right hand side and computed in modal terms. This gives the response computed as

$$v_f = K_{ff}^{-1}P_f - K_{ff}^{-1}(M_{fs} + M_{ff}G_{fs})\ddot{u}_s - K_{ff}^{-1}(B_{fs} + B_{ff}G_{fs})\dot{u}_s - iK_{ff}^{-1}(\tilde{K}_{fs} + \tilde{K}_{ff}G_{fs})u_s$$
$$- K_{ff}^{-1}M_{ff}\Phi\ddot{\gamma}_f - K_{ff}^{-1}B_{ff}\Phi\dot{\gamma}_f - iK_{ff}^{-1}\tilde{K}_{ff}\Phi\gamma_f$$

**Equation 6-32.**

Transforming back into total displacement coordinates, $u_f$, and using properties of the modal matrices, the absolute displacements can be computed as

$$u_f = K_{ff}^{-1}P_f - K_{ff}^{-1}(M_{fs} + M_{ff}G_{fs})\ddot{u}_s - K_{ff}^{-1}(B_{fs} + B_{ff}G_{fs})\dot{u}_s - iK_{ff}^{-1}(\tilde{K}_{fs} + \tilde{K}_{ff}G_{fs})\dot{u}_s - G_{fs}u_s$$
$$- \Phi\Omega^{-2}(\ddot{\gamma} + m^{-1}b\dot{\gamma} + im^{-1}\tilde{k}\gamma)$$

**Equation 6-33.**

where $\Omega$ is a square modal size matrix with the modal frequencies on the diagonal. The terms on the right hand side of **Eq. 6-33** are essentially different types of mode shapes. The names for the various shapes are

| | |
|---|---|
| $\Phi$ | = real normal modes |
| $G_{fs}$ | = static constraint modes |
| $K_{ff}^{-1}P_f$ | = applied force attachment modes |
| $-K_{ff}^{-1}(M_{fs} + M_{ff}G_{fs})$ | = inertia attachment modes |
| $-K_{ff}^{-1}(B_{fs} + B_{ff}G_{fs})$ | = viscous damping attachment modes |
| $-K_{ff}^{-1}(\tilde{K}_{fs} + \tilde{K}_{ff}G_{fs})$ | = structural damping attachment modes |

## Using Superelements for Data Recovery

If a structure has only a few points with dynamic loads or enforced boundary motion, a simple superelement (SE) operation will have an improvement in accuracy similar to the mode acceleration method. In transient analysis, the cost will be increased because the uncoupled modal solution step is changed to coupled solution with a few extra degrees-of-freedom. In frequency response analysis, the extra cost is very small. The basic procedure is as follows:

1. Subdivide the structure into two superelements corresponding to the external loads.

2. The residual (SE = 0) contains only the loaded structural points and scalar points for the modal coordinates.

3. The upstream (SE = N) contains all of the remaining points and all of the elements. It will also contain the residual SE points as boundary points.

4. Place a modal synthesis request for the upstream SE with a METHOD = request in the corresponding Case Control subcase.

5. By default, the mode shapes will be fixed at the residual points. Free boundary degrees-of-freedom may be specified on SECSET data.

6. The residual SE solution will contain both modal and grid point degrees-of-freedom. If the resulting size is small, a direct solution for the forced response is recommended.

7. The upstream solution will contain both modal response and static residuals (described above) due to motion of the residual grid points. Stresses and forces will be improved, especially on the interface elements.

In this method the static correction terms are assumed to be shapes corresponding to loads on the boundary points. The upstream displacements, $\{u_o\}$, are calculated from the equation

$$\{u_o\} = [G_{oa}]\{u_a\} + [[\phi_o] - [G_{oa}][\phi_a]]\{q\}$$

**Equation 6-34.**

where:

| | | |
|---|---|---|
| $\{u_a\}$ | = | the residual SE displacements |
| $[G_{oa}]$ | = | the Guyan reduction matrix |
| $[\varphi_o]$ | = | the $u_o$ partition of the eigenvectors |
| $[\varphi_a]$ | = | the $u_a$ partition of the eigenvectors |
| $\{q\}$ | = | the modal degrees-of-freedom |

The residual SE will obtain a dynamic solution for both $u_a$ and $\{q\}$.

## 6.5 Shock and Response Spectrum Analysis

Shock spectra analysis and response spectrum analysis are methods used by many engineers to estimate the maximum dynamic response of a structure. Most applications involve complicated time-dependent loads or accelerations that excite the base of a structure, such as an earthquake ground motion on a building or an explosive shock on a small component in a ship. (Note that the only difference between shock and response spectra is whether output displacements are measured in a fixed frame of reference or relative to the base motion.)

The advantage of these methods over a conventional transient analysis is economy and simplicity. The only major calculation step is obtaining a sufficient number of normal modes to represent the entire frequency range of the input excitation and resulting response. The disadvantage of the method is that the accuracy may be questionable and the requirement of special input data in the Simcenter Nastran solution sequences. In many cases, a direct transient analysis with the actual excitation load will be more accurate, easier to use, and faster.

The procedure involves two stages:

- First the applied loads or base excitations are converted in a direct transient solution (SOL 109) into a spectrum table consisting of peak response magnitudes for a set of single degree-of-freedom oscillators. Each oscillator is a scalar spring/mass/damper having a different natural frequency and damping ratio. This stage is optional since the shock spectrum data is frequently given in the contractual design specifications or, in the case of earthquakes, is available through governmental agencies.

- The second stage of the analysis consists of a modal analysis of the structure, data recovery, and the response calculation that combines the modal properties of the analysis model with the spectrum data of the applied loads. This stage is performed in a modal analysis solution sequence (SOL 103). If a database was saved from the first stage, a restart will provide the spectrum data automatically. Otherwise, the spectrum data must be supplied in a direct tabular input (response versus natural frequency for several damping ratios).

### Theoretical Background of Shock and Response Spectrum Analysis

Starting with a modal transient analysis, the general approximation for a response quantity at a degree-of-freedom *k* is

$$u_k(t) = \sum_i \phi_{ik} \xi_i(t)$$

**Equation 6-35.**

where $\varphi$ and $\xi$ are the modal outputs and generalized displacements over *i* modes. The actual modal equations are

$$\ddot{\xi}_i + g_i\omega_i\dot{\xi}_i + \omega_i^2\xi_i = \lfloor \phi_i \rfloor^T \{P(t)\}$$

**Equation 6-36.**

where $P$ is the vector of loading functions. See "Modal Transient Response Analysis" in the *Basic Dynamic Analysis User's Guide* for more information.

For loading due to base accelerations, the equivalent inertial loads are

$$\{P(t)\} = -[M_{aa}][D_{ar}]\{\ddot{u}_r(t)\}$$

**Equation 6-37.**

where the columns of $[D_{ar}$ represent vectors of rigid body motions of the whole structure and the accelerations correspond to the base motions. Substituting **Eq. 6-37** into **Eq. 6-36** and combining terms we can separate the modal quantities from the transient solutions. First we will develop the transient response functions. We begin by calculating the responses

$$\ddot{x}_r + g\omega\dot{x}_r + \omega^2 x_r = \ddot{u}_r(t)$$

**Equation 6-38.**

where $x_r$ is a response function in direction $r$, and is a function of the variables $\omega$, $g$, and $t$. The peak values of $x_r$, obtained over a range of frequencies and damping factors is called the response spectrum for the excitation, $\ddot{u}_r$.

Next, from the normal mode analysis, we define the participation factors $\psi_{ir}$ , for mode $i$ and direction $r$, as

$$\psi_{ir} = -\lfloor \phi_i \rfloor^T [M_{aa}]\{D_{ar}\}$$

**Equation 6-39.**

Then, from **Eq. 6-35**, the actual transient response at a physical point is

$$u_k(t) = \sum_i \sum_r \phi_{ik}\psi_{ir}x_r(\omega_i, g_i, t)$$

**Equation 6-40.**

The peak magnitudes of $u_k$ in **Eq. 6-40** are usually dominated by the peak values of $x(t)$ occurring at the natural frequencies. In spectrum analysis the peak values of $u_k$ are approximated by combining functions of the peak values, $\bar{x}_{ri}(\omega_i, g_i) = \max|x_{ri}(\omega_i, g_i, t)|$, in the approximation

$$\bar{u}_k(t) \cong \sum_i \sum_r |\phi_{ik}| \, |\psi_{ir} \bar{x}_{ri}(\omega_i, g_i)|$$

**Equation 6-41.**

### ABS Option

**Eq. 6-40** and **Eq. 6-41** define the ABS (Absolute Value) option. This method assumes the worst case scenario in which all of the modal peak values for every point on the structure are assumed to occur at the same time and in the same phase. Clearly in the case of a sudden impact, this is not very probable because only a few cycles of each mode will occur. However, in the case of a long term vibration, such as an earthquake when the peaks occur many times and the phase differences are arbitrary, this method is acceptable.

A second way of viewing the problem is to assume that the modal magnitudes and phases will combine in a probabilistic fashion. If the input loads are behaving randomly, the probable (RMS) peak values are

$$\bar{u}_k \cong \sqrt{\sum_i (\phi_{ik} \bar{\xi}_i)^2}$$

**Equation 6-42.**

where the average peak modal magnitude, $\bar{\xi}_i$ is

$$\bar{\xi}_i = \sqrt{\sum_r (\psi_{ir} \bar{x}_r(\omega_i, g_{i_i}))^2}$$

**Equation 6-43.**

### SRSS Method

This approach is known as the SRSS (square root of sum-squared) method. Note that the results in each direction are summed in vector fashion for each mode first, followed by an SRSS calculation for all modes at each selected output quantity $u_k$. It is assumed that the modal responses are uncorrelated and the peak value for each mode will occur at a different time. These results are optimistic and represent a lower bound on the dynamic peak values.

The SRSS method may underestimate the actual peaks since the result is actually a probable peak value for the period of time used in the spectrum analysis. The method is normally augmented with a safety factor of 1.5 to 2.0 on the critical outputs.

**NRL Method**

As a compromise between the two methods above, the NRL (Naval Research Laboratories) method was developed. Here, the peak response is calculated from the equation

$$\bar{u}_k \cong \left|\phi_{jk}\bar{\xi}_j\right| + \sqrt{\sum_{i \neq j} \left(\phi_{ik}\bar{\xi}_i\right)^2}$$

**Equation 6-44.**

where the j-th mode is the mode that produces the largest magnitude in the product $\phi_{jk}\bar{\xi}_j$. The peak modal magnitudes, $\left|\phi_{jk}\bar{\xi}_j\right|$, are calculated with **Eq. 6-43**.

The rationale for the method is that the peak response will be dominated by one mode and the SRSS average for the remaining modes could be added directly. The results will fall somewhere between the ABS and SRSS methods.

Modes that are close in frequency may have their peak response occur at about the same time (and with the same phase). For this reason, the SRSS and NRL methods contain a provision to sum modal responses via the ABS method for modes that have closely spaced natural frequencies. Close natural frequencies are defined by frequencies that meet the following inequality:

$$f_{i+1} < \text{CLOSE} \cdot f_i$$

The value for CLOSE is set by PARAM,CLOSE (the default is 1.0).

The modal summation option is set via PARAM,OPTION (the ABS method is the default). Both PARAM,OPTION and PARAM,CLOSE may be set in any subcase, allowing summation by several conventions in a single run.

## Generating Response Spectra Curves

A response spectrum is generated as follows:

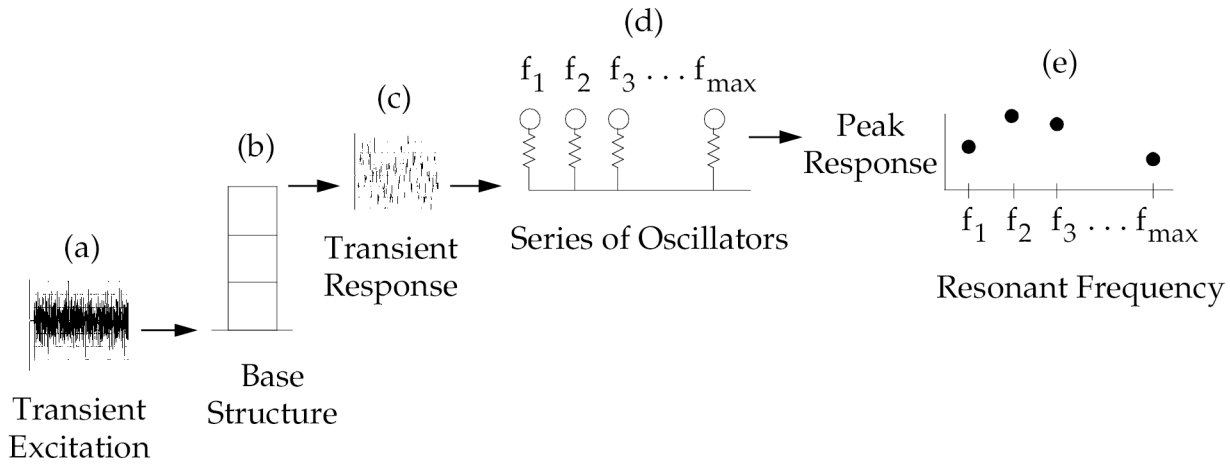
1. Assume that there is a series of small, single degree-of-freedom (SDOF) oscillators each attached to the same location at the connection to the base structure. (In the examples in the introduction to this section, the base structures are the building and the spacecraft). These oscillators each have a different resonant frequency, and all have the same modal damping ratio (2%, for example).

2. Apply a transient excitation to the base structure. Use the base structure's damping when computing the base structure's transient response at the location of the SDOF oscillators.

3. Use the base structure transient response as input to each SDOF oscillator. Compute the magnitude of peak response of each SDOF oscillator, and plot the peak response versus oscillator resonant frequency.

4. Change the modal damping ratio for the oscillators (to 5%, for example) and repeat Steps 2 and 3 for the range of expected damping.

The response spectrum, therefore, depicts the maximum response magnitude of an SDOF system as a function of resonant frequency and damping. **Figure 6-1** depicts the generation of a spectrum.




Transient excitation (a) is applied to a base structure (b), from which transient response (c) is computed for each floor. This response is applied to a series of oscillators (d), for which the peak response is plotted (e). Steps (d) and (e) are repeated for different damping values to form response spectra as shown below.

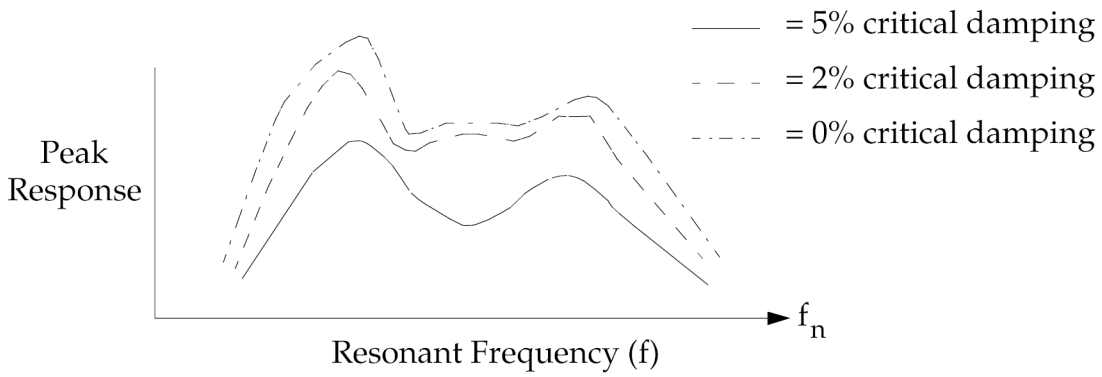



**Figure 6-1. Response Spectrum Generation**

Note that the peak response for one oscillator does not necessarily occur at the same time as the peak response for another oscillator. Note, too, that there is no phase information since only the magnitude of peak response is computed.

Several values of peak response are computed for the oscillators. These include the following response variable.

Relative velocity and absolute acceleration are approximately related to the relative displacement by

$$\dot{X}_r \approx \omega X_r$$
$$\ddot{X} \approx \omega^2 X_r$$

**Equation 6-45.**

Design spectra are most often defined in terms of $X_r$, $\dot{X}_r$, and $\ddot{X}_r$.

Note that for very low oscillator frequencies ($\omega \rightarrow 0$),

$$X \rightarrow 0$$
$$X_r \rightarrow U_b$$

**Equation 6-46.**

where $U_b$ is the motion of the base of the oscillator.

Similarly, for very high oscillator frequencies ($\omega \rightarrow \infty$),

$$X \rightarrow U_b$$
$$X_r \rightarrow 0$$

**Equation 6-47.**

The approximate relationships between $X_r$, $\dot{X}_r$, and $\ddot{X}_r$, shown in **Eq. 6-45**, are not valid at very low or very high oscillator frequencies or for large values of damping.

It is assumed in this process that the mass of each oscillator is very small relative to the base structure, so the oscillator's response does not influence the response of the base structure.

**Dynamic Response Predictions**

Once a spectrum is computed, it can be used for the dynamic response analysis of a Simcenter Nastran model of the component. For example, the spectrum generated for a floor in a building that is subjected to an earthquake can then be applied to a complex model of a piece of equipment attached to that floor. The peak response of each mode of the equipment model is obtained from the spectrum, and these peak responses are combined to create the overall response.

Because response spectrum generation involves transient response, similar modeling and analysis considerations apply. In addition, the time step (field 4 on the TSTEP Bulk Data entry, DT) should not be changed during the run, because Simcenter Nastran uses only the initial DT specification for the entire response spectrum generation run and therefore wrong answers could occur.

The time step, DT, and time duration, DT · N (where N is the number of time increments), must take into account the loading, the base structure, and the frequency range of the spectra generation. The time step must take into account the frequency content of the applied excitation, the frequencies of the base structure, and the highest frequency for which spectra are to be generated. There must be enough time steps per cycle of response for both the base structure and the highest frequency oscillator in order to accurately predict the peak response; 5 to 10 steps per cycle represents a typical value. In addition, the time duration of the loading, the frequencies of the base structure, and the lowest oscillator frequency must be considered when defining the time duration. There must be a long enough time duration of response both for the base structure and the lowest frequency oscillator in order to accurately predict the peak response. For short duration loadings, the peak response often occurs well after the load has peaked.

Initial conditions (specified via the TIC Bulk Data entry) are not used in response spectrum generation. Initial conditions are used in the calculation of the transient response of the base structure, but the calculation of the peak oscillator responses (i.e., the response spectrum calculation) ignores any initial conditions.

## User Interface for Response Spectra Generation

Response spectra are generated in the transient response solution sequences (SOL 109 for direct and SOL 112 for modal). Transient response input is required to apply the transient excitation to the base structure. Additional input required to generate response spectra are described in **Table 6-2** and **Table 6-3**.

**Table 6-2. Case Control Input for Response Spectrum Generation**

| Case Control Command | Description |
|---|---|
| XYPLOT SPECTRAL | Compute spectra. |
| XYPUNCH SPECTRAL | Punch spectra for subsequent use. |

The XYPLOT and XYPUNCH commands are included in the OUTPUT(XYPLOT) section. Further details about the OUTPUT(XYPLOT) section are described in "**Structure Plotting** " in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*.

Examples of these commands include:

```
$  Plot absolute acceleration spectra for grid point 85, T3 component
XYPLOT     ACCE     SPECTRAL     1     /85(T3RM)
$  Punch relative displacement spectra for grid point 3, T1 component
XYPUNCH    DISP     SPECTRAL     1     /3(T1IP)
```

Relative and absolute spectra are denoted by IP and RM, respectively, in the parentheses of the curve request.

**Table 6-3. Bulk Data Input for Response Spectrum Generation**

| Bulk Data Entry | Description |
|---|---|
| PARAM,RSPECTRA,0 | Requests calculation of spectra. |
| DTI, SPSEL, 0 | Header for DTI. |
| DTI, SPSEL, 1 | Selects oscillator frequencies, oscillator damping values, and grid points at which spectra will be computed. |
| FREQi | Specifies oscillator damping values. |
| FREQi | Specifies oscillator frequencies. |

There are two FREQi entries: one to specify oscillator frequencies (i.e., frequencies for which spectra will be computed) and the other to specify oscillator damping. (Note that damping for the base structure is specified in another manner, such as with the TABDMP1 entry used for modal transient response analysis.)

## Peak Response Calculation

Response spectra are applied to a structure for subsequent analysis. Note that the structure in this case is not the base structure for which the spectra were computed but, rather, a smaller structure such as a piece of equipment.

Response spectrum analysis approximates the peak structural response (typically stresses and displacements, though not limited to those quantities). Approximations are made by assuming:

- Only the peak response is computed for the oscillators (see the preceding section).
- There is no phase information or sign computed; only the magnitude of peak response is computed.
- The oscillator mass is small relative to the base structure's mass.
- The displacements, velocities, and accelerations are related by the approximate relationships in **Eq. 6-48**.

$$\dot{X}_r \approx \omega X_r$$
$$\ddot{X} \approx \omega^2 X_r$$

**Equation 6-48.**

For more information, see **Generating Response Spectra Curves**.

- The peak modal responses are combined to form the overall response via various combination methods (see below).

The spectra themselves are often approximate. For example, design spectra have been developed for seismic analysis, and these have conservatisms built into them by the fact that safety factors are incorporated by either increasing the spectra values and/or by decreasing the damping values.

These approximations make response spectrum analysis a tool that is useful for design and efficiency. The effects of these approximations are further described in the remainder of this section.

## Modeling and Analysis Considerations

Response spectrum application is a postprocessing function of normal modes analysis. It is run in the normal modes solution sequences, so the modeling and analysis considerations that apply for normal modes analysis also apply for response spectrum application. The additional considerations also need to be followed:

1. The structure is run as an unrestrained model in the direction(s) of the load spectrum application.

2. A large mass, on the order of $10^3$ to $10^6$ times the mass of the structure, must be used at the structure's base grid points where the input occurs. One way to do this is to use RBEs or MPCs to connect the base points to a separate grid point, and apply the large mass to that separate grid point. This separate grid point is where the spectrum is applied.

3. A SUPORT Bulk Data entry is required at the spectrum input location. See **Free Body Techniques** for an explanation.

4. The modes must be mass normalized (which is the default).

The spectra that Simcenter Nastran can apply are absolute acceleration, relative displacement, and relative velocity spectra. You specify A, D, or V for acceleration displacement or velocity, respectively, to specify the spectrum type.

Use all modes within the frequency range specified by the spectrum, but do not use modes outside of the spectrum range. Usually, spectra to apply are considered to have zero values outside of their range of definition; for example, an absolute acceleration spectrum defined from 0 to 30 Hz is assumed to be zero beyond 30 Hz. However, Simcenter Nastran extrapolates spectral values for modes beyond the spectral range, which may lead to unexpected answers. You can limit the number of modes used in the spectrum application by limiting the number of computed modes (via the EIGRL or EIGR entry) or by

using PARAM,HFREQ,r (where r is the highest frequency mode to use) or PARAM,LMODES,n (where n is the number of lowest modes to use).

Consider the entire response spectrum process—generation and application—as a two-step process. Step 1 is generation of the response spectra and Step 2 is the application of the response spectra. For a given input, transient applied to the base structure (Step 1), the same stresses occur (Step 2) regardless of whether acceleration or displacement spectra were computed in Step 1. However, displacements and accelerations are different, because answers computed by using the absolute acceleration spectrum contain the rigid body contribution, whereas answers computed by using the relative displacement spectrum do not contain the rigid body contribution. Displacement and acceleration responses can be made equal regardless of which spectra was used by using PARAM,LFREQ,0.01 (or some other small number) to remove the rigid body mode contribution from the answers.

Stresses and other element quantities are unaffected by the contribution of any rigid body modes. The same situation applies to relative velocity spectra as to relative displacement spectra. However, because the relationships in **Eq. 6-49** are approximate, all answers (including stresses) will be slightly different depending on whether displacement, velocity, or acceleration spectra were used.

$$\dot{X}_r \approx \omega X_r$$
$$\ddot{X} \approx \omega^2 X_r$$

**Equation 6-49.**

See **Generating Response Spectra Curves**.

## User Interface for Spectrum Application

Response spectrum application is done in the normal modes solution sequences (SOL 103, for example). In addition to the input for computing normal modes, input is required for applying the spectra, as shown in **Table 6-4**.

**Table 6-4. Input for Response Spectrum Application**

| Case Control Command | Description |
|---|---|
| METHOD | Selects eigenvalue extraction method. |
| SDAMP | Selects the TABDMP1 Bulk Data entry. |
| DLOAD | Selects the DLOAD Bulk Data entry. |
| **Bulk Data Entry** | **Description** |
| PARAM,SCRSPEC,0 | Requests response spectrum application. |
| EIGR or EIGRL | Eigenvalue extraction method. |
| TABDMP1 | Specifies damping for the structure. |

| Case Control Command | Description |
|---|---|
| DLOAD | Defines spectrum multipliers. |
| DTI,SPECSEL,0 | Header for DTI. |
| DTI,SPECSEL,1 | Specifies type of spectrum (A, V, or D) and selects damping. |
| | A = absolute acceleration spectrum.<br>V = relative velocity spectrum.<br>D = relative displacement spectrum. |
| TABLED1 | Specifies input spectrum values. |
| SUPORT | Specifies input spectrum grid points. |
| CONM2,CMASS2, etc. | Defines large mass used for the input spectrum. |
| PARAM,OPTION,a | Specifies modal combination method (a = ABS [default], SRSS, or NRL). |
| PARAM,CLOSE,r | Specifies closeness parameter for modal combinations (the default is 1.0). |

All input listed in the table is required with the exception of PARAM,OPTION and PARAM,CLOSE.

## Response Spectrum Examples

This section provides an example of response spectra generation and an example of response spectrum application. The base structure for which response spectra are computed is a model of a five-story, shear-wall building, excited by an earthquake time history. The resulting spectra at a point on the fourth floor are applied to a bracket model in order to compute stresses, displacements, and accelerations.

### Response Spectrum Generation

Figure 6-2 depicts the five-story building model. The fifteen grid points at the base of the building are tied to a separate grid point (999) via RBE2 entries. Earthquake excitation is applied to this grid point, in the x direction (T1). The absolute acceleration response spectrum is computed for grid point 502. SOL 112, modal transient response, is used to compute the spectrum. Damping of 5% critical is used for the building and damping of 2% critical is used for the generated response spectrum. Listing 6-1 shows a portion of the input file for this model.

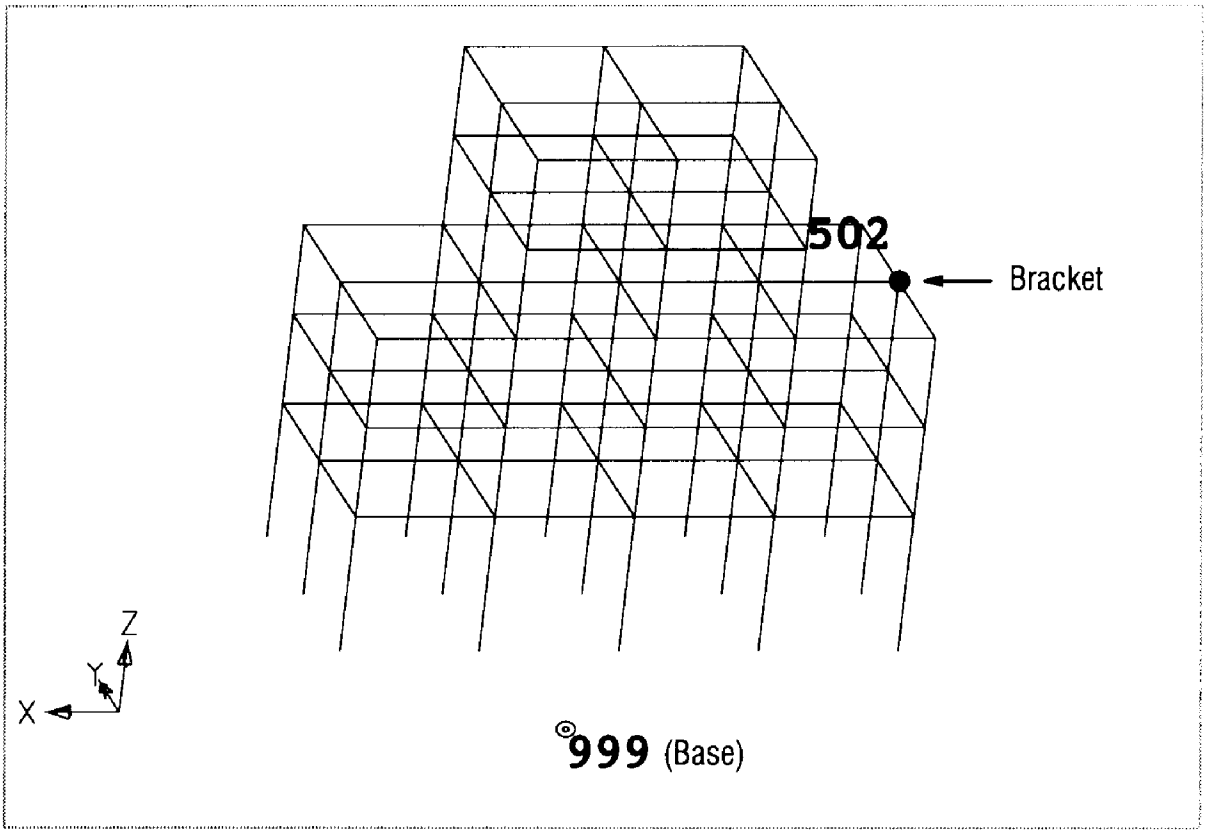



**Figure 6-2. Five-Story Building Model**

```
$  RESPONSE SPECTRUM GENERATION
$
$  Compute spectrum for absolute acceleration
$   for grid point 502 for 2% oscillator damping
$
ID NXN, DUG
TIME 15
SOL 112   $ MODAL TRANS. RESP.
CEND
$
TITLE = FIVE-STORY BUILDING, SHEAR MODEL
SUBTITLE = RESPONSE SPECTRUM GENERATION, X DIRECTION
$
SET 777 = 999, 101, 201, 301, 401, 502, 602
$
DISPLACEMENT(PLOT) = 777   $ AT LEAST DISP AND VELO MUST BE REQUESTED
VELOCITY(PLOT) = 777       $ FOR SPECTRUM GENERATION POINT
ACCELERATION(PLOT) = 777
$
METHOD = 999
```

```
DLOAD = 997
TSTEP = 995
SDAMPING = 991  $ DAMPING SPECIFICATION FOR THE BUILDING
$
OUTPUT(XYPLOT)
XGRID = YES
YGRID = YES
$
$  PLOT INPUT
XTITLE = TIME
YTITLE = ACCEL INPUT
XYPLOT ACCE /999(T1)
YTITLE = DISPL INPUT
XYPLOT DISP /999(T1)
$
$  RESPONSE SPECTRA GENERATION
XTITLE = FREQUENCY (HZ)
YTITLE = ABSOLUTE ACCELERATION
$
$  NOTE: IP -- IMAGINARY,PHASE -- IS RELATIVE
$  NOTE: RM -- REAL, MAGNITUDE -- IS ABSOLUTE
$
XYPLOT ACCE SPECTRAL 1 /502(T1RM)
XYPLOT ACCE SPECTRAL 1 /999(T1RM)
XYPUNCH ACCE SPECTRAL 1 /502(T1RM)
$
BEGIN BULK
$
$  REQUESTS CALCULATION OF SPECTRA
PARAM, RSPECTRA, 0
$
$  RESPONSE SPECTRUM GENERATION TABLE
$  DTI, SPSEL, RECNO, DAMPL, FREQL, Gi
$   RECNO = N, SELECTED BY XYPLOT
$   DAMPL = I, FREQi THAT SPECIFIES DAMPING VALUES
$   FREQL = FREQi THAT SPECIFIES FREQ. VALUES
$   Gi = K, L, ..., GRID POINTS FOR SPECTRA GENERATION
DTI, SPSEL, 0
DTI, SPSEL, 1, 885, 887, 101, 201, 301, 401,+DTI1
+DTI1, 502, 602
$
$  SPECIFY OSCILLATOR DAMPING VALUE (2%)
FREQ, 885, 0.02
$
$  SPECIFY FREQUENCIES FOR SPECTRA
FREQ1, 887, 0.01, 0.25, 121
$
$ ------------------------------------------------
$
```

```
$  ... rest of model ...
$
ENDDATA
```

**Listing 6-1. Abridged Input File for Response Spectrum Generation**

The XYPUNCH ACCE SPECTRAL command in the Case Control Section punches the response spectrum (which will be used later, in the response spectrum application). In the Bulk Data Section, PARAM,RSPECTRA,0 requests calculation of the spectrum. The DTI,SPSEL entry references the FREQ entries; one of the FREQ entries defines the oscillator damping and the other FREQ entry defines the frequency range over which to compute the spectra. The DTI,SPSEL entry also defines grid points at which to compute spectra; spectra are computed for grid points 101, 201, 301, 401, 502, and 602—one per floor—though only the one for grid point 502 is punched for subsequent use.

Note that a 2% damped spectrum was computed because that is the spectrum that will be applied to a component model (see below). In practice, however, spectra are often generated for multiple damping values (for example, 0%, 2%, and 5% damping).

The plotted absolute acceleration response spectra for grid points 999 and 502 are shown in **Figure 6-3**. Note that the response spectrum peaks in the region of about 1.5-7 Hz for grid point 999 and at about 2-3 Hz and again at about 7 Hz for grid point 502. The building has acted as both a filter (note the difference in frequency content) and magnifier (note the difference in spectral amplitudes). Floor response spectra computed closer to the base of the building will show less filtering and magnification.

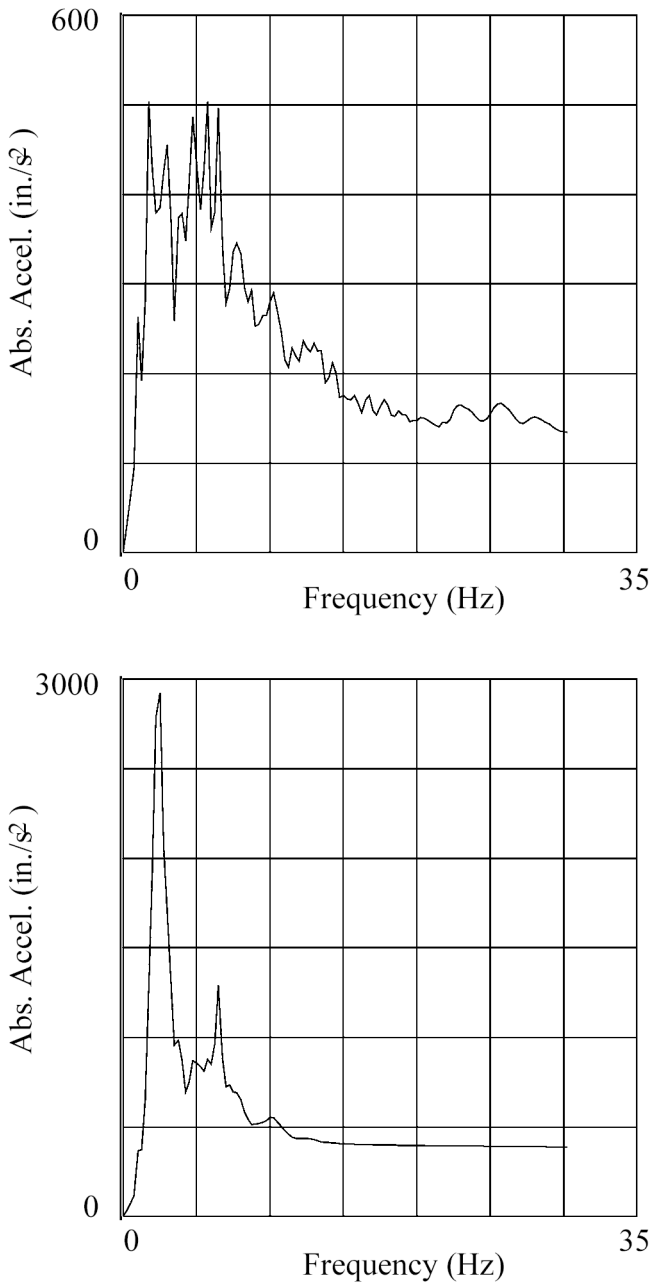



**Figure 6-3. Absolute Acceleration Response Spectrum Plots for Grid Points 999 (top) and 502 (bottom)**

A portion of the printed output file is shown in **Figure 6-4**. A portion of the absolute acceleration spectrum output for grid point 502 is shown. Spectra for the other grid points (101, 201, etc.) are also contained in the output file, as are displacement and velocity spectra. The printed format for each spectrum is similar to that of frequency response analysis, with the absolute spectrum output in the real location and the relative spectrum output in the imaginary location. Because relative acceleration spectra are not calculated, those components are zero. The punched spectrum for grid point 502 is shown in **Figure 6-5**; this output is contained in the punch file.

```
    FRACTION OF CRITICAL DAMPING =   .02        ABSOLUTE IN REAL LOCATION,  RELATIVE IN IMAG. LOCATION
    POINT-ID =     502
                                        C O M P L E X   A C C E L E R A T I O N   V E C T O R
                                                          (REAL/IMAGINARY)
FREQUENCY  TYPE          T1              T2              T3              R1              R2              R3
1.000000E-02     G     8.300698E-02   5.819805E-06   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
2.600000E-01     G     3.203262E+01   2.812203E-04   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
5.100000E-01     G     7.052912E+01   1.082366E-03   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
2.951000E+01     G     2.725781E+02   3.764749E-02   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
2.976000E+01     G     2.703750E+02   3.731322E-02   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
3.001000E+01     G     2.681875E+02   3.697610E-02   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
3.026000E+01     G     2.659297E+02   3.663826E-02   0.0             0.0             0.0             0.0
                       0.0            0.0            0.0             0.0             0.0             0.0
```

**Figure 6-4. Printed Output (Abridged)**

```
$ACCE          3      502       3                                                  1
$   2.000000E-02
TABLED1        2
         .01     .083007 .26     32.0326 .51     70.5291 .76     113.748
         1.01    365.142 1.26    371.666 1.51    648.599 1.76    1278.37
         2.01    1927.95 2.26    2781.64 2.51    2914.5  2.76    2063.92
         3.01    1700.21 3.26    1325.52 3.51    950.985 3.76    976.762
         4.01    865.546 4.26    688.777 4.51    743.792 4.76    863.43
         5.01    848.97  5.26    828.419 5.51    804.627 5.76    863.462
         6.01    835.807 6.26    949.985 6.51    1254.88 6.76    862.748
         7.01    707.527 7.26    715.255 7.51    680.692 7.76    679.488
         8.01    631.878 8.26    567.551 8.51    530.414 8.76    496.182
         9.01    499.051 9.26    501.652 9.51    506.18  9.76    514.3
         10.01   533.526 10.26   532.416 10.51   503.248 10.76   482.305
         11.01   459.648 11.26   440.015 11.51   423.567 11.76   414.748
         12.01   414.575 12.26   412.973 12.51   411.023 12.76   408.342
         13.01   403.66  13.26   396.885 13.51   389.822 13.76   386.508
         14.01   384.621 14.26   382.541 14.51   380.549 14.76   378.697
         15.01   376.879 15.26   375.092 15.51   373.363 15.76   371.656
         16.01   369.979 16.26   368.299 16.51   366.666 16.76   365.027
         17.01   363.346 17.26   361.717 17.51   360.109 17.76   358.52
         18.01   356.969 18.26   355.399 18.51   353.801 18.76   352.254
         19.01   350.629 19.26   348.945 19.51   347.336 19.76   345.734
         20.01   344.09  20.26   342.414 20.51   340.668 20.76   338.938
         21.01   337.25  21.26   335.535 21.51   333.848 21.76   332.16
         22.01   330.43  22.26   328.715 22.51   326.973 22.76   325.293
         23.01   323.609 23.26   321.914 23.51   320.195 23.76   318.484
         24.01   316.684 24.26   314.863 24.51   313.035 24.76   311.191
         25.01   309.398 25.26   307.578 25.51   305.594 25.76   303.672
         26.01   301.641 26.26   299.688 26.51   297.633 26.76   295.617
         27.01   293.648 27.26   291.57  27.51   289.516 27.76   287.5
         28.01   285.367 28.26   283.266 28.51   281.219 28.76   279.023
         29.01   276.953 29.26   274.742 29.51   272.578 29.76   270.375
         30.01   268.188 30.26   265.93  ENDT
```

-etc.-

**Figure 6-5. Punched Response Spectrum**

## Response Spectrum Application

Figure 6-6 depicts a bracket model to which the spectrum will be applied. The twelve grid points shown in the figure are connected via an RBE2 element to a separate grid point (888); this grid point has a SUPORT applied in the y direction (T2) and a large mass attached to it. The twelve grid points are

constrained in all but the T2 direction. This is the grid point to which the absolute acceleration spectrum generated in the previous run (see above) will be applied, in the y direction. Displacements, accelerations, and stresses are computed. A portion of the input file is shown in **Listing 6-2**.

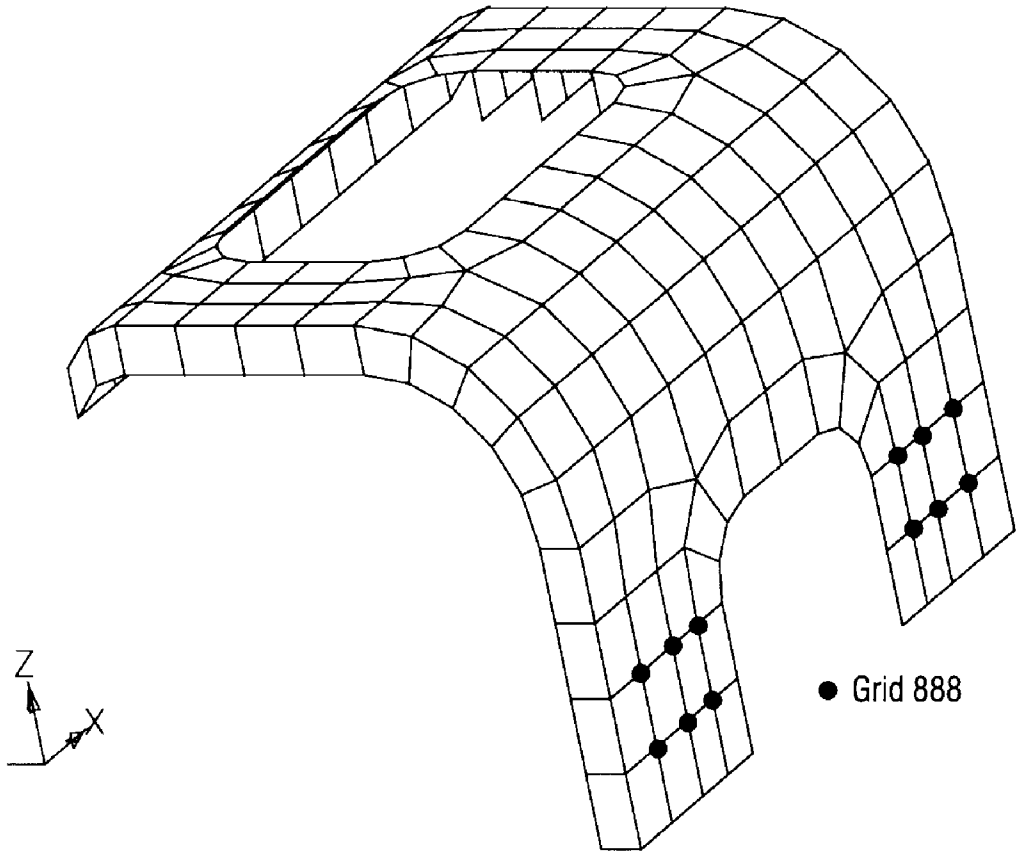



**Figure 6-6. Bracket Model**

```
$ RESPONSE SPECTRUM APPLICATION
TIME 60
SOL 103
CEND
TITLE =  BRACKET MODEL
SUBTITLE = APPLY ABS. ACCEL. SPECTRUM IN Y DIRECTION
SPC   = 1
$
METHOD = 777
$ DAMPING IS APPLIED DURING RESPONSE SPECTRUM ANALYSIS
SDAMP = 851  $ SELECTS TABDMP1 FOR MODAL DAMPING
DLOAD = 852  $ SELECTS DLOAD ENTRY
$
DISPLACEMENT = ALL
STRESS = ALL
ACCELERATION = ALL
$
$
BEGIN BULK
```

```
PARAM,AUTOSPC,YES
$
$     SID  V1  V2  ND
EIGRL,777,   ,   ,3
$
$  PERFORM SHOCK SPECTRUM ANALYSIS
PARAM,SCRSPEC,0
$
$  2% CRITICAL DAMPING
TABDMP1, 851, CRIT, , , , , , ,+TB1
+TB1, 0.0, 0.02, 50.0, 0.02, ENDT
$
$  SPECIFY SPECTRUM
DLOAD, 852, 1.0, 1.0, 1
$
$  RELATES SPECTRA LINES TO DAMPING
DTI,SPECSEL,0
DTI,SPECSEL,1, ,A, 2, 0.02
$
$  SUPORT POINT FOR SPECTRA
GRID, 888, , 3.0, 0.0, 0.0
CONM2, 888, 888, , 1.0E6
SPC, 1, 888, 13456
SUPORT, 888, 2
RBE2, 888, 888, 2, 14, 15, 22, 23, 133,+RB4
+RB4, 134, 141, 142, 145, 146, 158, 159
$
$  PUNCHED SPECTRUM FROM GENERATION RUN
$ACCE          3     502       3                                                1
$   2.000000E-02
TABLED1        2
         .01     .083007 .26     32.0326 .51     70.5291 .76     113.748
        1.01    365.142 1.26    371.666 1.51    648.599 1.76    1278.37
        2.01    1927.95 2.26    2781.64 2.51    2914.5  2.76    2063.92
        3.01    1700.21 3.26    1325.52 3.51    950.985 3.76    976.762
        4.01    865.546 4.26    688.777 4.51    743.792 4.76    863.43
        5.01    848.97  5.26    828.419 5.51    804.627 5.76    863.462
        6.01    835.807 6.26    949.985 6.51    1254.88 6.76    862.748
        7.01    707.527 7.26    715.255 7.51    680.692 7.76    679.488
        8.01    631.878 8.26    567.551 8.51    530.414 8.76    496.182
        9.01    499.051 9.26    501.652 9.51    506.18  9.76    514.3
       10.01    533.526 10.26   532.416 10.51   503.248 10.76   482.305
       11.01    459.648 11.26   440.015 11.51   423.567 11.76   414.748
       12.01    414.575 12.26   412.973 12.51   411.023 12.76   408.342
       13.01    403.66  13.26   396.885 13.51   389.822 13.76   386.508
       14.01    384.621 14.26   382.541 14.51   380.549 14.76   378.697
       15.01    376.879 15.26   375.092 15.51   373.363 15.76   371.656
       16.01    369.979 16.26   368.299 16.51   366.666 16.76   365.027
       17.01    363.346 17.26   361.717 17.51   360.109 17.76   358.52
       18.01    356.969 18.26   355.399 18.51   353.801 18.76   352.254
       19.01    350.629 19.26   348.945 19.51   347.336 19.76   345.734
       20.01    344.09  20.26   342.414 20.51   340.668 20.76   338.938
       21.01    337.25  21.26   335.535 21.51   333.848 21.76   332.16
       22.01    330.43  22.26   328.715 22.51   326.973 22.76   325.293
       23.01    323.609 23.26   321.914 23.51   320.195 23.76   318.484
       24.01    316.684 24.26   314.863 24.51   313.035 24.76   311.191
       25.01    309.398 25.26   307.578 25.51   305.594 25.76   303.672
       26.01    301.641 26.26   299.688 26.51   297.633 26.76   295.617
       27.01    293.648 27.26   291.57  27.51   289.516 27.76   287.5
       28.01    285.367 28.26   283.266 28.51   281.219 28.76   279.023
```

```
           29.01   276.953 29.26   274.742 29.51   272.578 29.76   270.375
           30.01   268.188 30.26   265.93 ENDT
$
$-------------------------------------------------------
$
$  ...  rest of model  ...
$
ENDDATA
```

**Listing 6-2. Abridged Input File for Response Spectrum Application**

Response spectrum application is invoked via PARAM,SCRSPEC,0. The DTI,SPECSEL entry specifies that the input spectrum is acceleration (denoted by the A in field 5). The TABDMP1 entry defines the modal damping (every mode has 2% critical damping). The TABLED1 entry defines the input spectrum; this is the punch file that was generated from the spectrum generation run.

### Discussion of Results

Figure 6-7 shows a portion of the resulting printed output. The eigenvalue summary shows the computed natural frequencies; note the rigid body mode, which occurred because the bracket was not constrained in the y direction (there was a SUPORT for that DOF). Matrix FN is the list of natural frequencies of the modes used for analysis with response spectrum input (two in this case—the rigid body mode is not included because our spectrum started at 0.01 Hz). Matrix PSIT lists the modal participation factors in transposed form, with one column for each mode (three in this case) and one row for each input point (one in this case). Note that the third mode (the second elastic mode) cannot be readily excited by base motion, since its response to the load is 10 orders of magnitude less than the second (first elastic) mode. Matrix UHVR occurs once for every analysis subcase with response spectrum input, and it lists the peak modal response. The first column is displacement, the second is velocity, and the third is acceleration. There is one row for each mode used in the response spectrum analysis (two in this case). A portion of the resulting maximum displacements, accelerations, and stresses—quantities selected for output via Case Control—are also shown. These quantities are computed using the ABS (default) method.

```
                                                R E A L   E I G E N V A L U E S
   MODE    EXTRACTION      EIGENVALUE            RADIANS             CYCLES           GENERALIZED         GENERALIZED
    NO.       ORDER                                                                      MASS              STIFFNESS
        1         1       0.0                 0.0                 0.0                 1.000000E+00        0.0
        2         2       3.930305E+03        6.269214E+01        9.977764E+00        1.000000E+00        3.930305E+03
        3         3       2.878402E+04        1.696586E+02        2.700200E+01        1.000000E+00        2.878402E+04

^^^ DMAP INFORMATION MESSAGE 9047 (POSTREIG) -  SCALED RESPONSE SPECTRA FOR RESIDUAL STRUCTURE ONLY

      MATRIX FN       (GINO NAME 101 ) IS A DB  PREC           1 COLUMN X           3 ROW RECTANG  MATRIX.
COLUMN      1      ROWS      2 THRU      3     ----------------------------------------------------
ROW
     2)    9.977767703979D+00  2.700200654232D+01
THE NUMBER OF NON-ZERO TERMS IN THE DENSEST COLUMN =       2
THE DENSITY OF THIS MATRIX IS  66.66 PERCENT.
              PSIT
     POINT          VALUE     POINT          VALUE     POINT          VALUE     POINT          VALUE     POINT          VALUE

COLUMN       1
          888 T2 -1.00000E+03
COLUMN       2
          888 T2  1.51209E-01
COLUMN       3
          888 T2  8.54947E-11
               U S E T   D E F I N I T I O N   T A B L E   ( I N T E R N A L   S E Q U E N C E ,   R O W   S O R T )
                                                    R    DISPLACEMENT SET
                 -1-         -2-         -3-         -4-         -5-         -6-         -7-         -8-         -9-         -10-

     1=      888-2
  SCALED SPECTRAL RESPONSE, ABS  OPTION, DLOAD =        852  CLOSE =       1.00
      MATRIX UHVR     (GINO NAME 101 ) IS A REAL             3 COLUMN X           3 ROW SQUARE   MATRIX.
COLUMN      1      ROWS      2 THRU      3     ----------------------------------------------------
ROW
     2)   -2.0431E-02 -8.7238E-13
COLUMN      2      ROWS      2 THRU      3     ----------------------------------------------------
ROW
     2)   -1.2808E+00 -1.4801E-10
COLUMN      3      ROWS      2 THRU      3     ----------------------------------------------------
ROW
     2)   -8.0299E+01 -2.5111E-08
THE NUMBER OF NON-ZERO TERMS IN THE DENSEST COLUMN =       2
THE DENSITY OF THIS MATRIX IS  66.66 PERCENT.
     TIME =  0.000000E+00
                                          D I S P L A C E M E N T   V E C T O R

     POINT ID.   TYPE         T1            T2            T3            R1            R2            R3
            5      G      1.982064E-05  5.079640E-04  1.304063E-05  4.452030E-04  0.0           3.272993E-04
            6      G      2.216715E-05  4.475303E-04  5.667240E-06  2.254874E-03  0.0           6.874480E-05
            7      G      5.777859E-05  3.471966E-03  3.236318E-05  1.011459E-02  0.0           1.185161E-02
            8      G      6.727507E-05  9.782481E-03  9.783542E-05  7.597393E-03  2.088506E-03  1.105588E-02
           13      G      4.799800E-06  2.431473E-04  1.162851E-05  2.776468E-03  0.0           1.267932E-03
           14      G      0.0           3.089312E-09  0.0           0.0           0.0           0.0
           15      G      0.0           3.089312E-09  0.0           0.0           0.0           0.0
          999      G      7.124796E-09  3.353467E-02  4.805730E-02  1.705154E-02  3.335225E-10  2.368395E-08
     TIME =  0.000000E+00
                                          A C C E L E R A T I O N   V E C T O R

     POINT ID.   TYPE         T1            T2            T3            R1            R2            R3
            5      G      7.790122E-02  1.996455E+00  5.125370E-02  1.749785E+00  0.0           1.286387E+00
            6      G      8.712371E-02  1.758932E+00  2.227400E-02  8.862347E+00  0.0           2.701882E-01
            7      G      2.270876E-01  1.364589E+01  1.271973E-01  3.975345E+01  0.0           4.658046E+01
            8      G      2.644117E-01  3.844816E+01  3.845233E-01  2.986009E+01  8.208470E+00  4.345302E+01
           13      G      1.886469E-02  9.556437E-01  4.570363E-02  1.091238E+01  0.0           4.983363E+00
           14      G      0.0           1.214195E-05  0.0           0.0           0.0           0.0
           15      G      0.0           1.214195E-05  0.0           0.0           0.0           0.0
          999      G      2.806829E-05  1.318016E+02  1.888800E+02  6.701779E+01  1.318517E-06  9.310856E-05
     TIME =  0.000000E+00
                                S T R E S S E S   I N   Q U A D R I L A T E R A L   E L E M E N T S   ( Q U A D 4 )
ELEMENT      FIBRE              STRESSES IN ELEMENT COORD SYSTEM            PRINCIPAL STRESSES (ZERO SHEAR)
  ID.       DISTANCE            NORMAL-X      NORMAL-Y      SHEAR-XY      ANGLE         MAJOR         MINOR         VON MISES
    4   -2.500000E-02       9.571945E+02  1.588326E+02  6.315743E+02  28.8527     1.305162E+03  -1.891354E+02  1.409281E+03
         2.500000E-02       8.720101E+02  4.871640E+02  7.984738E+02  38.2254     1.500919E+03  -1.417454E+02  1.576578E+03
    6   -2.500000E-02       1.215178E+04  9.352043E+03  5.037602E+03  37.2351     1.598040E+04   5.523426E+03  1.405755E+04
         2.500000E-02       1.361088E+04  8.486970E+03  3.161832E+03  25.4915     1.511842E+04   6.979430E+03  1.310577E+04
  235   -2.500000E-02       2.137487E+02  3.799783E+00  1.563414E+00   0.4266     2.137604E+02   3.788147E+00  2.118917E+02
         2.500000E-02       1.987584E+02  4.320534E+00  9.967642E-01   0.2937     1.987635E+02   4.315422E+00  1.966413E+02
```

**Figure 6-7. Printed Output (Abridged)**

## 6.6 Random Analysis with Coupled Excitations

In Simcenter Nastran, random analysis is treated as a data reduction procedure that is applied to the results of a frequency response analysis. First, the frequency response analysis is performed for sinusoidal loading conditions, $\{P_a\}$, each a separate subcase, at a sequence of frequencies $\omega_i$. Normal data reduction procedures are then applied to the output of the frequency response analysis module, resulting in a set of output quantities $u_{ja}$ $(\omega_i)$, corresponding to an output $_j$ and subcase $a$. The calculation of power spectral density (PSD) and autocorrelation (ATOC) functions for the output quantities is performed in the random analysis module.

Each loading condition subcase represents a unique random load source, which may be applied to many grid points. Typically, these loads are chosen to be unit loads such as unit "g" loads or unit pressures. You define the probabilistic magnitude of each load source with PSD functions. You can use RANDPS bulk entries to specify PSD functions from within the Simcenter Nastran input file, or you can use RANDPEX bulk entries to reference PSD functions that are contained in an external HDF5 format file.

> Note:
>
> The RANDOM case control command can reference either RANDPS or RANDPEX bulk entries, but not both.

If the load subcases are correlated, the coupling spectral density is also defined on one or more PSD functions. An example of coupled spectral density would be the forces on four wheels of a vehicle traveling over a rough road.

Whenever the forcing frequencies are different from those used to define the PSD functions and load tables, the interpolation of the loads should ideally happen at the total PSD level. However, because the response calculation occurs in different subcases prior to the random analysis calculations, this is not possible. In Simcenter Nastran the load tables are interpolated to conform to the forcing frequencies for the response calculations. The PSD functions are interpolated to conform to the forcing frequencies during the random calculations, which use the results of the response calculations.

**Figure 6-8** is a flow diagram for the random analysis module. The inputs to the module are the frequency responses, $H_{ja}(\omega_i)$, of quantities $u_j$ to loading conditions $\{P_a\}$ at frequencies $\omega_i$, and the auto- and cross-spectral densities of the loading conditions $S_a$ and $S_{ab}$. The response quantities, $S_j$, may be displacements, velocities, accelerations, internal forces, or stresses. The power spectral densities of the response quantities are calculated by different procedures depending on whether the loading conditions are correlated or uncorrelated. The spectral densities due to all sources, considered independent, are combined into one set of outputs.

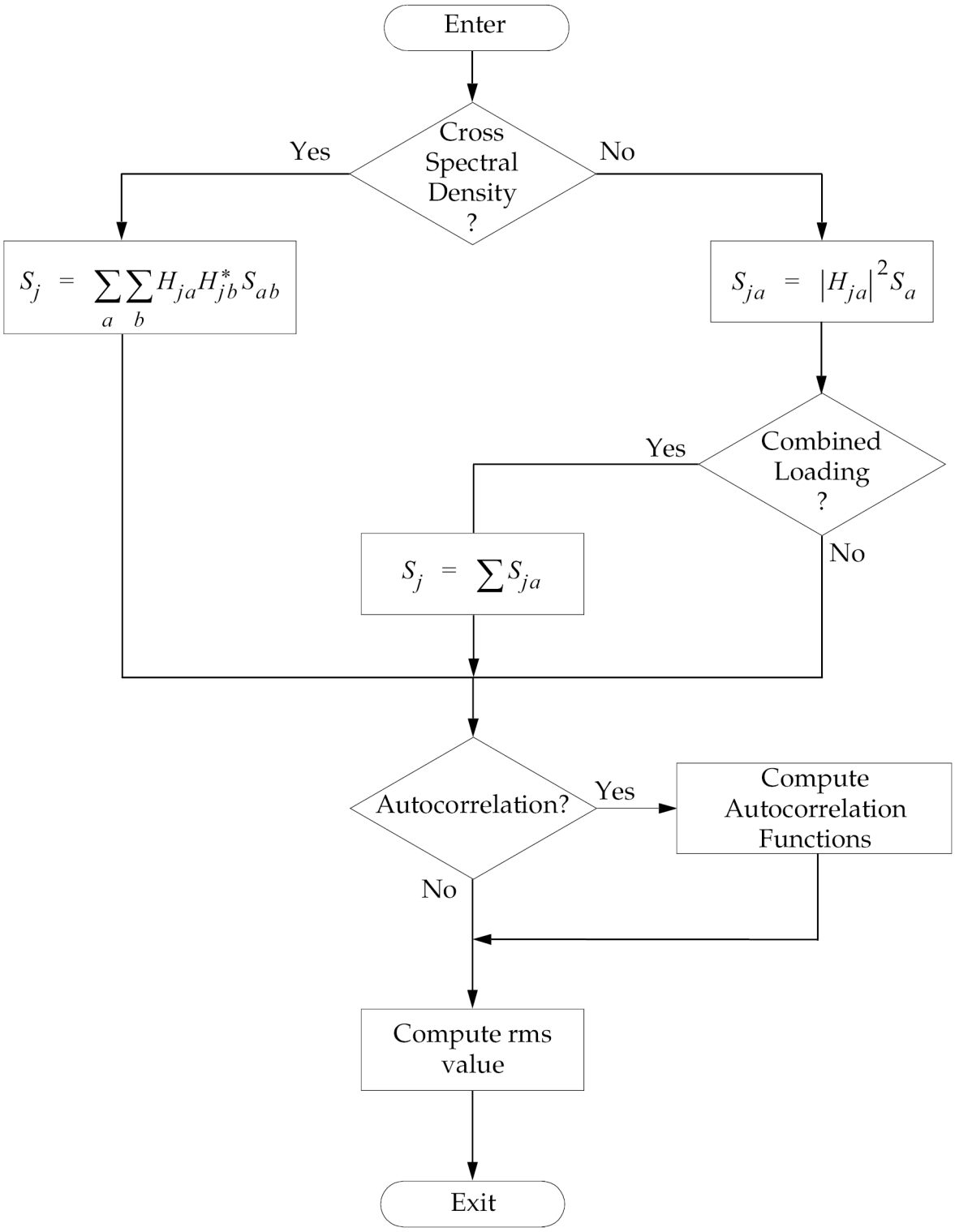



**Figure 6-8. Flow Diagram for Random Analysis Module**

## Theory

The application of these frequency response techniques to the analysis of random processes requires that the system be linear and the excitation be stationary with respect to time.

The autocorrelation (ATOC) function $R_j$, of a physical variable is given by:

$$R_j(\tau) = \lim_{T\to\infty}\left(\frac{1}{T}\int_0^T u_j(t)u_j(t-\tau)dt\right)$$

**Equation 6-50.**

where $u_j$ is a physical variable such as displacement or stress, among others, and $\tau$ is the time lag.

The RMS value of a function, $f(t)$, over all time is:

$$\bar{f} = \lim_{T\to\infty}\sqrt{\frac{1}{T}\int_0^T [f(t)]^2\,dt}$$

**Equation 6-51.**

Thus, when the time lag is set to zero, the ATOC function represents the square of the RMS value of the physical variable as follows:

$$R_j(0) = \lim_{T\to\infty}\left(\frac{1}{T}\int_0^T \left[u_j(t)\right]^2 dt\right) = \bar{u}_j^2$$

**Equation 6-52.**

where $\bar{u}_j$ is the RMS value of the physical response.

The one-sided power spectral density (PSD) of $u_j(t)$ is given by:

$$S_j(\omega) = \lim_{T\to\infty}\left(\frac{2}{T}\left|\int_0^T e^{-i\omega t}u_j(t)dt\right|^2\right)$$

**Equation 6-53.**

Because the ATOC function and the PSD are Fourier transforms of one another, the ATOC function can be expressed in terms of frequency as follows:

$$R_j(\tau) = \frac{1}{2\pi}\int_0^\infty S_j(\omega)\cos(\omega\tau)d\omega$$

**Equation 6-54.**

Thus, the square of the RMS value of the physical response is also given as follows:

$$\bar{u}_j^2 = R_j(0) = \frac{1}{2\pi}\int_0^\infty S_j(\omega)d\omega$$

**Equation 6-55.**

The expected value of the number of zero crossings with positive slope per unit time is often used in fatigue analysis models. It is calculated from the PSD as follows:

$$N_0 = \sqrt{\frac{\frac{1}{(2\pi)^3}\int_0^\infty \omega^2 S_j(\omega)d\omega}{\frac{1}{2\pi}\int_0^\infty S_j(\omega)d\omega}}$$

**Equation 6-56.**

## Transfer Function Theorem

Suppose that $u_j(\omega) = H_{ja}(\omega)\cdot Q_a(\omega)$, where $H_{ja}(\omega)$ is the frequency response of a physical variable $u_j(t)$ when the system is excited by load $Q_a(t)$. If $u_j(\omega)$ is the Fourier transform of $u_j(t)$, and $Q_a(\omega)$ is the Fourier transform of $Q_a(t)$, then the PSD of the response $S_j(\omega)$, is related to the PSD of the source, $S_a(\omega)$, as follows:

$$S_j(\omega) = \left|H_{ja}(\omega)\right|^2 \cdot S_a(\omega)$$

**Equation 6-57.**

Thus, it is valid to use frequency response techniques to evaluate statistical properties of a system such as its ATOC function when the system is subjected to random excitation. In addition, if multiple excitation sources are present and they are statistically independent of one another, then the PSD of the total response is the sum of the responses due to individual sources as follows:

$$S_j(\omega) = \sum_a S_{ja}(\omega) = \sum_a \left|H_{ja}(\omega)\right|^2 S_a(\omega)$$

**Equation 6-58.**

If the excitation sources are statistically correlated, then the PSD of the total response is as follows:

$$S_j = \sum_a \sum_b H_{ja} H_{jb}^* S_{ab}$$

**Equation 6-59.**

where $H_{jb}^*$ is the complex conjugate of $H_{jb}$, and the degree of correlation is expressed by the cross-spectral density, $S_{ab}$.

## Cross-Power Spectral Density and Cross-Correlation Functions

The cross-correlation function between a pair of physical responses is as follows:

$$R_{ab}(\tau) = \lim_{T\to\infty} \left( \frac{1}{T} \int_o^T u_a(t) u_b(t-\tau) dt \right)$$

**Equation 6-60.**

It the physical responses are statistically independent, then the cross-correlation function is null.

The cross-power spectral density (cross-PSD) of a pair of physical responses is as follows:

$$S_{ab}(\omega) = \lim_{T\to\infty}\left[\frac{2}{T}\left(\int_0^T u_a(t)e^{-i\omega t}dt\right)\left(\int_0^T u_b(t)e^{i\omega t}dt\right)\right] = 2\int_{-\infty}^{+\infty} R_{ab}(\tau)e^{-i\omega\tau}d\tau$$

**Equation 6-61.**

where $R_{ab}(\tau)$ and $S_{ab}(\omega)$ are related as follows:

$$R_{ab}(\tau) = \frac{1}{4\pi}\int_{-\infty}^{+\infty} S_{ab}(\omega)e^{i\omega t}d\omega$$

**Equation 6-62.**

The cross-PSD is a complex-valued function of angular frequency. The following property applies to a cross-PSD:

$$S_{ab}(\omega) = S_{ab}^{*}(-\omega)$$

**Equation 6-63.**

where $S_{ab}^{*}(\omega)$ is the complex conjugate of $S_{ab}(\omega)$.

Expressing Equation 6-63 in terms of real and imaginary parts yields the following:

$$S_{ab}^{\mathrm{Re}}(\omega) + iS_{ab}^{\mathrm{Im}}(\omega) = S_{ab}^{\mathrm{Re}}(-\omega) - iS_{ab}^{\mathrm{Im}}(-\omega)$$

**Equation 6-64.**

By equating the real parts of Equation 6-64, it is apparent that the real part of the cross-PSD is an even function.

$$S_{ab}^{\mathrm{Re}}(\omega) = S_{ab}^{\mathrm{Re}}(-\omega)$$

**Equation 6-65.**

By equating the imaginary parts of Equation 6-64, it is apparent that the imaginary part of the cross-PSD is an odd function.

$$S_{ab}^{\mathrm{Im}}(\omega) = -S_{ab}^{\mathrm{Im}}(-\omega)$$

**Equation 6-66.**

Making use of the even/odd property, Equation 6-62 can be rewritten as follows:

$$R_{ab}(\tau) = \frac{1}{2\pi}\int_0^\infty \left[S_{ab}^{\mathrm{Re}}(\omega)\cos(\omega t) - S_{ab}^{\mathrm{Im}}(\omega)\sin(\omega t)\right] dt$$

**Equation 6-67.**

When the software evaluates this integral, it uses a trapezoidal numerical integration scheme.

For a system subject to multiple excitation sources, the cross-PSD of the physical responses is related to the PSDs of excitation sources by the frequency response functions as follows:

$$S_{ab}(\omega) = \sum_j \sum_k H_{aj}(\omega) H_{bk}^*(\omega) S_{jk}(\omega)$$

**Equation 6-68.**

where $H_{aj}(\omega)$ is the frequency response function that corresponds to the excitation source $Q_j(t)$, $H_{bk}(\omega)$ is the frequency response function that corresponds to the excitation source $Q_k(t)$, $H_{bk}^*(\omega)$ is the complex conjugate of $H_{bk}(\omega)$, and $S_{jk}(\omega)$ is the cross-PSD of excitation sources when $j \neq k$.

For the case where the excitation sources are statistically independent, the cross-PSD of the excitation sources is null.

## Numerical Integration Options for Calculating RMS Values

From Equation 6-55, the RMS value of a physical response is given by the following:

$$\bar{u}_j = \sqrt{\frac{1}{2\pi}\int_0^\infty S_j(\omega) d\omega}$$

**Equation 6-69.**

The software evaluates the integral in Equation 6-69 over a specified range of frequencies as follows:

$$\int_0^{\infty} S_j(\omega)d\omega \simeq \sum_{i=1}^{N-1}\left(\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)d\omega\right)$$

**Equation 6-70.**

where $i = 1, 2, ..., N$ represents *N* sub-intervals of the frequency range.

Thus, the RMS value of the physical response is approximated by the following relation:

$$\bar{u}_j \simeq \sqrt{\frac{1}{2\pi}\sum_{i=1}^{N-1}\left(\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)d\omega\right)}$$

**Equation 6-71.**

Similarly, the expected value of the number of zero crossings with positive slope per unit time as given by Equation 6-56 is approximated by the following relation:

$$N_0 \simeq \sqrt{\frac{\frac{1}{(2\pi)^3}\sum_{i=1}^{N-1}\left(\int_{\omega_i}^{\omega_{i+1}} \omega^2 S_j(\omega)d\omega\right)}{\frac{1}{2\pi}\sum_{i=1}^{N-1}\left(\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)d\omega\right)}}$$

**Equation 6-72.**

The software supports the following numerical integration options:

- Numerical integration in linear space.
  This method is most accurate when a plot of the PSD versus frequency produces a straight line on a Cartesian plot with linear axes.

- Numerical integration in logarithmic space.
  This method is most accurate when a plot of the PSD versus frequency produces a straight line on a Cartesian plot with logarithmic axes.

> Note:
>
> Use the RMSINT parameter to specify the numerical integration option.

## Numerical Integration in Linear Space

When you select numerical integration in linear space, within a given frequency range, $\omega_i \leq \omega \leq \omega_{i+1}$, the PSD is assumed to vary linearly as follows:

$$S_j(\omega) = S_j(\omega_i) + \left( \frac{S_j(\omega_{i+1}) - S_j(\omega_i)}{\omega_{i+1} - \omega_i} \right) (\omega - \omega_i)$$

**Equation 6-73.**

Integrating Equation 6-73 over the frequency sub-interval $\omega_i \leq \omega \leq \omega_{i+1}$ yields the following:

$$\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)\, d\omega = \frac{1}{2} \left( S_j(\omega_{i+1}) + S_j(\omega_i) \right) (\omega_{i+1} - \omega_i)$$

**Equation 6-74.**

Substituting Equation 6-74 into Equation 6-71 gives the RMS value of the physical response as follows:

$$\bar{u}_j \simeq \sqrt{\frac{1}{4\pi} \sum_{i=1}^{N-1} \left[ \left( S_j(\omega_{i+1}) + S_j(\omega_i) \right) (\omega_{i+1} - \omega_i) \right]}$$

**Equation 6-75.**

Similarly, the expected value of the number of zero crossings with positive slope per unit time is given by the following relation:

$$N_0 \simeq \sqrt{\frac{\frac{1}{(2\pi)^3} \sum_{i=1}^{N-1} \left( \int_{\omega_i}^{\omega_{i+1}} \omega^2 S_j(\omega) d\omega \right)}{\frac{1}{4\pi} \sum_{i=1}^{N-1} \left[ \left( S_j(\omega_{i+1}) + S_j(\omega_i) \right) (\omega_{i+1} - \omega_i) \right]}}$$

**Equation 6-76.**

where:

$$\int_{\omega_i}^{\omega_{i+1}} \omega^2 S_j(\omega) d\omega = \frac{4\left(S_j(\omega_i)\omega_{i+1} - S_j(\omega_{i+1})\omega_i\right)\left(\omega_{i+1}^3 - \omega_i^3\right) + 3\left(S_j(\omega_{i+1}) - S_j(\omega_i)\right)\left(\omega_{i+1}^4 - \omega_i^4\right)}{12\left(\omega_{i+1} - \omega_i\right)}$$

### Numerical Integration in Logarithmic Space

When you select numerical integration in logarithmic space, within a given frequency range, $\omega_i \leq \omega \leq \omega_{i+1}$, the PSD is assumed to vary as follows:

$$S_j(\omega) = B\omega^m$$

**Equation 6-77.**

where *B* and *m* are constants. Taking the logarithm of Equation 6-77 yields the following:

$$\log_{10} S_j(\omega) = \log_{10} B + m\log_{10}\omega$$

**Equation 6-78.**

which is a straight line in log-log space.

Substituting $X = \log_{10}\omega$ and $Y = \log_{10} S_j(\omega)$ into Equation 6-78 yields the following:

$$Y = mX + b$$

**Equation 6-79.**

where $b = \log_{10} B$.

Using the end points of the frequency range to evaluate *m* yields:

$$m = \frac{\log_{10} S_j(\omega_{i+1}) - \log_{10} S_j(\omega_i)}{\log_{10}\omega_{i+1} - \log_{10}\omega_i} = \frac{\log_{10}\left(\frac{S_j(\omega_{i+1})}{S_j(\omega_i)}\right)}{\log_{10}\left(\frac{\omega_{i+1}}{\omega_i}\right)}$$

**Equation 6-80.**

Consequently, the interpolated value of the PSD at $\omega_i \leq \omega \leq \omega_{i+1}$ is:

$$\log_{10} S_j(\omega) = \log_{10} S_j(\omega_i) + m\left(\log_{10}\omega - \log_{10}\omega_i\right) = \log_{10} S_j(\omega_i)(\omega/\omega_i)^m$$

**Equation 6-81.**

Equation 6-81 simplifies to the following:

$$S_j(\omega) = \frac{S_j(\omega_i)}{\omega_i^m}\,\omega^m$$

**Equation 6-82.**

Integrating Equation 6-82 over the frequency sub-interval $\omega_i \le \omega \le \omega_{i+1}$ yields the following:

$$\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)\,d\omega = \int_{\omega_i}^{\omega_{i+1}} \frac{S_j(\omega_i)}{\omega_i^m}\,\omega^m d\omega = \frac{S_j(\omega_i)}{(m+1)\,\omega_i^m}\left(\omega_{i+1}^{m+1} - \omega_i^{m+1}\right) \text{ for } m \neq -1$$

$$\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)\,d\omega = \int_{\omega_i}^{\omega_{i+1}} \frac{S_j(\omega_i)}{\omega_i^m}\,\omega^m d\omega = \omega_i S_j(\omega_i)\ln\left(\frac{\omega_{i+1}}{\omega_i}\right) \text{ for } m = -1$$

**Equation 6-83.**

Thus, the RMS value for the physical quantity is as follows:

$$\bar{u}_j \simeq \sqrt{\frac{1}{2\pi}\sum_{i=1}^{N-1}\left(\frac{S_j(\omega_i)}{(m+1)\,\omega_i^m}\left(\omega_{i+1}^{m+1} - \omega_i^{m+1}\right)\right)} \text{ for } m \neq -1$$

$$\bar{u}_j \simeq \sqrt{\frac{1}{2\pi}\sum_{i=1}^{N-1}\left(\frac{S_j(\omega_i)}{\omega_i^m}\ln\left(\frac{\omega_{i+1}}{\omega_i}\right)\right)} \text{ for } m = -1$$

**Equation 6-84.**

Similarly, the expected value of the number of zero crossings with positive slope per unit time is given by the following relation:

$$N_0 \simeq \sqrt{\frac{\frac{1}{(2\pi)^3}\sum_{i=1}^{N-1}\left(\int_{\omega_i}^{\omega_{i+1}} \omega^2 S_j(\omega)d\omega\right)}{\frac{1}{2\pi}\sum_{i=1}^{N-1}\left(\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)d\omega\right)}}$$

**Equation 6-85.**

where the integral in the numerator of Equation 6-85 evaluates to the following:

$$\int_{\omega_i}^{\omega_{i+1}} \omega^2 S_j(\omega)d\omega = \frac{S_j(\omega_i)}{(m+3)\,\omega_i^m}\left[(\omega_{i+1})^{m+3} - (\omega_i)^{m+3}\right] \text{ for } m \neq -3$$

$$\int_{\omega_i}^{\omega_{i+1}} \omega^2 S_j(\omega)d\omega = \omega_i^3 S_j(\omega_i)\ln\left(\frac{\omega_{i+1}}{\omega_i}\right) \text{ for } m = -3$$

**Equation 6-86.**

and the integral in the denominator of Equation 6-85 evaluates to the following:

$$\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)\,d\omega = \frac{S_j(\omega_i)}{(m+1)\,\omega_i^m}\left(\omega_{i+1}^{m+1} - \omega_i^{m+1}\right) \text{ for } m \neq -1$$

$$\int_{\omega_i}^{\omega_{i+1}} S_j(\omega)\,d\omega = \omega_i S_j(\omega_i)\ln\left(\frac{\omega_{i+1}}{\omega_i}\right) \text{ for } m = -1$$

**Equation 6-87.**

## Cumulative Root Mean Square

The cumulative RMS function of a physical response at a set of specified frequencies is evaluated from the following relation:

$$\bar{u}_j(\omega_i) = \sqrt{\frac{1}{2\pi}\int_{\omega_1}^{\omega_i} S_j(\omega)d\omega}$$

**Equation 6-88.**

where $i = 1, 2, ..., N$.

## Random Analysis Output

You can request output from a random analysis by specifying the PSDF, ATOC, CRMS, RMS, or RALL describer on the ACCELERATION, DISPLACEMENT, FORCE, MPCFORCES, OLOAD, SPCFORCES, STRAIN, STRESS, and VELOCITY case control commands.

| | |
|---|---|
| **PSDF** | Requests the power spectral density (PSD) function be calculated and output. |
| **ATOC** | Requests the autocorrelation (ATOC) function be calculated and output. |
| **CRMS** | Requests the cumulative root-mean-square (CRMS) function be calculated and output. |
| **RMS** | Requests the root-mean-square (RMS) and the zero crossing functions be calculated and output. |
| **RALL** | Requests the PSD, ATOC, CRMS, RMS, and zero crossing functions be calculated and output. |

> Note:
>
> To scale the RMS and CRMS output, use the parameter RMSSF.

By default, frequency response output is suppressed during a random analysis. To obtain the frequency response output, specify SYSTEM(524)=1.

### RMS von Mises Stress

If you request RMS output from the STRESS case control command, RMS von Mises stress is included in the stress output for those elements that support von Mises stress calculation. The software also calculates the zero-mean crossing for all of the elements and at all of the same locations that the software calculates the RMS von Mises stress. The zero-mean crossing represents the apparent (or dominant) frequency of the response.

When the software calculates zero-mean crossings, it uses the computed spectral density function of the RMS result. You can specify the interpolation method that the software uses to obtain the zero-mean crossings with the RMSINT describer on the RANDOM case control command.

- To use the trapezoidal approximation, specify RMSINT = LINEAR.
- To use log-log interpolation, specify RMSINT = LOGLOG.

The methodology used to calculate the RMS von Mises stress is based on the Sandia Report SAND98-0260 from Sandia National Laboratories titled “An Efficient Method for Calculating RMS von Mises Stress in a Random Vibration Environment”. This methodology is as follows.

Consider the following quadratic function of stress:

$$p(t)^2 = \sigma^T A \sigma$$

where *A* is a symmetric, constant, positive semi-definite matrix.

In the case of von Mises stress:

$$p(t)^2 = \sigma_{xx}^2 + \sigma_{yy}^2 + \sigma_{zz}^2 - (\sigma_{xx}\sigma_{yy} + \sigma_{xx}\sigma_{zz} + \sigma_{yy}\sigma_{zz}) + 3(\sigma_{xy}^2 + \sigma_{xz}^2 + \sigma_{yz}^2)$$

and

$$[A] = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} & & & \\ -\frac{1}{2} & 1 & -\frac{1}{2} & & & \\ -\frac{1}{2} & -\frac{1}{2} & 1 & & & \\ & & & 3 & & \\ & & & & 3 & \\ & & & & & 3 \end{bmatrix}$$

The RMS value for von Mises stress in frequency domain, $P_{RMS}$, is given by

$$P_{RMS} = \sqrt{\langle p^2 \rangle}$$

After some trigonometric manipulations, the time-averaged value of the square of von Mises stress, $\langle p^2 \rangle$, is

$$\langle p^2 \rangle = \frac{1}{2} \sum_{n=1}^{N_\omega} \left[ \hat{\sigma}_n^H A \hat{\sigma}_n \right]$$

where $(\ )^H$ denotes the Hermitian operator (complex conjugate transpose).

To be useful, the above expressions must be expressed in terms of the input forces

$$\langle p^2 \rangle = \frac{1}{2} \sum_{n=1}^{N_\omega} \hat{f}_n^H H_{\sigma,n}^H A H_{\sigma,n} \hat{f}_n$$

With ensemble averaging, this expression can be expressed in terms of the input cross spectral density matrix

$$\left\langle p^2 \right\rangle = \frac{1}{T}\sum_{n=1}^{N_\omega}\sum_{a,a'}^{N_F}(H_{\sigma,n}^{H}AH_{\sigma,n})_{a,a'}S_F(n)_{a,a'}$$

## Laminate Output

The PCOMP and PCOMPG property entries can be used to model a laminate with the shell elements CTRIA3, CTRIA6, CTRIAR, CQUAD4, CQUAD8, and CQUADR. The PCOMPS property entry can be used to model a laminate with the solid elements CHEXA and CPENTA. You can use the PSDF, ATOC, CRMS, RMS, or RALL describers on the STRESS or STRAIN case control commands to request random output for an element which is defined as a laminate. The requested random output is computed and output for each ply. To define a pair of response quantities for computing the cross-power spectral density function and cross-correlation functions in random analysis, use the RCROSSC bulk entry.

The following output is supported for shell and solid laminates in random analysis:

- Stress and strain output for the individual lamina in either SORT1 or SORT2 format.
- Stress resultants, which are requested with the FORCE case control command.
- Interlaminar shear stress in the bonding material.
- The failure indices and strength ratios.
  Because frequency response analysis results are complex, the software calculates the failure indices and strength ratios at discrete phase angles over a full 360 degree range. The worst case value is output. By default, the calculation is performed in one degree increments. However, you can optionally specify that the software use a different angular increment by using the parameter SWPANGLE.

For more information on random analysis output for laminates, see the remarks on the PCOMP, PCOMPG, and PCOMPS bulk entries in the *Quick Reference Guide* and the “Laminates” section of the *Simcenter Nastran User’s Guide*.

## Solving for Multiple PSD Functions

When you want to compute multiple random responses, the software can use the results from the same frequency response subcases in multiple random responses. You can use this capability by organizing your subcases as follows:

- For modal frequency response, place the modal subcase above the other subcases in the input file.
- Below the modal subcase, include a subcase for each frequency response that you want to use in a random spectrum calculation. Each of these subcases must reference the same set of frequencies.
- After the frequency response subcases, include a subcase for each random spectrum that you want to evaluate.

For example, suppose that a structure is excited by two loads, and you want to evaluate the random response of the structure for two PSD functions using SOL 111. To do so, you can organize the subcases as follows:

```
SUBCASE 1
$
$ Subcase 1 calculates the normal modes
$
ANALYSIS=MODES
DISP=ALL
$
SUBCASE 2
$
$ Subcase 2 calculates the frequency response of the structure to the
$ loading specified by DLOAD 111 at the frequencies specified by
$ FREQUENCY set 13
$
FREQUENCY=13
DLOAD=111
$
SUBCASE 3
$
$ Subcase 3 calculates the frequency response of the structure to the
$ loading specified by DLOAD 211 at the frequencies specified by
$ FREQUENCY set 13
$
FREQUENCY=13
DLOAD=211
$
SUBCASE 4
$
$ Subcase 4 uses the frequency responses from Subcases 2 and 3 to
$ calculate the random response of the structure for the PSD function
$ specified by RANDOM 100. RANDOM 100 references RANDPS and RANDPEX
$ bulk entries with SID 100.
$
ANALYSIS=RANDOM
RANDOM=100
$
SUBCASE 5
$
$ Subcase 5 uses the frequency responses from Subcases 2 and 3 to
$ calculate the random response of the structure for the PSD function
$ specified by RANDOM 200. RANDOM 200 references RANDPS and RANDPEX
$ bulk entries with SID 200.
$
ANALYSIS=RANDOM
RANDOM=200
```

As you can see, in the above subcase organization, the software calculates each frequency response once regardless of the number of PSD functions that you want to consider.

Note:

To use multiple frequency responses in a random calculation:

- For SOL 108 or 111, each frequency response must reference the same set of frequencies.
- For SOL 111, if SDAMPING is specified, each frequency response must reference the same SDAMPING specification.

## Example: Random Analysis of a Simple Structure

There are a number of physical phenomena that result in nondeterministic data where specific responses cannot be predicted in a deterministic sense. Examples are earthquake ground motion, the height and spacing of ocean waves, pressure gusts encountered by aircraft in flight, by bridges during storms, and acoustic excitation such as jet engine noise. The response of physical systems to such unpredictable values of excitation is an application of random analysis methods.

In Simcenter Nastran, ergodic random analysis is treated as a data reduction process that expands the results of a frequency response analysis.

The inputs for these data recovery activities are output quantities from the frequency response analysis, and user-supplied spectral data for the loading conditions. The outputs can include the PSD and ATOC functions, the RMS and CRMS values of the response, and the number of positive crossings of the line $y(t) = 0$ by the response record in a unit time.

The number of positive crossings and the RMS value of the response are of particular importance to the stress analyst. The number of positive crossings, $N_0$, is obtained from the joint probability density function and is a measure of the apparent frequency of the response. The RMS value of the response is a measure of the dynamic magnitudes.

### Model Description

The model illustrated here consists of three masses connected by springs and viscous dampers. The base mass, $M_1$, is considered to be the ground and is used to enforce a excitation. The base is excited by a noise excitation displacement $y$ (t) having a white noise PSD $S_2(f)$in-lb/Hz. The middle mass is excited by a band-limited white noise excitation force $F(t)$ having a constant PSD $S_1(f)$ in-lb$^2$/Hz. The two loadings are correlated through a cross-spectral density.

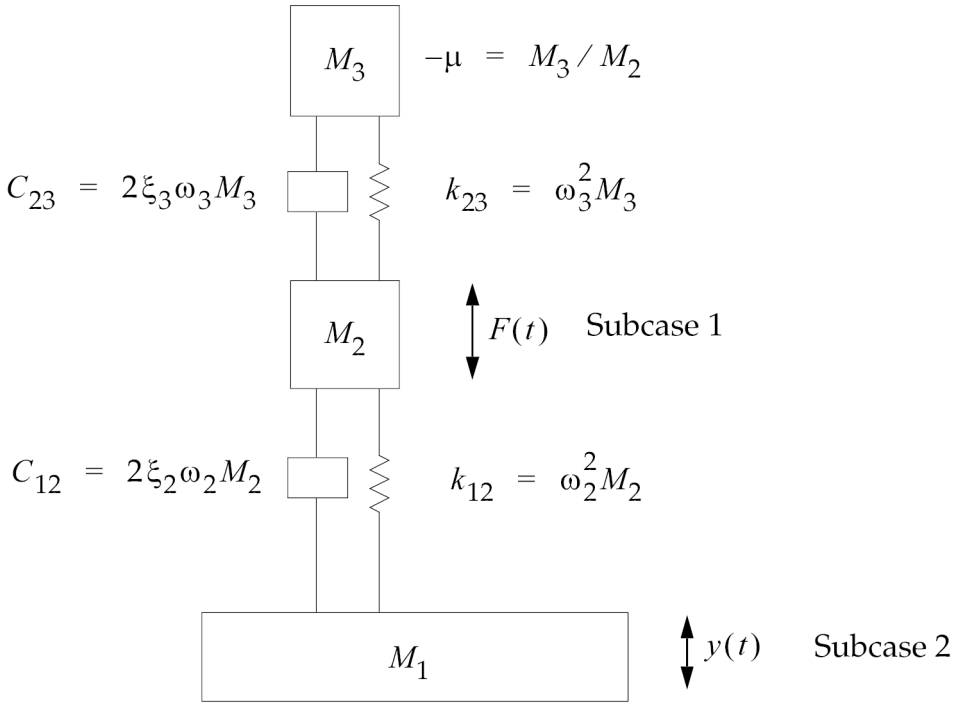



**Figure 6-9. Example Model**

In the figure, the following notation has been used:

| | |
|---|---|
| $M_i$ | Mass of i-th grid point. |
| $k_{ij}$ | Stiffness between i-th and j-th grid points. |
| $C_{ij}$ | Viscous damping between i-th and j-th grid points. |
| $2\xi_j$ | Nondimensional damping ratio = $C_{ij} / \sqrt{k_{ij} M_j}$ |
| $\omega_j$ | Circular frequency = $\sqrt{k_{ij} / M_j}$ |

| Mass Properties | | |
|---|---|---|
| **EID** | **G** | **M** |
| 1 | 1 | 1.0E9 lb-sec$^2$ / in |
| 2 | 2 | 0.1 lb-sec$^2$/in |
| 3 | 3 | 1.0 lb-sec$^2$/in |

| | Spring Properties |
|---|---|
| **EID** | **K** |
| 12 | 5.0E6 lb/in |
| 23 | 10.0E6 lb/in |

| Damper Properties | |
|---|---|
| **EID** | **B** |
| 120 | 14.14 lb-sec/in |
| 230 | 63.25 lb-sec/in |

| Constraints | |
|---|---|
| **Node** | **Constraint** |
| 1 | 13456 |
| 2 | 13456 |
| 3 | 13456 |

| Frequencies (f) |
|---|
| 200 Hz to 600 Hz by 10 Hz |

| Loads | | |
|---|---|---|
| **Node** | **Load** | **Subcase** |
| 2 | $F_y = 100.$ | 1 |
| 1 | $F_y = (1.0E9)(2\pi f)^2$ lb | 2 |

| Spectral Densities | | |
|---|---|---|
| **Excited Set (j)** | **Applied Set (h)** | $S_{jk}$ |
| 1 | 1 | $1 \times 10^5 lb^2/Hz$ |
| 2 | 2 | $1 \times 10^4 in^2/Hz$ |
| 1 | 2 | 3 in-lb-Hz |

For a check on the Simcenter Nastran results, the responses at 200 Hz, 280 Hz, and 500 Hz are tabulated. Results are given as magnitude/phase format.

| Frequency | Response | Theoretical | Simcenter Nastran |
|---|---|---|---|
| 200 Hz | $100 \cdot H_{2f-d}$ | $3.3705 \times 10^{-5}$ / 359.6029° | $3.3705 \times 10^{-5}$ / 359.6029° |
| | $100 \cdot H_{3f-d}$ | $4.0025 \times 10^{-5}$ / 359.5175° | $4.0025 \times 10^{-5}$ / 359.5175° |
| | $H_{2d-d}$ | 2.0013 / 179.7211° | 1.6853 / 179.8065° |
| | $H_{3d-d}$ | 1.6853 / 179.8065° | 2.0013 / 179.7211° |
| 280 Hz | $100 \cdot H_{2f-d}$ | $4.6806 \times 10^{-4}$ / 347.2289° | $4.6806 \times 10^{-4}$ / 347.2288° |
| | $100 \cdot H_{3f-d}$ | $6.7782 \times 10^{-4}$ / 346.9431° | $6.7782 \times 10^{-4}$ / 346.9430° |
| | $H_{2d-d}$ | 23.4031 / 167.5139° | 23.4031 / 167.5138° |
| | | 33.8912 / 167.2282° | 33.8911 / 167.2281° |

| | | | |
|---|---|---|---|
| | $H_{3d-d}$ | | |
| 500 Hz | $100 \cdot H_{2f-d}$<br>$100 \cdot H_{3f-d}$<br>$H_{2d-d}$<br>$H_{3d-d}$ | $2.4206 \times 10^{-7}$ / 236.0504°<br>$1.0187 \times 10^{-5}$ / 180.4626°<br>$1.2103 \times 10^{-2}$ / 56.5594°<br>.50935 / .9717° | $2.4025 \times 10^{-7}$ / 236.0506°<br>$1.0187 \times 10^{-5}$ / 180.4626°<br>$1.2103 \times^{-2}$ / 56.5600°<br>0.50934 / .9717° |

The auto-power spectral density of the j-th response is defined by the relationship

$$S_j(f) = \sum_a \sum_b H_a^*(f) S_{ab} H_b(f)$$

**Equation 6-89.**

The results for selected frequencies are tabulated below.

| Frequency | PSD $M_2$ | |
|---|---|---|
| | Theoretical | Simcenter Nastran |
| 200 Hz | $5.6805 \times 10^{-5}$ | $5.6803 \times 10^{-5}$ |
| 280 Hz | $1.0955 \times 10^{-2}$ | $1.0955 \times 10^{-2}$ |
| 500 Hz | $2.9307 \times 10^{-9}$ | $2.9305 \times 10^{-9}$ |

As discussed in the problem description, the number of positive crossings $N_0$ is a measure of the apparent response frequency. This measure is compared below.

| Frequency 1st Mode | $N_0$ | |
|---|---|---|
| | Theoretical $M_2$ | Displacement $M_2$ |
| 284.2 Hz | 282.4 Hz | 284.3 Hz |

```
ID FREQRSPN,RANDOM
TIME 5
DIAG 8
SOL 111
CEND
TITLE=FREQ RSPN - RANDOM ANALYSIS DEMONSTRATION
ECHO=UNSORT
METHOD=1  $ Reduce to Modal Coordinates
FREQUENCY=1
RANDOM=1      $ Basic request
SVEC=ALL      $ Modal Coordinate output
SDISP(SORT2,PHASE)=ALL
DISP(SORT2,PHASE)=ALL
VELO(SORT2,PHASE,PLOT)=ALL
```

```
ACCEL(SORT2,PHASE,PLOT)=ALL
ELFORCE(SORT2,PHASE,PLOT)=ALL
ELSTRESS(SORT2,PHASE,PLOT)=ALL
SUBCASE 1
LABEL=FORCING FUNCTION ON MASS AT GRID POINT NUMBER 2
DLOAD=1
SUBCASE 2
LABEL=ENFORCED DISPLACEMENT ON GRID POINT NUMBER 1
DLOAD=2
$  ******************************************************
$  *** RANDOM ANALYSIS CASE CONTROL ***
OUTPUT (XYOUT)
XGRID=YES
YGRID=YES
XTITLE=FREQUENCY (HERTZ)
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
XTITLE=FREQUENCY (HERTZ)
YTITLE=VELOCITY RESPONSE LOAD CASE 1
YLOG=YES
XYPLOT VELO RESPONSE 1 /2(T2RM),3(T2RM)
YLOG=NO
YTITLE=PHASE
XYPLOT VELO RESPONSE 1 /2(T2IP),3(T2IP)
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
YLOG=YES
YTITLE=DISP P S D
XYPLOT DISP PSDF /2(T2)/3(T2)
YTITLE=ACCE P S
XYPLOT ACCE PSDF /2(T2)/3(T2)
YTITLE=FORCE P S D
XYPLOT ELFORCE PSDF /12(2)/23(2)
XTITLE=TIME (SEC)
YLOG=NO
YTITLE=VELOCITY AUTO CORROLATION
XYPLOT VELO AUTO /2(T2)/3(T2)
BEGIN BULK
$  *** STRUCTURE DEFINITION ***
GRID 1  0. 0. 0.  13456
GRID 2  0. 1. 0.  13456
GRID 3  0. 2. 0.  13456
CONM2 1 1  1.+9
CONM2 2 2  .1
CONM2 3 3  1.
CELAS2 12 5.+6 1 2 2 2               .5
CELAS2 23 10.+6 2 2 3 2               .5
CDAMP2 120 14.14 1 2 2 2
CDAMP2 230 63.25 2 2 3 2
SUPORT 1 2
$  *** EIGENVALUE SETUP ***
EIGR 1 MGIV    3   EIGR1
+EIGR1 MASS
$  *** FREQUENCY SELECTION ***
FREQ1 1 200. 10. 40
$  *** APPLIED DYNAMIC LOAD SUBCASE 1 ***
DAREA 10 2 2 1.+2
RLOAD1 1 10   10
TABLED1 10        TABD1
+TABD1 1. 1. 10000. 1. ENDT
$  *** ENFORCED DISPLACEMENT SUBCASE 2 ***
DAREA 20 1 2 1.+9
```

```
   RLOAD1 2 20   110
   TABLED4 110 0. 1. 1. 10000.    TABD4
   +TABD4 0. 0. 39.478 ENDT
   $  *** RANDOM ANALYSIS DATA ***
   RANDPS 1 1 1 1.+5  1000
   RANDPS 1 2 2 1.-4  1000
   RANDPS 1 1 2 3.  1000
   TABRND1 1000        TABR1
   +TABR1 0. 0. 199.99 0. 200. 1. 600. 1. TABR2
   +TABR2 600.01 0. 10000. 0. ENDT
   RANDT1 1 100 0. .1
   ENDDATA
```

**Listing 6-3. Data File for Example from Figure 6-9**

The following is the key printed output:

```
  FREQ RSPN - RANDOM ANALYSIS DEMONSTRATION
AUGUST  31, 2004  Simcenter Nastran   8/31/04   PAGE     6




                                              R E A L   E I G E N V A L U E
S
  MODE    EXTRACTION      EIGENVALUE            RADIANS
CYCLES            GENERALIZED         GENERALIZED
   NO.
ORDER
 MASS               STIFFNESS
       1
1         .0                  .0                  .0
1.000000E+00          .0
       2          2        3.188542E+06        1.785649E+03        2.841949E
+02        1.000000E+00        3.188542E+06
       3          3        1.568106E+08        1.252240E+04        1.993002E
+03        1.000000E+00        1.568106E+08




   FREQ RSPN - RANDOM ANALYSIS
DEMONSTRATION                                 AUGUST  31, 2004  Simcenter
Nastran   8/31/04   PAGE    53



                           X Y - O U T P U T   S U M M A R Y   ( A U T O
O R   P S D F )
```

```
 PLOT  CURVE FRAME                          RMS        NO. POSITIVE    XMIN
FOR    XMAX FOR    YMIN FOR     X FOR      YMAX FOR     X FOR*
 TYPE    TYPE    NO.     CURVE ID.          VALUE         CROSSINGS    ALL
DATA    ALL DATA    ALL DATA      YMIN      ALL DATA      YMAX
 PSDF DISP        5         2(  4)     4.427970E-01  2.823553E+02  2.000E+02
6.000E+02  2.931E-09  5.000E+02  1.095E-02  2.800E+02
 PSDF DISP        6         3(  4)     6.487856E-01  2.843350E+02  2.000E+02
6.000E+02  2.025E-06  6.000E+02  2.297E-02  2.800E+02
 PSDF ACCE        7         2(  4)     1.400392E+06  2.856203E+02  2.000E+02
6.000E+02  2.855E+05  5.000E+02  1.049E+11  2.800E+02
 PSDF ACCE        8         3(  4)     2.091627E+06  2.934378E+02  2.000E+02
6.000E+02  1.997E+08  2.000E+02  2.201E+11  2.800E+02
 PSDF EL FOR      9        12(  2)     2.375014E+06  3.161015E+02  2.000E+02
6.000E+02  5.009E+08  2.200E+02  2.306E+11  2.800E+02
 PSDF EL FOR     10        23(  2)     2.091485E+06  2.934356E+02  2.000E+02
6.000E+02  1.997E+08  2.000E+02  2.201E+11  2.800E+02
 AUTO VELO       11         2(  4)     7.854708E+02  2.837833E+02  0.000E+00
1.000E-01 -5.550E+05  2.000E-03  6.170E+05  0.000E+00
 AUTO VELO       12         3(  4)     1.158947E+03  2.872666E+02  0.000E+00
1.000E-01 -1.179E+06  2.000E-03  1.343E+06  0.000E+00
   FREQ RSPN - RANDOM ANALYSIS
DEMONSTRATION                                  AUGUST  31, 2004  Simcenter
Nastran   8/31/04   PAGE    54
```

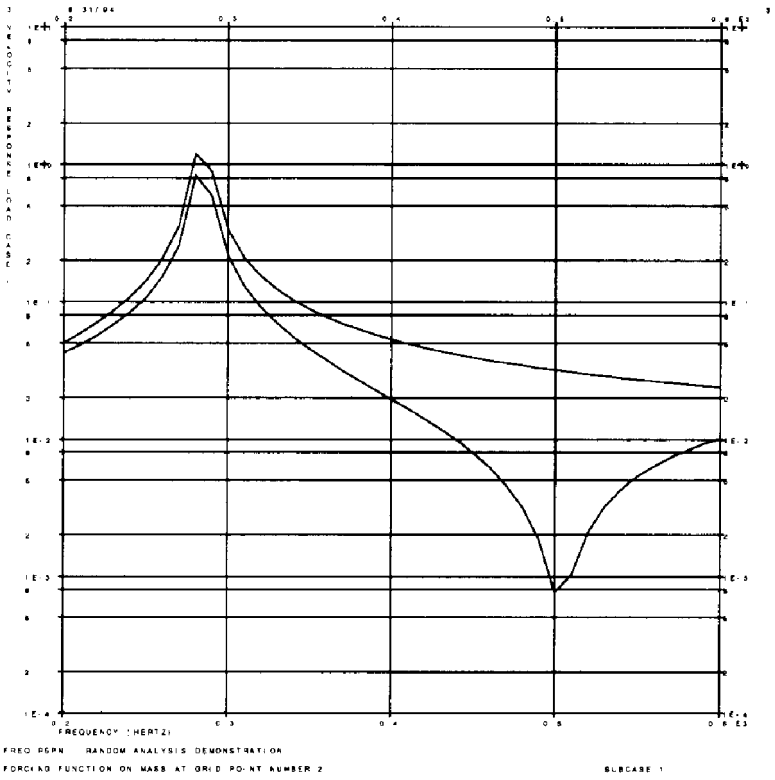



**Figure 6-10. Velocity Response, Points 2 and 3 (Subcase 1)**

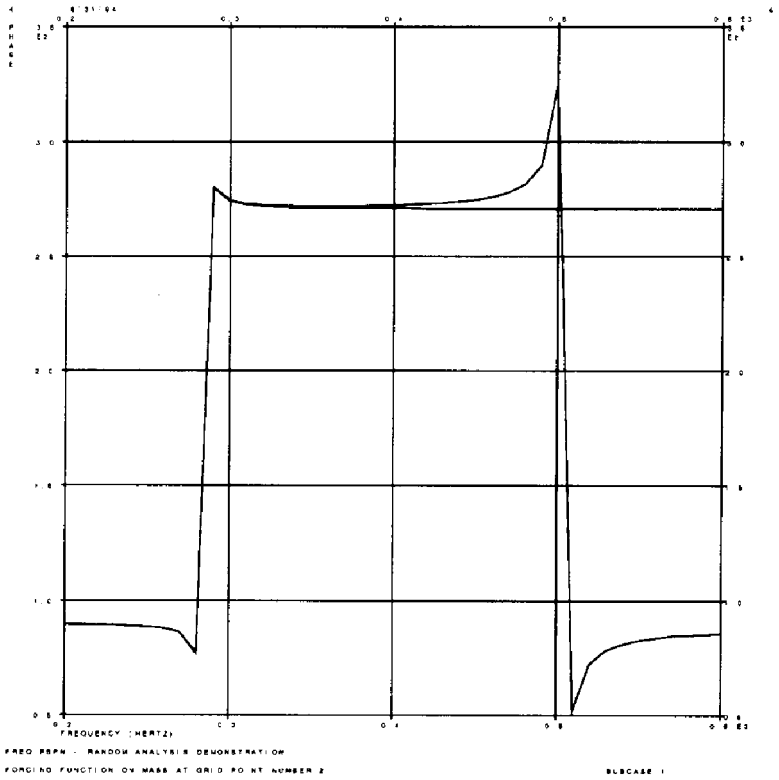



**Figure 6-11. Velocity Phase Angle, Points 2 and 3 (Subcase 1)**

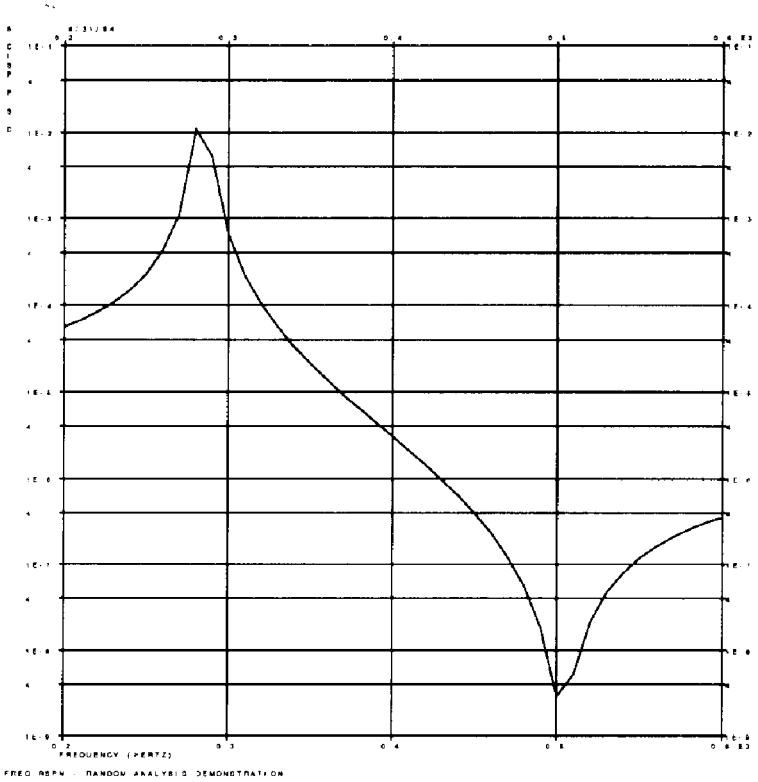


**Figure 6-12. Power Spectral Density, Point 2 Displacement**

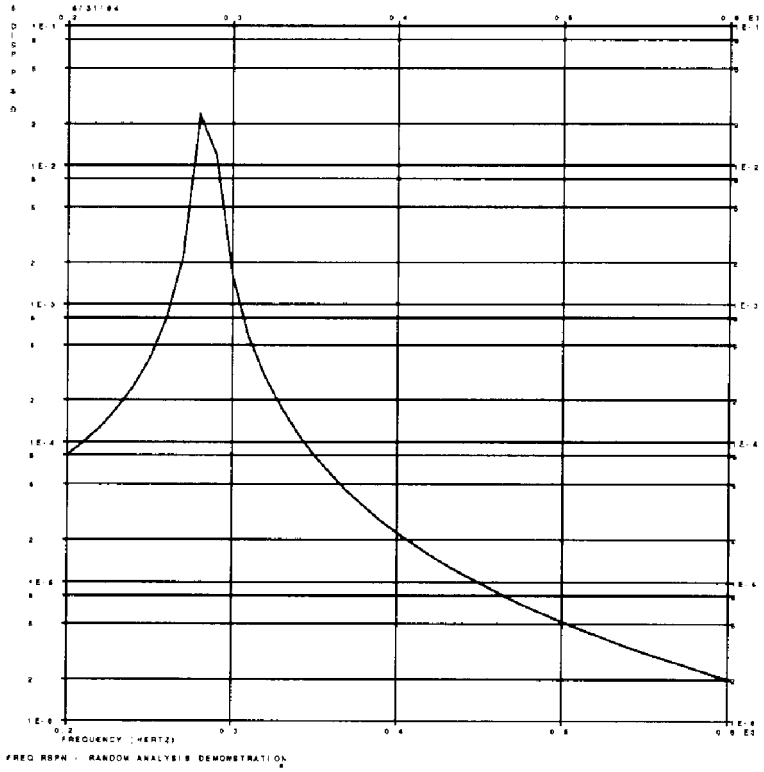


**Figure 6-13. Power Spectral Density, Point 3 Displacement**

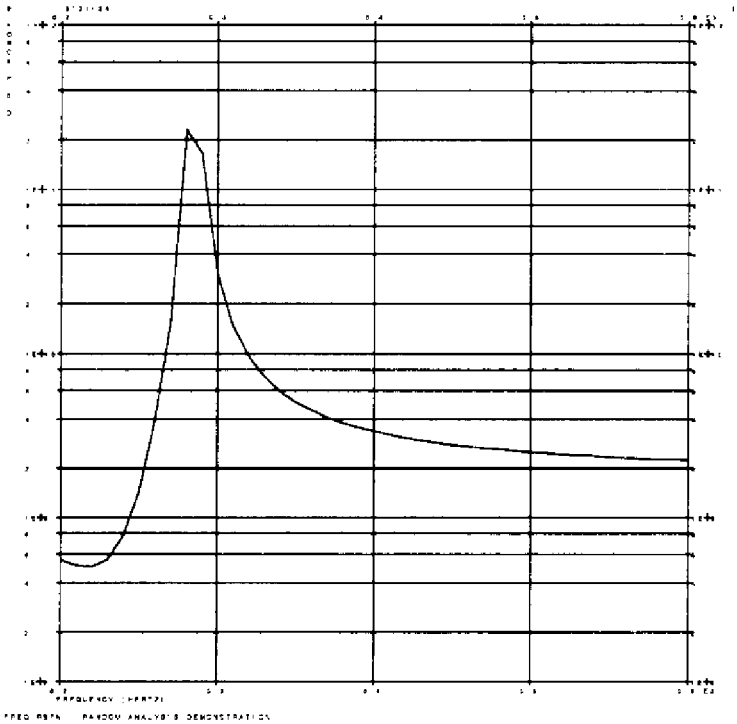


**Figure 6-14. PSDF Forces in Element 12**

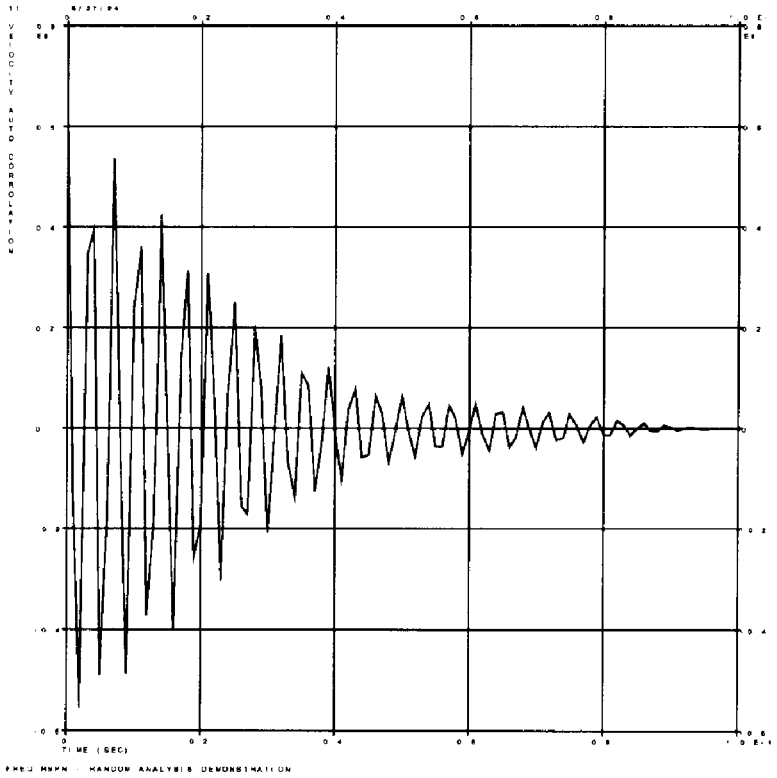


**Figure 6-15. Autocorrelation, Velocity of Point 2**

# 7. Multi-body Dynamics and Control System Software Interfaces

## 7.1 Overview of Multi-body Dynamics and Control System Software Interfaces

Simcenter Nastran can:

- Export modal data to Simcenter Motion, RecurDyn, Adams, and SIMPACK multi-body dynamics software.
- Export standard or state-space data to control system software.
- Export standard or state-space data to a MATLAB script file.
- Recover results directly from RecurDyn and Adams multi-body dynamics simulations.

## 7.2 Export

You can export a Simcenter Motion input file (OP2), RecurDyn flex input file (RFI), an Adams modal neutral file (MNF), or a SIMPACK flexible body input file (FBI) from a Simcenter Nastran modal solution. These files contain reduced order matrices from the modal solution results. The reduced order matrices for RFI, MNF, and FBI files can be created from a SOL 103, 111, or 112 solve. The reduced order matrices for an OP2 file can be created from a SOL 103 solve only.

You can then import the OP2, RFI, MNF, or FBI files into Simcenter Motion, RecurDyn, Adams or SIMPACK, respectively, and use them to represent flexible components in a multi-body dynamics simulation. The OP2, RFI, MNF, and FBI export capability streamlines the process of creating flexible components for multi-body dynamics simulations from finite element models.

From a modal solution, Simcenter Nastran can also write a standard (second-order) or a state-space (first-order) representation of a flexible body suitable for use with control system software including MATLAB. During the modal solution, Simcenter Nastran reduces the mass, stiffness, and damping matrices in modal space to limit the size of the representation while still retaining accuracy. It then writes the representation to either an OUTPUT4 file that control system software can use or a script file for use with MATLAB.

The modal solution used to produce an OP2, RFI, MNF, or FBI requires special considerations. Because the flexible component will be attached to other components in the multi-body dynamics simulation, local flexibility effects at the connection locations are important.

Two types of modal reduction are supported.

### Normal Modes Reduction

A one-step modal solution of the system matrices is performed and used to reduce the modal mass, stiffness, and damping matrices. The solved modes provide a good representation of the global dynamics of the component, but the local flexibility effects at the connections are typically not captured because of modal truncation. To account for modal truncation, you can request additional residual vector modes for the input and output DOF.

### Component Mode Synthesis (a general form for Craig-Bampton reduction)

A two-step modal solution of the component matrices is performed and used to reduce the modal mass, stiffness, and damping matrices. This method is recommended over normal modes reduction because it directly accounts for both local and global flexibility effects. Using ASET bulk entries, you define possible connection and load DOF.

- Connection DOF are DOF that may get connected to DOF of other components in the multi-body dynamics analysis or control system simulation.
- Load DOF are DOF where loads may be applied in the multi-body dynamics analysis or control system simulation.

The first eigensolution computes:

- The constrained normal modes of the system, which are the modes with the connection DOF restrained.
- The constraint modes of the ASET DOF, which are the static modes solved by applying unit displacements at individual ASET DOF while restraining all other ASET DOF.

Simcenter Nastran then uses the constrained normal modes and the constraint modes to reduce the number of DOF in the model. Then, a second eigensolution is performed on the reduced system.

### Differential Stiffness

You can account for differential stiffness in an external multi-body dynamics or control system simulation. Differential stiffness is used to model static stiffening that results from bolt preload, gravity, or centrifugal force among others. To request that differential stiffness be written to an interface file like an RFI file, include a static subcase that contains the stress-stiffening loads. In another subcase include STATSUB = *n* where *n* is the number of the static subcase.

### Related Commands

- MBDEXPORT
  The MBDEXPORT case control command exports a Simcenter Motion input file (OP2), RecurDyn flex input file (RFI), an Adams modal neutral file (MNF), or a SIMPACK flexible body input file (FBI) from a Simcenter Nastran modal solution. The MBDEXPORT case control command can also generate

standard or state-space matrices from a Simcenter Nastran modal solution and write the matrices to either an OUTPUT4 file or a MATLAB script file for use with control system software.

- ADAMSMNF
  Prior to NX Nastran 6.1, the ADAMSMNF case control command was the only command used to request an Adams modal neutral file (MNF). Beginning with NX Nastran 6.1, the MBDEXPORT case control command can also be used to request an Adams modal neutral file (MNF). The ADAMSMNF case control command cannot be used to generate files for use with control system software.

- RECURDYNRFI (Obsolete command)
  Prior to NX Nastran 7.0, the RECURDYNRFI case control command could be used to request a RecurDyn flex input file (RFI). Starting with NX Nastran 7.0, RECURDYNRFI is obsolete. The MBDEXPORT case control command should be used to request a RecurDyn flex input file (RFI).

## 7.3 Results Recovery

Simcenter Nastran can directly recover results for a component of a RecurDyn or Adams multi-body dynamics simulation and use the recovered results to calculate displacements, stresses, strains, and forces among others for the component. The component results recovery run uses a component modal deformation file (MDF) along with a results recovery OUTPUT2 file. Simcenter Nastran can create the OUTPUT2 file during the component creation run or during the component results recovery run. The DMAP alter mnfx.alt can also create an OUTPUT2 file.

The MDF for a component analyzed using RecurDyn or Adams is written directly by the RecurDyn or Adams software. However, the Simcenter Motion and SIMPACK software cannot write a MDF. As a work around, you can output and reformat displacement, velocity, and acceleration results from a Simcenter Motion or SIMPACK run into the MDF format. This format is described under **Modal Deformation File (MDF) Format**.

### Related Commands

- MBDEXPORT
  The RECVROP2=YES option of the MBDEXPORT case control command is used during the component creation run to create an OUTPUT2 file for results recovery from either a RecurDyn or Adams multi-body dynamics simulation.

- ADAMSMNF
  Prior to NX Nastran 6.1, the ADAMSMNF case control command with ADMOUT=YES was the only command to request an OUTPUT2 file for results recovery from an Adams flexible body during the component creation run. Beginning with NX Nastran 6.1, the MBDEXPORT case control command can also be used to request an OUTPUT2 file for results recovery from an Adams flexible body during the component creation run.

- MBDRECVR
  The MBDRECVR case control command is used during the component results recovery run for results recovery from a modal deformation file (MDF) that contains the results of a multi-body dynamics simulation.

- ADMRECVR
  Prior to NX Nastran 6.1, the ADMRECVR case control command was the only command to recovery results for an Adams flexible body during the component results recovery run. Beginning with NX Nastran 6.1, the MBDRECVR case control command can also be used to recover results from an Adams flexible body during the component results recovery run.

## 7.4 Simcenter Motion, RecurDyn, Adams, and SIMPACK Flexible Body Export Procedure

The input file used to create a Simcenter Motion input file (OP2), a RecurDyn flex input file (RFI), an Adams modal neutral file (MNF), or a SIMPACK flexible body input file (FBI) is similar to that for a SOL 103 modal solution except that:

- The MBDEXPORT case control command is used with the SCMOTION, RECURDYN, ADAMS, or SIMPACK describer specified to define the flexible body file type.

- The DTI,UNITS bulk entry is used to specify the system of units for the data written to the RFI, MNF, or FBI file. For Simcenter Motion, because the entire analysis occurs within Pre/Post, the DTI,UNITS bulk entry is not required.

**Example: Use of Normal Modes Reduction to Write an RFI File**

In this example, an RFI file is created using modal results from a normal modes reduction. Because normal modes reduction is used, the input file requests additional residual vector modes to avoid modal truncation.

```
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a RecurDyn flex input file (RFI) from a normal
$ modes reduction
$
MBDEXPORT RECURDYN FLEXBODY=YES
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
```

```
$
$ Use DTI,UNITS to specify the unit system used in the RFI file
$
DTI,UNITS,KG,N,M,S
$
$ Use PARAM,RESVEC to request additional residual vector modes
$
PARAM,RESVEC,YES
$
.....
.....
$
ENDDATA
```

If you select component mode synthesis as the modal reduction method, the input file must also include:

- An RSMETHOD case control command to select the eigenvalue options used for the component mode synthesis reduction.
- An ASET bulk entry to define the constraint DOF.
- An SPOINT bulk entry to define the component modal coordinates.
- A QSET bulk entry for the component modal coordinates.

**Example: Use of Component Mode Synthesis Reduction to Write an RFI File**

In this example, an RFI file is created using modal results from a component mode synthesis modal reduction.

```
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a RecurDyn flex input (RFI) file from a
component
$ mode synthesis reduction
$
MBDEXPORT RECURDYN FLEXBODY=YES
$
$ Use RSMETHOD to select the EIGRL options used for the component mode
$ synthesis reduction
$
RSMETHOD=100
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
```

```
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
$
$ Use ASET1 to define the constraint DOF
$
ASET1,123456,25,175
$
$ Use SPOINT to define the  component modal coordinates
$
SPOINT,10001,THRU,10009
$
$ Use QSET1 for the component modal coordinates
$
QSET1,0,10001,THRU,10009
$
$ Use EIGRL to define eigenvalue options for component mode synthesis
reduction
$
EIGRL,100,,,9
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
$
$ Use DTI,UNITS to specify the unit system used in the RFI file
$
DTI,UNITS,KG,N,M,S
$
.....
.....
$
ENDDATA
```

Note:

In both of the above examples, a Simcenter Motion input file (OP2), an Adams modal neutral file (MNF), or a SIMPACK flexible body input file (FBI) can be created by replacing the RECURDYN describer with the SCMOTION, ADAMS, or SIMPACK describer, respectively, in the MBDEXPORT case control command, or the ADAMSMNF case control command can be used instead of the MBDEXPORT case control command.

## 7.5 Control System Export Procedure

The input file used to create an OUTPUT4 file for use with control system software is similar to the input file for a SOL 103 modal solution. However, the following entries are also required:

- An MBDEXPORT case control command with the OP4=unit and STATESPACE/STANDARD describers specified to produce a state-space/standard representation written to an OUTPUT4 file.
- An ASSIGN statement in the file management section that specifies the filename of the OUTPUT4 file.

If a state-space representation is requested, the input file must additionally include:

- A USETi bulk entry with the U7 set specifying the input DOF of the system.
- A USETi bulk entry with the U8 set specifying the output DOF of the system.

You can optionally use the DIGITS parameter to control the numerical precision of the matrix data written to the OUTPUT4 file.

**Example: Use of Normal Modes Reduction to Export a State-Space Representation**

In this example, the state-space representation is generated using modal results from a normal modes reduction. DOF 1-6 for node 25 are the input DOF and DOF 1-6 for node 175 are the output DOF. Because normal modes reduction is used, the input file requests additional residual vector modes.

```
ASSIGN OUTPUT4='filename.op4' UNIT=22 FORM=FORMATTED
$
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a state-space representation written to an
$ OUTPUT4 file
$
MBDEXPORT OP4=22 STATESPACE FLEXBODY=YES
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
```

```
EIGRL,1,,,50
$
$ Use USET U7 to define the input DOF
$
USET,U7,25,123456
$
$ Use USET U8 to define the output DOF
$
USET,U8,175,123456
$
$ Use USET U6 and PARAM RESVEC to request residual vector modes at input
DOF
$
USET,U6,25,123456
$
PARAM,RESVEC,YES
$
.....
.....
$
ENDDATA
```

**Example: Use of Normal Modes Reduction to Export a Standard Representation**

This example is identical to the previous example except that a standard representation is exported.

```
ASSIGN OUTPUT4='filename.op4' UNIT=22 FORM=FORMATTED
$
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a standard representation written to an
$ OUTPUT4 file
$
MBDEXPORT OP4=22 STANDARD FLEXBODY=YES
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
```

```
$
$ Use USET U6 and PARAM RESVEC to request residual vector modes at input
DOF
$
USET,U6,25,123456
$
PARAM,RESVEC,YES
$
.....
.....
$
ENDDATA
```

If you select component mode synthesis as the modal reduction method, the input file must include:

- An MBDEXPORT case control command with the OP4=unit and STATESPACE/STANDARD describers specified to produce a state-space/standard representation written to an OUTPUT4 file.
- An ASSIGN statement in the file management section that specifies the filename of the OUTPUT4 file.
- An RSMETHOD case control command to select the eigenvalue options used for the component mode synthesis reduction.
- An ASET bulk entry to define the connection and/or input DOF.
- An SPOINT bulk entry to define the component modal coordinates.
- A QSET bulk entry for the component modal coordinates.

If a state-space representation is requested, the input file must additionally include:

- A USETi bulk entry with the U7 set specifying the input DOF of the system.
- A USETi bulk entry with the U8 set specifying the output DOF of the system.

**Example: Use of Component Mode Synthesis Reduction to Export a State-Space Representation**

In this example, the state-space representation is generated using modal results from a component mode synthesis modal reduction. DOF 1-6 of node 25 are the constraint DOF. DOF 1-6 for node 25 are the input DOF and DOF 1-6 for node 175 are the output DOF.

```
ASSIGN OUTPUT4='filename.op4' UNIT=22 FORM=FORMATTED
$
SOL 103
$
CEND
$
```

```
$ Use MBDEXPORT to request a state-space representation written to an
$ OUTPUT4 file
$
MBDEXPORT OP4=22 STATESPACE FLEXBODY=YES
$
$ Use RSMETHOD to select the EIGRL options used for the component mode
$ synthesis reduction
$
RSMETHOD=100
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use ASET1 to define the constraint DOF
$
ASET1,123456,25
$
$ Use SPOINT to define the component modal coordinates
$
SPOINT,10001,THRU,10009
$
$ Use QSET1 for the component modal coordinates
$
QSET1,0,10001,THRU,10009
$
$ Use EIGRL to define eigenvalue options for component mode synthesis
reduction
$
EIGRL,100,,,9
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
$
$ Use USET U7 to define the input DOF
$
USET,U7,25,123456
$
$ Use USET U8 to define the output DOF
$
USET,U8,175,123456
$
.....
```

```
.....
$
ENDDATA
```

**Example: Use of Component Mode Synthesis Reduction to Export a Standard Representation**

This example is identical to the previous example except that a standard representation is exported.

```
ASSIGN OUTPUT4='filename.op4' UNIT=22 FORM=FORMATTED
$
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a standard representation written to an
$ OUTPUT4 file
$
MBDEXPORT OP4=22 STANDARD FLEXBODY=YES
$
$ Use RSMETHOD to select the EIGRL options used for the component mode
$ synthesis reduction
$
RSMETHOD=100
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use ASET1 to define the constraint DOF
$
ASET1,123456,25
$
$ Use SPOINT to define the component modal coordinates
$
SPOINT,10001,THRU,10009
$
$ Use QSET1 for the component modal coordinates
$
QSET1,0,10001,THRU,10009
$
$ Use EIGRL to define eigenvalue options for component mode synthesis
reduction
$
EIGRL,100,,,9
```

```
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
$
.....
.....
$
ENDDATA
```

## 7.6 MATLAB Export Procedure

State-space (first-order) and standard (second-order) representations of the component can be written to a MATLAB script file. The input files used to create a MATLAB script file are identical to those used to produce an OUTPUT4 file except:

- The OP4=unit describer in the MBDEXPORT case control command is replaced with the MATLAB describer.
- The ASSIGN statement specifying the filename of the OUTPUT4 file is not applicable and should be omitted.

## 7.7 Simcenter Motion, RecurDyn, Adams, and SIMPACK Flexible Body Results Recovery Procedure

During the component creation run generating a Simcenter Motion input file (OP2), a RecurDyn flex input file (RFI), an Adams modal neutral file (MNF), or a SIMPACK flexible body input file (FBI), you can create an OUTPUT2 results recovery file by specifying the RECVROP2=YES option in the MBDEXPORT case control command. If the ADAMSMNF case control command is used to create an Adams modal neutral file (MNF), you can create an OUTPUT2 results recovery file by specifying the ADMOUT=YES option. In either case, you should include an ASSIGN statement in the file management section that specifies the filename of the OUTPUT2 file and UNIT=20.

**Example: Use of Normal Modes Reduction to Write an RFI File and an OUTPUT2 File**

In this example, a binary OUTPUT2 results recovery file named *filename1.out* and a RecurDyn flex input file (RFI) are created. Because normal modes reduction is used, the input file requests additional residual vector modes to avoid modal truncation.

```
ASSIGN OUTPUT2='filename1.out' UNIT=20
$
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a RecurDyn flex input (RFI) file and an
```

```
OUTPUT2
$ file from a normal modes reduction
$
MBDEXPORT RECURDYN FLEXBODY=YES RECVROP2=YES
$
$ Use METHOD to select the EIGRL options used for the normal modes
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
$
$ Use PARAM,RESVEC to request additional residual vector modes
$
PARAM,RESVEC,YES
$
$ Use DTI,UNITS to specify the unit system used in the RFI file
$
DTI,UNITS,KG,N,M,S
$
.....
.....
$
ENDDATA
```

**Example: Use of Component Mode Synthesis Reduction to Write an RFI File and an OUTPUT2 File**

In this example, a binary OUTPUT2 results recovery file named *filename1.out* and a RecurDyn flex input file (RFI) are created.

```
ASSIGN OUTPUT2='filename1.out' UNIT=20
$
SOL 103
$
CEND
$
$ Use MBDEXPORT to request a RecurDyn flex input (RFI) file and an
OUTPUT2
$ file from a component mode synthesis reduction
$
MBDEXPORT RECURDYN FLEXBODY=YES RECVROP2=YES
$
$ Use METHOD to select the EIGRL options used for the normal modes
```

```
solution
$
METHOD=1
$
DISPLACEMENT=ALL
$
BEGIN BULK
$
$ Use EIGRL to define eigenvalue options for the normal modes solution
$
EIGRL,1,,,50
$
$ Use ASET1 to define the connection DOF
$
ASET1,123456,25,175
$
$ Use SPOINT to define the component modal coordinates
$
SPOINT,10001,THRU,10009
$
$ Use QSET1 for the component modal coordinates
$
QSET1,0,10001,THRU,10009
$
$ Use DTI,UNITS to specify the unit system used in the RFI file
$
DTI,UNITS,KG,N,M,S
$
.....
.....
$
ENDDATA
```

The Simcenter Motion input file (OP2), RecurDyn flex input file (RFI), Adams modal neutral file (MNF), or SIMPACK flexible body input file (FBI) created during the component creation run is then imported into Simcenter Motion, RecurDyn, Adams, or SIMPACK, respectively, along with any other OP2, RFI, MNF, or FBI files representing other components of a multi-body system. From RecurDyn and Adams multi-body dynamics simulations, the dynamic response results are automatically written to a modal deformation file (MDF) in either binary (OUTPUT2) or ASCII (Punch) formats as specified by the user. From a Simcenter Motion or SIMPACK multi-body dynamics simulation, you must manually reformat the simulation results into the modal deformation file (MDF) format. (For additional information, see **Modal Deformation File (MDF) Format**.)

The component results recovery run requires a binary or ASCII formatted modal deformation file (MDF) and an OUTPUT2 results recovery file. If the OUTPUT2 file was created during the component creation run (or created using the DMAP alter mnfx.alt), you can use the MBDRECVR case control command or, if applicable, the ADMRECVR case control command to enter the needed recovery options. An ASSIGN statement should also be added to the file management section that specifies the filename of the OUTPUT2 file and UNIT=20.

### Example: Results Recovery Using an ASCII Modal Deformation File

In the previous two examples, a binary OUTPUT2 results recovery file named *filename1.out* was created along with an RFI file. The RFI file was subsequently used as an input for a RecurDyn multi-body dynamics simulation. Suppose that the multi-body dynamics simulation produced an ASCII modal deformation file named *filename2.mdf*. The following input file accesses the binary OUTPUT2 results recovery file (*filename1.out*) and the ASCII modal deformation file (*filename2.mdf*) for a component results recovery run.

```
ASSIGN INPUTT2='filename1.out' UNIT=20
$
SOL 103
$
CEND
$
$ Use MBDRECVR to import the the ASCII modal deformation file (MDF)
$ and the binary OUTPUT2 results recovery file
$
MBDRECVR ASCII
$
DISPLACEMENT=ALL
STRESS=ALL
STRAIN=ALL
FORCE=ALL
$
BEGIN BULK
$
$ Use INCLUDE to insert the ASCII modal deformation file (MDF) into
$ the input file
$
INCLUDE 'filename2.mdf'
$
.....
.....
$
ENDDATA
```

### Example: Results Recovery Using a Binary Modal Deformation File

Suppose the RecurDyn multi-body dynamics simulation produced a binary modal deformation file named *filename2.mdf*. The following input file accesses the binary OUTPUT2 results recovery file (*filename1.out*) and the binary modal deformation file (*filename2.mdf*) for a component results recovery run.

```
ASSIGN INPUTT2='filename1.out' UNIT=20
ASSIGN INPUTT2='filename2.mdf' UNIT=13
$
SOL 103
```

```
$
CEND
$
$ Use MBDRECVR to import the the binary modal deformation file (MDF)
$ and the binary OUTPUT2 results recovery file
$
MBDRECVR BINARY
$
DISPLACEMENT=ALL
STRESS=ALL
STRAIN=ALL
FORCE=ALL
$
BEGIN BULK
$
.....
.....
$
ENDDATA
```

> Note:
>
> In the above examples, you can use the ADMRECVR case control command instead of the MBDRECVR case control command if results are being recovered from Adams.

For efficiency and accuracy, it is strongly recommended that the OUTPUT2 results recovery file be created during the component creation run.

If the OUTPUT2 results recovery file was not created during the component creation run, you can create it by recomputing the SOL 103 results during the component results recovery run. To do so, you can use the MSRMODE=2 option in the MBDRECVR case control command or, if applicable, the ADMRECVR case control command. However, this approach is not recommended because:

- It is inefficient because modal results calculated during the component creation run must be recomputed during the component results recovery run.
- It is highly susceptible to errors in the results recovery due to slight differences in the modal results used to create either the Simcenter Motion input file (OP2), RecurDyn flex input file (RFI), the Adams modal neutral file (MNF), or the SIMPACK flexible body input file (FBI), and the modal deformation file (MDF).

## 7.8 Modal Deformation File (MDF) Format

Simcenter Nastran can use the displacement, velocity, and acceleration data from a multi-body dynamics software product to recover results back to the original Simcenter Nastran model. The Simcenter Nastran results recovery requires this data be stored in a modal deformation file (MDF).

The MDF is formatted to store time step data in direct table input format (DTI bulk entry) and modal displacement, velocity, acceleration output in direct matrix input format (DMI bulk entry).

The data order in the file is:

- Time step data stored with a DTI bulk entry named “TOL”.
- Modal displacement, velocity, acceleration output for the flexible modes stored with a DMI bulk entry named “UHT”.
- Modal displacement, velocity, acceleration output for the rigid body modes stored with a DMI bulk entry named “BHT”.

See the *Simcenter Nastran Quick Reference Guide* for a description of DTI and DMI bulk entries.

**Time Step Data**

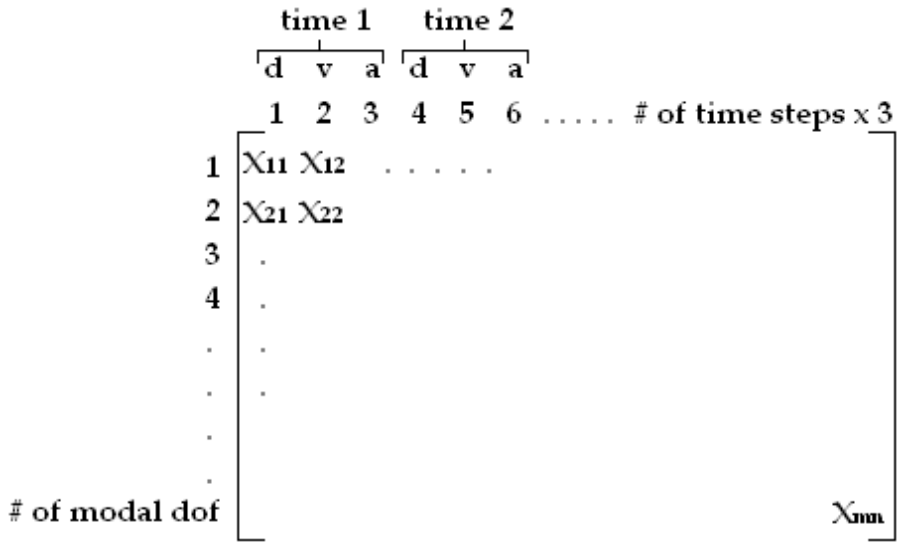
The fourth field in the header row of the DTI entry defines the number of time steps. The time steps are then listed beginning in the first continuation row.

In the example below, there are 101 time steps, the first is 0.00000000E+00 and the last is 5.00000000E-01.

```
DTI      TOL             0      101        1        2        1        1    10000
*          0.00000000E+00  5.00000000E-03  1.00000000E-02  1.50000000E-02
*          2.00000000E-02  2.50000000E-02  3.00000000E-02  3.50000000E-02
*          4.00000000E-02  4.50000000E-02  5.00000000E-02  5.50000000E-02
....
....
*          4.80000000E-01  4.85000000E-01  4.90000000E-01  4.95000000E-01
*          5.00000000E-01          ENDREC
```

**Modal Displacement, Velocity, Acceleration - Flexible Modes**

The DMI bulk entry stores the displacement (d), velocity (v), and acceleration (a) in the following matrix format:




The number of rows is the total number of modal degrees of freedom including rigid body modes, although the actual data for rigid body modes is written to a separate DMI entry described below. For example, if there are six rigid body modes, the output data for the flexible modes begins in row seven.

The matrix is designed to optionally store d, v, and a, so the number of columns is the number of time steps multiplied by three.

In the first row of the DMI entry, field 8 defines the number of rows and field 9 defines the number of columns. In the example below, the matrix has 38 rows and 303 columns corresponding to 38 modal dof, and 101 times steps multiplied by three.

Each column definition begins with a unique continuation line with the column number in field 3. The column numbers 1, 4, ... 303 are highlighted in the example below. In this same line, the integer value entered in field four is the row number in which the data begins. In the example below, the data begins at row seven because the first six rows are rigid body modes. Also in this example, only modal displacement values are included; the modal velocity and modal acceleration columns do not exist. Simcenter Nastran interprets missing data to be 0.0.

```
DMI      UHT             0       2       1       0              38     303
DMI*     UHT                             1                      7 0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00  0.00000000E+00
*          0.00000000E+00  0.00000000E+00  0.00000000E+00
DMI*     UHT                             4                      7-2.04199568E-04
*          3.39347076E-04  2.09681266E-04 -5.99340300E-05 -2.50379161E-05
*          1.02017980E-05 -3.35077415E-05 -5.83890993E-06 -2.19535986E-05
*          9.49341824E-07  3.01761250E-06 -1.11499227E-05 -3.62404862E-07
*         -6.94741724E-06 -9.92703344E-06 -8.23406835E-06 -3.58618463E-06
*          6.17187754E-06 -9.33733393E-07  2.38506635E-06  4.75702908E-06
*         -3.10199374E-06  4.39767782E-06  5.34601237E-06  6.51051287E-06
*          9.73742291E-06 -5.11626257E-06 -9.18547869E-06  2.72121539E-05
*          2.40024328E-07  4.63807967E-05  3.42981015E-06
....
....
DMI*     UHT                           301                      7-2.49201200E-04
*          4.57257434E-04 -7.55568163E-04  5.76127320E-06 -7.50645755E-05
*          1.39623033E-05  2.34905978E-05  8.40027111E-06 -3.30498177E-05
*          1.58961912E-06  1.89136267E-05 -2.04989270E-06  2.78021307E-06
*         -8.63598543E-07  1.23130850E-06 -1.46538932E-06 -1.04200900E-06
*          3.57640780E-06 -4.53643422E-06  9.83088935E-08  1.89997079E-06
*         -9.13273600E-07  1.36953505E-06  1.40960823E-06  2.76899638E-06
*          4.10463662E-06 -1.74377940E-06 -2.77913536E-06  9.98286312E-06
*          1.41199701E-07  1.63503410E-05  1.17215365E-06
```

### Modal Displacement, Velocity, Acceleration – Rigid Body Modes

The matrix format for the rigid body modes is similar to the flexible modes described above. The number of rows equals the number of rigid body modes, and the number of columns is the same as the flexible modes.

There are six rigid body modes in the example below.

```
DMI     BHT            0       2       1       0              6     303
DMI*    BHT                                  1              1 1.20121513E-09
*         1.04583009E-08 -8.39819680E-09  7.35259284E-11  4.30989982E-10
*         0.00000000E+00
DMI*    BHT                                  4              1 2.56806802E-03
*        -4.90348020E-02 -1.61892233E-01  2.49284051E-01  4.94846725E-03
*        -2.49801170E-01
....
....
DMI*    BHT                                301              1 2.24782635E-02
*         8.30637504E-03 -8.21837215E-01  1.53797890E-01  4.14410835E-02
*        -1.52663337E-01
```

## 7.9 Mathematical Description

> Note:
>
> Throughout Simcenter Nastran documentation, damping is represented symbolically with the letter *B*. However, in the following article, damping is denoted with the letter *D* to avoid confusion with the *b* matrix of a state-space representation.

Simcenter Nastran supports both the normal mode reduction method and component mode synthesis (a general form for Craig-Bampton reduction). Both reduction methods can produce a standard representation of the equations of motion and a state-space representation of the equations of motion.

**Standard Representation using Normal Modes Reduction**

The equation of motion for a system can be written as

$$M_{ff}\ddot{u}_f + D_{ff}\dot{u}_f + K_{ff}u_f = P_f$$

**Equation 7-1.**

where $M_{ff}$, $D_{ff}$, $K_{ff}$ are the mass, viscous damping, and stiffness matrices, respectively. $P_f$ is the load vector and $u_f$ is the displacement vector. The subscript *f* denotes the Simcenter Nastran free set of DOF (f-set), which is the set of DOF that are not restrained or dependent on other DOF. Thus, the subscript denotes size and form of the associated matrix and should not be interpreted as an index.

The modes of the undamped system can be found by solving the eigenvalue problem:

$$K_{ff}\Phi_{fh} = M_{ff}\Phi_{fh}\Omega_{hh}$$

**Equation 7-2.**

where $\Phi_{fh}$ is the matrix of mass normalized mode shapes with *h* representing the number of modes. $\Omega_{hh}$ is a diagonal matrix of the eigenvalues:

$$\Omega_{hh} = \begin{bmatrix} & & 0 \\ & \omega^2 & \\ 0 & & \end{bmatrix}$$

where $\omega$ are the natural frequencies.

The response in modal space $\gamma_h$ is found from:

$$u_f = \Phi_{fh}\gamma_h$$

**Equation 7-3.**

Substituting **Equation 3** into **Equation 1** and premultiplying by $\Phi^T_{fh}$, we obtain:

$$m_{hh}\ddot{\gamma}_h + d_{hh}\dot{\gamma}_h + k_{hh}\gamma_h = \Phi^T_{fh}P_f$$

**Equation 7-4.**

where:

$$m_{hh} = \Phi^T_{fh}M_{ff}\Phi_{fh}$$
$$d_{hh} = \Phi^T_{fh}D_{ff}\Phi_{fh}$$
$$k_{hh} = \Phi^T_{fh}K_{ff}\Phi_{fh}$$

are the reduced mass, damping, and stiffness matrices, respectively. Similarly, premultiplying **Equation 2** by $\Phi^T_{fh}$ and rearranging yields:

$$\Omega_{hh} = m^{-1}_{hh}k_{hh}$$

**State-Space Representation using Normal Modes Reduction**

For the state-space representation used in a control system simulation, the output DOF are a subset of the f-set DOF. The output DOF are designated as the j-set for the following calculations. It is possible to use a partition matrix $S_{jf}$ to reduce the response from the full f-set to the smaller j-set as follows:

$$y_j = S_{jf} u_f$$

$$\dot{y}_j = S_{jf} \dot{u}_f$$

$$\ddot{y}_j = S_{jf} \ddot{u}_f$$

**Equation 7-5.**

In its simplest form, the matrix $S_{jf}$ is a partitioning matrix of 0s and 1s. If some of the output DOF are dependent (that is, in the Simcenter Nastran m-set), the form is slightly more complicated. Similarly, there is a subset of the f-set DOF that is the force input DOF. These DOF are designated here as the i-set. A partition matrix $R_{if}$ relates the f-set $P_f$ force vector to the i-set input $P_i$ force vector as follows:

$$P_f = R_{if}^T P_i$$

**Equation 7-6.**

To obtain a state-space representation, begin by rearranging **Equation 4** and inserting **Equation 6** to obtain:

$$\ddot{\gamma}_h = m_{hh}^{-1} \Phi_{fh}^T R_{if}^T P_i - m_{hh}^{-1} d_{hh} \dot{\gamma}_h - m_{hh}^{-1} k_{hh} \gamma_h$$

**Equation 7-7.**

For simplicity, assume that the modal damping matrix $d_{hh}$ is given by:

$$d_{hh} = 2 m_{hh} \omega_{hh} Z_{hh}$$

where $Z_{hh}$ is a diagonal matrix of all the modal damping ratios and $\omega_{hh}$ is given by:

$$\omega_{hh} = \begin{bmatrix} \ddots & & 0 \\ & \omega & \\ 0 & & \ddots \end{bmatrix}$$

where $\omega$ are the natural frequencies. Because $m_{hh}$ is also a diagonal matrix, $d_{hh}$ is a diagonal matrix.

Now convert the above system of $h$ second order differential equations to a system of $2h$ first order differential equations by creating a state vector $\xi_{2h}$ which is related to $\gamma_h$ as follows:

$$\xi_{2h} = \begin{Bmatrix} \dot{\gamma}_h \\ \gamma_h \end{Bmatrix} \qquad \dot{\xi}_{2h} = \begin{Bmatrix} \ddot{\gamma}_h \\ \dot{\gamma}_h \end{Bmatrix}$$

**Equation 7-8.**

Using **Equation 8**, **Equation 7** can be rewritten as:

$$\dot{\xi}_{2h} = a_{2h\times 2h}\xi_{2h} + b_{2h\times i}P_i$$

where:

$$a_{2h\times 2h} = \begin{bmatrix} -m_{hh}^{-1}d_{hh} & -m_{hh}^{-1}k_{hh} \\ I_{hh} & 0_{hh} \end{bmatrix} \qquad b_{2h\times i} = \begin{bmatrix} m_{hh}^{-1}\Phi_{fh}^{T}R_{if}^{T} \\ 0_{hi} \end{bmatrix}$$

Here the submatrix $I_{hh}$ is the identity matrix and the submatrices $0_{hh}$ and $0_{hi}$ are null matrices. The state-space output vector $Y_{2j}$ is computed by substituting **Equation 3** into **Equation 5**. Comparing the resulting expression to **Equation 8**, one can write:

$$Y_{3j} = \begin{Bmatrix} \ddot{y}_j \\ \dot{y}_j \\ y_j \end{Bmatrix} = c_{3j\times 2h}\xi_{2h} + e_{3j\times i}P_i$$

where:

$$c_{3j\times 2h} = \begin{bmatrix} -S_{jf}\Phi_{fh}m_{hh}^{-1}d_{hh} & -S_{jf}\Phi_{fh}m_{hh}^{-1}k_{hh} \\ S_{jf}\Phi_{fh} & 0_{jh} \\ 0_{jh} & S_{jf}\Phi_{fh} \end{bmatrix} \qquad e_{3j\times i} = \begin{bmatrix} S_{jf}\Phi_{fh}m_{hh}^{-1}\Phi_{fh}^{T}R_{if}^{T} \\ 0_{jh} \\ 0_{jh} \end{bmatrix}$$

and the submatrix $0_{jh}$ is a null matrix.

**Standard Representation using Component Mode Synthesis (a general form for Craig-Bampton reduction)**

Component mode synthesis reduction is the preferred method to use when multiple connected flexible bodies are modeled. In such a case, it is important to account for static stiffness at the connections in order to obtain the correct local stiffness.

To begin, partition **Equation 1** into an a-set and an o-set. The a-set contains the connection DOF and the o-set contains all other DOF. After partitioning, **Equation 1** takes the following form:

$$\begin{bmatrix} M_{oo} & M_{oa} \\ M_{ao} & M_{aa} \end{bmatrix} \begin{Bmatrix} \ddot{u}_o \\ \ddot{u}_a \end{Bmatrix} + \begin{bmatrix} D_{oo} & D_{oa} \\ D_{ao} & D_{aa} \end{bmatrix} \begin{Bmatrix} \dot{u}_o \\ \dot{u}_a \end{Bmatrix} + \begin{bmatrix} K_{oo} & K_{oa} \\ K_{ao} & K_{aa} \end{bmatrix} \begin{Bmatrix} u_o \\ u_a \end{Bmatrix} = P_f$$

**Equation 7-9.**

The constrained normal modes $\Psi_{oq}$ are found by solving:

$$K_{oo}\Psi_{oq} = M_{oo}\Psi_{oq}\Omega_{qq}$$

where the subscript $q$ is the number of constrained normal modes.

The constraint modes are found from:

$$G_{oa} = -K_{oo}^{-1}K_{oa}$$

The constrained normal modes and the constraint modes are used together in component mode synthesis reduction to relate the response $u_f$ to a reduced response $u_z$ from a combination of physical response $u_a$ and generalized response $\eta_q$ as follows:

$$u_f = \begin{Bmatrix} u_o \\ u_a \end{Bmatrix} = \begin{bmatrix} \Psi_{oq} & G_{oa} \\ 0_{aq} & I_{aa} \end{bmatrix} \begin{Bmatrix} \eta_q \\ u_a \end{Bmatrix} = T_{fz} \begin{Bmatrix} \eta_q \\ u_a \end{Bmatrix} = T_{fz}u_z$$

**Equation 7-10.**

where the z-set is the sum of the a-set and the q-set, and $T_{fz}$ is the Craig-Bampton transformation matrix.

Substituting **Equation 10** into **Equation 9** and premultiplying by $T^T_{fz}$ we get the Craig-Bampton reduced equation of motion:

$$M_{zz}\ddot{u}_z + D_{zz}\dot{u}_z + K_{zz}u_z = T_{fz}^T P_f$$

**Equation 7-11.**

where:

$$M_{zz} = T_{fz}^T M_{ff} T_{fz}$$
$$D_{zz} = T_{fz}^T D_{ff} T_{fz}$$
$$K_{zz} = T_{fz}^T K_{ff} T_{fz}$$

The modes of the system are found by solving the following eigenvalue problem:

$$K_{zz}\Phi_{zh} = M_{zz}\Phi_{zh}\Omega_{hh}$$

where $\Phi_{zh}$ is the matrix of eigenvectors for the reduced system, $\Omega_{hh}$ is the diagonal matrix of natural frequencies squared for the reduced system, and $h$ is the number of modes. When solving this eigenvalue problem, it is important that modal truncation does not occur, which means that the dimension of $h$ and $z$ should be the same. If modal truncation occurs, static residual stiffness effects from the component mode synthesis reduction will be lost.

Now the physical response $u_z$ is transformed to modal response $\gamma_h$ using:

$$u_z = \Phi_{zh}\gamma_h$$

**Equation 7-12.**

Substituting **Equation 12** into **Equation 11** and premultiplying by $\Phi^T_{zh}$ gives:

$$m_{hh}\ddot{\gamma}_h + d_{hh}\dot{\gamma}_h + k_{hh}\gamma_h = \Phi^T_{zh}T^T_{fz}P_f$$

**Equation 7-13.**

where:

$$m_{hh} = \Phi^T_{zh}M_{zz}\Phi_{zh}$$
$$d_{hh} = \Phi^T_{zh}D_{zz}\Phi_{zh}$$
$$k_{hh} = \Phi^T_{zh}K_{zz}\Phi_{zh}$$

are the reduced mass, damping, and stiffness matrices, respectively. The matrix of eigenvalues is given by:

$$\Omega_{hh} = m^{-1}_{hh}k_{hh}$$

**State-Space Representation using Component Mode Synthesis (a general form for Craig-Bampton reduction)**

To obtain a state-space representation, rearrange **Equation 13** and insert **Equation 6** to yield:

$$\ddot{\gamma}_h = m_{hh}^{-1}\Phi_{zh}^T T_{fz}^T R_{if}^T P_i - m_{hh}^{-1} d_{hh}\dot{\gamma}_h - m_{hh}^{-1} k_{hh}\gamma_h$$

**Equation 7-14.**

**Equation 14** can be put into state-space format using **Equation 8**. As before, the resulting state vector is given by:

$$\dot{\xi}_{2h} = a_{2h\times 2h}\xi_{2h} + b_{2h\times i}P_i$$

where:

$$a_{2h\times 2h} = \begin{bmatrix} -m_{hh}^{-1}d_{hh} & -m_{hh}^{-1}k_{hh} \\ I_{hh} & 0_{hh} \end{bmatrix} \qquad b_{2h\times i} = \begin{bmatrix} m_{hh}^{-1}\Phi_{zh}^T T_{fz}^T R_{if}^T \\ 0_{hi} \end{bmatrix}$$

The output state-space vector $Y_{2j}$ is related to the state vector $\xi_{2h}$ by:

$$Y_{3j} = \begin{Bmatrix} \ddot{y}_j \\ \dot{y}_j \\ y_j \end{Bmatrix} = c_{3j\times 2h}\xi_{2h} + e_{3j\times i}P_i$$

where:

$$c_{3j\times 2h} = \begin{bmatrix} -S_{jf}T_{fz}\Phi_{zh}m_{hh}^{-1}d_{hh} & -S_{jf}T_{fz}\Phi_{zh}m_{hh}^{-1}k_{hh} \\ S_{jf}T_{fz}\Phi_{zh} & 0_{jh} \\ 0_{jh} & S_{jf}T_{fz}\Phi_{zh} \end{bmatrix}$$

and:

$$e_{3j\times i} = \begin{bmatrix} S_{jf}T_{fz}\Phi_{zh}m_{hh}^{-1}\Phi_{zh}^T T_{fz}^T R_{if}^T \\ 0_{jh} \\ 0_{jh} \end{bmatrix}$$

### Partitioning Matrices

In the most general case, the input and output DOF for a control system can be either a dependent or an independent DOF. For example, the DOF on a center node of an RBE3 spoke are dependent and can be

used as an output DOF to get an average output of points connected by the spoke. Similarly, the same node can be used as an input DOF to distribute a load over an area.

The full model DOF size (g-set) can be partitioned for the output DOF (j-set) as follows:

$$y_j = \overline{S}_{jg} u_g$$

**Equation 7-15.**

The matrix $\overline{S}_{jg}$ is a partitioning matrix populated with 0s and 1s. Further partitioning **Equation 15** into dependent DOF (m-set) and independent DOF (n-set) gives:

$$y_j = \begin{bmatrix} \overline{S}_{jm} & \overline{S}_{jn} \end{bmatrix} \begin{Bmatrix} u_m \\ u_n \end{Bmatrix}$$

**Equation 7-16.**

The transformation $G_{mn}$ relates the dependent DOF (m-set) to the independent DOF (n-set) as follows:

$$u_m = G_{mn} u_n$$

Utilizing this transformation in **Equation 16** yields:

$$y_j = \left[ \overline{S}_{jm} G_{mn} + \overline{S}_{jn} \right] u_n = S_{jn} u_n$$

**Equation 7-17.**

Partitioning the n-set of **Equation 17** into the free DOF (f-set) and restrained DOF (s-set) gives:

$$y_j = \begin{bmatrix} S_{js} & S_{jf} \end{bmatrix} \begin{Bmatrix} u_s \\ u_f \end{Bmatrix}$$

**Equation 7-18.**

Typically, restrained DOF have zero displacement. Thus, a restrained DOF cannot be an output DOF and **Equation 18** simplifies to:

$$y_j = S_{jf} u_f$$

which is identical to **Equation 5** when:

$$S_{jf} = \quad f\text{–}set \ \ partition \ of \ \left[\overline{S}_{jm} G_{mn} + \overline{S}_{jn}\right]$$

If output DOF are only in the f-set, the partitioning matrix consists of 0s and 1s. If an output DOF is on a dependent DOF (m-set), the $G_{mn}$ transformation matrix is used in the calculation and the partitioning factors may assume values other than 0 and 1.

The full model DOF size (g-set) can be partitioned to the force input DOF (i-set) using:

$$P_g = \overline{R}_{ig}^T P_i$$

**Equation 7-19.**

**Equation 19** can then be partitioned into dependent DOF (m-set) and independent DOF (n-set) as follows:

$$\begin{Bmatrix} P_m \\ \overline{P}_n \end{Bmatrix} = \begin{bmatrix} \overline{R}_{im} & \overline{R}_{in} \end{bmatrix}^T P_i = \begin{Bmatrix} \overline{R}_{im}^T P_i \\ \overline{R}_{in}^T P_i \end{Bmatrix}$$

**Equation 7-20.**

where bar overscripts indicate that the matrix was a vector prior to transformation. By transforming **Equation 20**, it can be shown that the dependent and independent forces are related as follows:

$$P_n = \overline{P}_n + G_{mn}^T P_m$$

**Equation 7-21.**

Substituting vectors $\overline{P}_n$ and $P_m$ from **Equation 20** into **Equation 21** gives:

$$P_n = \left[\overline{R}_{in}^T + G_{mn}^T \overline{R}_{im}^T\right] P_i = R_{in}^T P_i$$

**Equation 7-22.**

Partitioning the n-set of **Equation 22** into the free DOF (f-set) and restrained DOF (s-set) gives:

$$P_n = \begin{bmatrix} P_s & P_f \end{bmatrix} = \begin{bmatrix} R_{is}^T & R_{if}^T \end{bmatrix} P_i$$

**Equation 7-23.**

Because input forces are not specified on restrained DOF, a restrained DOF cannot be an input force DOF. Thus, **Equation 23** simplifies to:

$$P_f = R_{if}^T P_i$$

which is identical to **Equation 6** when:

$$R_{if}^T = \quad f\text{–}set \ \ partition \ of \ \left[ \overline{R}_{in}^T + G_{mn}^T \overline{R}_{im}^T \right]$$

If the input DOF are only in the f-set, the partitioning matrix consists of 0s and 1s. If an input DOF is on a dependent DOF (m-set), the $G_{mn}$ transformation matrix is used in the calculation and the partitioning factors may assume values other than 0 and 1.

### Damping

Multi-body dynamics and control system software uses viscous damping only. As such, Simcenter Nastran must convert structural (hysteretic) damping to viscous damping for use with multi-body dynamics and control system software.

In Simcenter Nastran, the overall viscous damping matrix in physical space, $D_{ff}$, results from summing:

- Viscous damping from CDAMPi elements, among others.
- User-defined viscous damping from DMIG entry.
- Structural damping converted to viscous damping using the W3 and W4 parameters. The sources of this structural damping includes:
  - Structural damping from GE on MATi bulk entries and PARAM,G.
  - User-defined structural damping from DMIG entry.

After Simcenter Nastran converts $D_{ff}$ to modal space to yield the viscous modal damping matrix $d_{hh}$, other viscous damping contributions are added into $d_{hh}$ including:

- User-defined viscous modal damping from SDAMPING.
- Structural damping converted to viscous modal damping using the WMODAL parameter. The sources of this structural damping include:

- Structural damping from GE on MATi bulk entries and PARAM,G.
- User-defined structural damping from DMIG entry.

By default, Simcenter Nastran writes the full viscous modal damping matrix $d_{hh}$ to standard and state-space MATLAB files, standard and state-space OP4 files, and Adams MNF files. This capability is not applicable to Simcenter Motion OP2 files, RecurDyn RFI files, and SIMPACK FBI files because they do not support damping matrices.

You can optionally specify that Simcenter Nastran write only the diagonal terms of the viscous modal damping matrix to the interface file by either:

- Including the NONCUP = -2 describer with the ADAMSMNF or MBDEXPORT case control command specification.
- Including PARAM,NONCUP,-2 in the bulk data section of the input file.

If you specify both the NONCUP describer and the NONCUP parameter, the NONCUP describer specification takes precedence.

For proportionally damped systems, the modal damping matrix is always diagonal. Thus, for proportionally damped systems, writing only the diagonal terms of the viscous modal damping matrix to the interface file is perfectly representative. However, if the system is not proportionately damped, you should utilize the default behavior and write all the viscous modal damping matrix terms to the interface file to avoid possible inaccuracies in the analysis results.

# A. Theory of Eigenvalue Extraction Methods

## A.1 Methods of Computation – Sturm Sequence

A numerical technique which is very helpful in a modal extraction is Sturm Sequence. In this approach, a frequency is selected and computations are performed to determine the number of roots which exist below that frequency. In some cases, you manually input this frequency, other times Simcenter Nastran automatically selects this frequency. This choice depends upon which modal extraction technique you're using.

Starting with the basic characteristic equation

$$[K + \lambda_s M] = 0$$

**Equation A-1.**

a frequency, $\omega^2 = \lambda_s$ is substituted. This frequency is called the shift point. The determinant $K - \lambda_s M$ is evaluated for the frequency $\lambda_s$. The resulting matrix is factored into it $[L][D][_L]^T$ components. The diagonal matrix, $[D]$, is known as the Factor Diagonal Matrix. The number of negative terms in the Factor Diagonal Matrix, known as the Sturm Number, are the number of frequencies that exist below the shift point frequency, $\lambda_s$.

Sturm Sequence logic is a very important part of the many of the modal extraction and dynamic reduction techniques. When it is utilized, the Sturm Sequence logic issues a number of messages which indicate the various shifts within the frequency range of interest. These messages help the various techniques to determine how many roots are in specific ranges and also if all modes have been found in the range.

## A.2 Eigensolution Methods

Simcenter Nastran provides seven methods of real eigenvalue extraction. These methods are numerical approaches to solving the equation of motion for its natural frequencies and modes shapes. No one method is the best for all problems. While most of the methods can be applied to all problems, the difference between the methods many times reduces to the efficiency of the solution process.

Most methods of algebraic eigenvalue extraction belong to one of two groups:

- transformation methods
- tracking methods

In a transformation method, the eigenvalue equation is first transformed into a special form (such as tridiagonal) from which eigenvalues may easily be extracted. In a "tracking" method, the roots are extracted from the original dynamic matrix one at a time using an iterative procedure.

Two of the real eigenvalue extraction methods available in Simcenter Nastran are transformation methods. These transformation methods use a "tridiagonal" solution method:

- Householder Method
- Modified Householder Method

The final real eigenvalue extraction method is the Lanczos. The Lanczos method combines the best characteristics of the tracking and transformation methods.

The Inverse Power Method available in Simcenter Nastran is classified as a tracking method.

## The Standard Tridiagonal Method (HOU)

The Householder (HOU) modal extraction method requires a positive definite mass matrix. There are no restrictions on the stiffness matrix other than that it must be symmetric. These matrices will always result in real (positive) roots. The tridiagonal methods are the most efficient methods for small problems and problems with dense matrices when a large portion of the eigenvectors are needed. These methods find all of the eigenvalues in the structure as a result of the transformation process. The user must make a specific request for the required eigenvectors (mode shapes). The tridiagonal methods do not take advantage of the sparse matrices, but they are efficient with the dense matrices sometimes created using dynamic reduction. The steps used in the tridiagonal methods are as follows:

1. Perform a Cholesky decomposition of the mass matrix:

$$[M] = [L][L]^T$$

**Equation A-2.**

where [*L*] is a lower triangular matrix.

2. The symmetric eigenvalue equation can be written as:

$$[J - \lambda I]\{w\} = 0$$

**Equation A-3.**

where [*J*] is the symmetric matrix and [*I*] is the identity matrix. To do this, premultiply the equation in **Methods of Computation – Sturm Sequence** by $[L]^{-1}$ and substitute for [*M*] from **Eq. A-2** to set the following:

$$[L]^{-1}[K]\{u\} - \lambda[L]^{-1}[L][L]^{T}\{u\} = 0$$

**Equation A-4.**

Then make the transformation:

$$\{w\} = [L]\{u\}$$

**Equation A-5.**

which reduces **Eq. A-4** to **Eq. A-3** using the following:

$$[J] = [L]^{-1}[K][L]^{-1,T}$$

**Equation A-6.**

3. The [*J*] matrix is converted to a tridiagonal matrix by a transformation method according to the Householder method. (A tridiagonal matrix is a matrix where the only nonzero terms in the i-th column are the i – 1, i, i + 1 row.)

4. All the eigenvalues of the tridiagonal matrix are extracted using a modified QR algorithm.

5. The eigenvectors are computed over a given frequency range or, at user option, for a given number of eigenvalues using inverse iteration. Since the roots are known to high accuracy at this point, the vectors converge rapidly.

6. Physical eigenvectors are recovered by performing the inverse transformation to that made in Step 3 and then solving **Eq. A-5** as follows:

$$\{u\} = [L]^{T,-1}\{w\}$$

**Equation A-7.**

7. if a SUPORT entry is used, the first $N_r$ eigenvectors computed are discarded. The rigid body modes from the SUPORT computation are put in their place, and the corresponding eigenvalues are set to zero. The SUPORT operation does cause some redundant operations to be performed as the rigid body modes are computed by two methods and then one set is discarded. This is generally a minor issue.

8. The back-transformed vectors [*PHI*] are given a final stage of orthogonalization with respect to the mass matrix. The modal mass matrix is calculated as follows:

$$[M_{modal}] = [\phi]^T[M][\phi]$$

**Equation A-8.**

then decomposed,

$$[M_{modal}] = [L]^T[L]$$

**Equation A-9.**

The refined eigenvectors, $[\bar{\phi}]^T$ are found by a forward pass on the equation:

$$[L][\bar{\phi}]^T = [\phi]^T$$

**Equation A-10.**

Note that the tridiagonal methods fail in Step 1 if the mass matrix is not positive definite. In order to minimize this eventuality, degrees-of-freedom with null columns are removed from (*u*) by the automatic application of static condensation. This is called AUTO-OMIT. (This procedure can be bypassed by using PARAM,ASING). The application of the AUTO-OMIT process is a precaution and may not remove all possible causes of mass matrix singularity such as, a point mass offset from a grid point, but it greatly improves the reliability and convenience of the standard tridiagonal methods.

## The Modified Tridiagonal Methods (MHOU, AHOU)

The modified tridiagonal method, Modified Householder (MHOU), is similar to the standard tridiagonal method (HOU) with the exception that the mass matrix can be singular.

The steps used in the modified methods are as follows:

1. Perform a Cholesky decomposition of the positive definite matrix

$$[K + \lambda_s M] = [L][L]^T$$

**Equation A-11.**

where $\lambda_s$ is a positive number automatically selected by the program to optimize the reliability and accuracy of eigenvalue extraction. It is large enough to stabilize all rigid body modes but also be as small as possible to reduce the numerical contamination. The goal of the algorithm is to find a

value near the first flexible frequency of the system. It uses the diagonal terms of the mass and stiffness matrices as follows:

$$\lambda_s = \frac{1}{N^{1/2} \sum_{i=1}^{N} \frac{M_{ii}}{K_{ii}}}$$

**Equation A-12.**

where $N$ is the dimension of the matrices. If $M_{ii}$ or $X_{ii} = 0$ (an under defined degree-of-freedom) or $M_{ii}/X_{ii} < 10^{**}8$ (a likely "very large" mass term used to approximate a rigid body mode), the corresponding term is omitted from the summation. If the value calculated by this approximate method does not result in a stable Cholesky decomposition, $\lambda_s$ is increased by adding the arbitrary amount of 0.01. If this value also results in failure, the original value is increased by 0.02. If this value also fails, a fatal error is issued.

2. Write the eigen-equation standard form as:

$$[\bar{J} - \bar{\lambda} I]\{\bar{w}\} = 0$$

**Equation A-13.**

where:

$$\bar{\lambda} = \frac{1}{\lambda + \lambda_s}$$

**Equation A-14.**

and:

$$[\bar{J}] = [L]^{-1}[M][L]^{-1,T}$$

**Equation A-15.**

by rearranging the terms to read as follows:

$$[K + \lambda_s M - (\lambda + \lambda_s)M]\{u\} = 0$$

**Equation A-16.**

Then premultiply **Eq. A-16** by $-1/(\lambda + \lambda_s)$ and substitute for $[K + \lambda_s M]$ from **Eq. A-11** to obtain:

$$\left[M - \frac{LL^T}{\lambda + \lambda_s}\right]\{u\} = 0$$

**Equation A-17.**

Premultiply by $[L]^{-1}$ and then substitute:

$$\{\bar{w}\} = [L]^T\{u\}$$

**Equation A-18.**

resulting in **Eq. A-13**, **Eq. A-14**, and **Eq. A-15**.

3. The remaining steps are identical to those for the unmodified methods where:

$$[J] = [\tilde{J}] \text{ and } \{w\} = \{\bar{w}\}$$

**Equation A-19.**

Although the mass matrix is not required to be nonsingular in the modified methods, a singular mass matrix can produce one or more zero eigenvalues, $\lambda_s$, which are equivalent to infinite eigenvalues, $\lambda_i$. Due to roundoff error, these will appear in the output as very large positive or negative eigenvalues. To reduce the incidence of such meaningless results, degrees-of-freedom with null masses are eliminated by the AUTO-OMIT static condensation as in the case of the unmodified methods. This step can be bypassed by using PARAM,ASING.

Many times the user is not conversant in the status of the mass matrix prior to the actual matrix assembly. To assist the user in choosing the appropriate method, Automatic Householder (AHOU) is available. The automatic method, AHOU, initially uses the standard method. In the first step of the method, if the mass matrix is not well-conditioned for decomposition, the method shifts to the corresponding modified method. The modified method is more expensive and introduces numerical "noise" due to the shift, but it resolves most of the numerical problems of the ill-conditioned mass matrix.

Operations involved in the computation of the natural frequencies and mode shapes is proportional to the number of degrees-of-freedom (*N*). The time-consuming operations for the tridiagonal methods are mass matrix decomposition and reduction to standard form ($5/2\ N^3$ for a full mass matrix, negligible for

a diagonal matrix) tridiagonalization (2/3 $N^3$ for HOU), QR root iteration (9 $N^2$), inverse iteration for vectors ($N^2$ per vector), and inverse transformation of vectors ($N^2$ per vector for HOU). As $N$ becomes large (greater than 100, perhaps), terms in $N^3$ dominate.

## The Inverse Power Method (SINV)

The inverse power method with shifts is particularly effective when the mass and stiffness matrices are sparse and when only a few of the eigenvalues are required. In Simcenter Nastran, the method is used as a stand-alone method to find all of the eigenvalues within a user-specified domain.

If we define the eigenvalue problem as:

$$[K - \lambda M]\{u\} = 0$$

**Equation A-20.**

Let:

$$\lambda = \lambda_o + \Lambda$$

**Equation A-21.**

where $\lambda_o$ is called the shift point.

The iteration algorithm is:

$$[K - \lambda_o M]\{w_n\} = [M]\{u_{n-1}\}$$

**Equation A-22.**

$$\{u_n\} = \frac{1}{C_n}\{w_n\}$$

**Equation A-23.**

where $C_n$ is equal to the value of the element of $\{w_n\}$ with the largest absolute value. $\{u_n\}$ is then placed on the right side of **Eq. A-22** and iteration continues. It is easy to prove that $1/C_n$ converges to $\lambda_1$, the shifted eigenvalue nearest to the shift point, and that $\{u_n\}$ converges to the corresponding eigenvector of **Eq. A-20**. The following observations are pertinent.

1. A triangular decomposition of the matrix $[K - \lambda_o M]$ is required in order to evaluate $\{w_n\}$ from **Eq. A-22**. The effort required to perform the triangular decomposition is greatly reduced if $[K - \lambda_o M]$ is

a narrow band matrix. Structural analysis is characterized by narrowly banded stiffness and mass matrices. From Sturm's theorem, the number of negative terms on the factor diagonal is exactly equal to the number of roots below $\lambda_o$. This number is printed out at every shift and is identified as the Sturm sequence number.

2. It is unnecessary for [*K*] to be nonsingular, so that rigid body modes cause no special difficulty. Similarly, null rows in [*K*] cause no problems. If they are removed by static condensation, they will increase the cost of the solution unnecessarily. This occurs primarily due to the increase in bandwidth inherent in static condensation

3. The shift point $\lambda_o$ may be changed (at the cost of an additional triangular decomposition) in order to improve the rate of convergence toward a particular eigenvalue or to improve the accuracy and convergence rate after several roots have been extracted from a given shift point. The next shift point is computed from the residual terms remaining after computing the present root by a root estimation algorithm (See the*Simcenter Nastran Numerical Methods User's Guide* for details.)

4. $\lambda_o$ can be placed to obtain the eigenvalues within a desired frequency band and not just those that have the smallest absolute value.

5. The Sturm sequence data can be used to ensure that all eigenvalues within the requested frequency band have been found. This can be done automatically with the SINV method (see the *Simcenter Nastran Numerical Methods User's Guide* for details). In the SINV method, the requested band is broken into subregions, and the Sturm number is determined at each boundary of the subregion. If the root estimation algorithm predicts that the next root is out of the subregion but the Sturm sequence data states that more roots remain in the subregion, a bisection method is used to iterate to the missing roots. All roots for a subregion must be found before the method is allowed to enter the next subregion.

In the practical application of the tracking methods, most of the numerical efficiency issues (trial eigenvectors, convergence criteria, etc.) are automatically handled by Simcenter Nastran. The shift points must be selected so that the numerical algorithms can operate in an optimum manner. Generally, this is handled by the software, but you can assist the process by carefully selecting the frequency range.

If you use the SUPORT entry, the software directly uses the rigid body modes as the first set of eigenvectors with their eigenvalues set to zero. Iteration then proceeds for the nonzero roots. Using SUPORT reduces costs of the SINV method considerably and also provides the aesthetic benefits of exact zero-frequency rigid body modes. In general, you should use the SUPORT entry with these methods.

## The Shifted Block Lanczos Method

The Lanczos method (LAN) available in Simcenter Nastran is an extremely reliable and accurate method for solving the large sparse eigenvalue problems typical of many structures. Simcenter Nastran uses a shifted block Lanczos algorithm.

For the standard eigenvalue problem written in the form:

$$[A]\{x\} = \lambda\{x\}$$

**Equation A-24.**

the Lanczos method computes a sequence of vectors *q* called Lanczos vectors and scalars $Q_j$,*A*. The basic recurrence relation for the j-th step of the Lanczos algorithm can be written as follows:

$$\{r_{j+1}\} = [A]\{q_j\} - \alpha_j\{q_j\} - \beta_j\{q_{j-1}\}$$

**Equation A-25.**

$$\alpha_j = \{q_i\}^T[A]\{q_j\}$$

**Equation A-26.**

$$\beta_{j+1} = \|r_{j+1}\|$$

**Equation A-27.**

$$\{q_{j+1}\} = \frac{\{r_{j+1}\}}{B_{j+1}}$$

**Equation A-28.**

The first j-steps of the method can be combined and written in matrix form:

$$[A][Q_j] - [Q][T_j] = \beta_{j+1}\{q_{j+1}\}\{e_j\}^T$$

**Equation A-29.**

where $\{e_j\}$ = unit vector with the j-th component equal to one and all other components equal to zero

$$\{Q_j\} = (\{q_1\}, \{q_2\}, ..., \{q_j\}) \text{ with } [Q_j^T][Q_j] = [I]$$

**Equation A-30.**

$$[T_j] = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_2 & & \\ & \beta_3 & \alpha_3 & \beta_4 & \\ & & \ddots & \ddots & \ddots \\ & & & \beta_j & \alpha_j \end{bmatrix}$$

**Equation A-31.**

The eigenvalues of the tridiagonal matrix $[T_j]$ are called Ritz values. Some of these Ritz values are good approximations to eigenvalues of $[A]$. The corresponding Ritz vectors ($x_i$) are computed from the eigenvectors of $[T_j]$ and the Lanczos vectors. An eigenpair is defined as an eigenvalue ($s_i$) and its corresponding eigenvector $\{\theta_{ik}\}$. Let ($\{\theta_{ik}\}$, $s_i$) be an eigenpair of the tridiagonal matrix $[T_j]$. Then an approximate eigenpair for the original problem is ($\lambda_i$, $\{x_i\}$) where:

$$\lambda_i = \theta_i$$

**Equation A-32.**

$$\{x_i\} = \{Q_j\}\{s_i\}$$

**Equation A-33.**

As implied by the formulation of the algorithm, the method computes a tridiagonal matrix and a set of Lanczos vectors that become larger with each iteration. The algorithm in exact arithmetic can be terminated when $B_{j+1}$ or when sufficiently many eigenvalues of $[A]$ have been computed. Lanczos also handles the effects of roundoff.

Simcenter Nastran uses a shifted block version of the basic Lanczos recurrence relation. The blocking approach to the Lanczos algorithm (working with blocks of $p$ Lanczos vectors simultaneously) enables the method to handle eigenvalues of high multiplicity including shifting in the algorithm which helps to speed the convergence of the method.

In this method, the $A$ matrix is replaced by the sum of the stiffness matrix plus shifted mass matrix, similar to the approach in the modified tridiagonal methods. This allows solution of problems with rigid body modes and generally faster convergence. The shifts (trial roots) are influenced by user input. The user is required to input either a frequency band of interest, a desired number of roots, or a combination of this data.

As an example, suppose that both the lower bound (V1) and upper bound (V2) of the frequency band are defined by the user as well as the number of roots desired (ND). The main objective for the shift strategy is to encompass the region containing the eigenvalues of interest in which all of the eigenvalues have been computed and the number of eigenvalues has been verified with a Sturm sequence check.

The first shift is at V1 to determine the Sturm sequence number. The factor matrix is stored for possible later use. If V1 is not defined by the user, the first shift is made at $\lambda_s$. The second shift point is chosen at V2.

After a Lanczos iteration based on the selected shift and the computation of the corresponding Ritz values, some values will be found to be acceptable eigenvalues while other eigenpairs will fail convergence tests. A decision is made as to whether it is more efficient to re-shift or continue with further iteration. The re-shift point is chosen from the existing Ritz values (smaller than V2) or from the stored factor from the shift at V2. At this point, the number of accepted eigenvalues and the Sturm sequence numbers at each end of the region are known.

If iteration with the factor from V2 does not produce all of the roots desired, two more types of shift will be used. If it appears that the missing roots are near the lowest shift, an approximate first mode eigenvalue is used. This is either a nonzero SHFSCL field from the EIGRL entry or the a similar shift term used in the modified tridiagonal methods. If the missing roots are elsewhere, bisection between prior shifts is used. If all requested eigenpairs still cannot be found, a user information message is printed, and all accepted eigenpairs are output. If some roots have been found, the module exits normally. A fatal error is issued if no roots have been found. If accepted Ritz values are found outside the requested region, they are discarded.

The strategy for selecting block size is to first find the minimum of N/2, 2E, p and MAXSET. N is the problem size, E the number of roots desired, p the number of Lanczos vectors that will fit into memory, and MAXSET the user selection for block size (which has a default of 7).

Another efficiency feature unique to the Lanczos method is selective orthogonalization of the trial eigenvectors. All of those contained in the same block are made orthogonal to one another. Tests are made, at several steps, to determine if further orthogonalization between vectors in different blocks is required. This avoids the cost term cubic on E (the number of eigenvectors calculated). which becomes the dominant term for the SINV methods as E becomes large.

If the SUPORT entry is used, the $[D_{ir}]$ matrix is merged with an identity matrix $[I_{rr}]$ to form raw rigid body vectors $[D_{ar}]$. These raw vectors are used as initial vectors. If the r-set has been well-chosen, the rigid body modes will converge rapidly. If they have been chosen poorly, the Lanczos iterations will continue until a good set of eigenvectors have been found. The corresponding roots are set to the aesthetically pleasing value of zero rather than to the small numerical zeros likely to be produced by the actual computations.

The Lanczos method is more forgiving than the other methods since it repairs the effects of a poor r-set selection. However, the repeatability of rigid body modes computed by Lanczos with the SUPORT entry is not the same as with other extraction methods. Rigid body modes computed using the SUPORT entry and a non-Lanczos method will produce identical modes, irrespective of the method. If the rigid body modes produced by the Lanczos method with SUPORT entries are compared with those produced by any

of the other methods, the Lanczos modes will be a proper linear combination of the modes from the other methods and may be markedly different in appearance. Furthermore, if small changes are made to the stiffness matrix, the Lanczos modes again may be markedly different, although they are again a proper linear combination of the set produced by other methods. This lack of repeatability makes the Lanczos method unusable with the scaled response spectra capability of Simcenter Nastran.

The cost for the LAN method is dominated by the decomposition, inverse iteration, and orthogonalization operations at the size of the input matrices N. The number of roots per decomposition varies by problem and is difficult to determine *apriori*.

## User Interface for Real Eigenvalue Analysis

The EIGR and EIGRL Bulk Data entries are the heart of the user interface because they define a method and select parameters that control the eigenvalue extraction procedure. The EIGRL entry is used for the LANCZOS method, and the EIGR entries are used for all of the other methods.

The data used for the EIGR entry depend on the method selected. The basic data required is related to the eigenvalue extraction method chosen and to the frequency range or number of required roots. The basic format of the Bulk Data entry is as follows:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| EIGR | SID | METHOD | F1 | F2 | NE | ND | | | +BC |
| +BC | NORM | GID | CID | | | | | | |

The SID (Field 2) is the Set ID associated with this EIGR entry. A particular EIGR entry is activated in an analysis using the METHOD = SID entry in the Case Control Section. If the Set ID on the METHOD and EIGR match, the eigenvalues will be extracted using the method referenced on that entry.

The METHOD Entry (Field 3) selects the eigenvalue method from the following list:

1. SINV Sturm Modified Inverse Power
2. HOU Householder Method
3. MHOU Modified Householder Method
4. AHOU Automatic Householder

The F1 Entry (Field 4) is used to specify the lowest frequency of interest in the eigenvalue extraction.

The F2 Entry (Field 5) is used to specify the highest frequency of interest in the eigenvalue extraction.

The ND entry (Field 7) is used to specify the desired number of roots in the range of interest.

The NORM (Field 2 – continuation) entry on the continuation card is used to specify the method of eigenvector normalization. The choices are:

1. MASS Mass normalization (Default – if used continuation entry not required)

2. MAX Normalization to maximum A-set component

3. POINT Normalization to user-defined DOF

The POINT entry (Field 3 – continuation) is used to specify the point of interest from the analysis set for POINT normalization only.

The CID entry (Field 4 – continuation) is used to specify the component of displacement for POINT normalization only.

There is an interrelationship between the F1, F2, and ND field on the EIGR entry.

**Table A-2. EIGR Input Parameters**

| | When METHOD = INVor SINV | When METHOD = HOU, or MHOU |
|---|---|---|
| F1,F2 | Frequency range of interest (Real ≥ 0.0). F1 must be input. If METHOD = SINV and ND is blank, then F2 must be input. | Frequency range of interest (Real ≥ 0.0; F1 < F2. (If ND is blank, F1 and F2 are ignored. If ND is blank, eigenvectors are found whose natural frequencies lie in the range between F1and F2. |
| NE | Estimate of number of roots in range. Not used by SINV method. | Not used. |
| ND | Desired number of roots. If this field is blank and METHOD = SINV, then all roots between F1 and F2 are searched and the limit is 600 roots. | Desired number of eigenvectors (Integer ≥ 0). If ND is zero, the number of eigenvectors is determined from F1 and F2 (Default = 0). If all three are blank, then ND is automatically set to one more than the number of degrees-of-freedom listed on SUPORT entries. |

The rules for METHOD = HOU, MHOU, and AHOU are identical. If you select any of these methods, Simcenter Nastran finds all the eigenvalues but only computes the eigenvectors specified by F1 and F2 or those specified by ND (the “desired” number).

F1 and F2 specify the lower and upper bounds of the frequency range in which eigenvectors are computed, and ND specifies the number of eigenvectors, starting with the lowest (or the first rigid body mode, if present). If F1, F2, and ND entries are present, ND takes precedence. Note that the default for F1 is 0.0.

The following examples demonstrate the use of the EIGR data entry.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| EIGR | 1 | AHOU | | | 10 | | | | |

In the first example (SID = 1), the automatic Householder method is selected with 10 roots requested. Since the default “MASS” eigenvector normalization is requested, no continuation entry is needed.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| EIGR | 2 | AHOU | | 100.0 | | | | | +2 |
| +2 | MAX | | | | | | | | |

For the second example (SID = 2), the same method is requested, but all the roots below 100 cycles per unit time are requested with “MAX” vector normalization.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| EIGR | 3 | SINV | 0.0 | 100.0 | | 6 | | | +3 |
| +3 | POINT | 32 | 4 | | | | | | |

For the third example (SID = 3), the inverse iteration method with Sturm sequence checks is requested for the first six roots in the range specified. The POINT normalization is requested for grid point 32 in the R1 direction.

If METHOD = SINV, the values of F1, F2, and ND determine both the number of eigenvalues and the eigenvectors that will be computed. These entries also provide hints to help Simcenter Nastran find the eigenvalues. F1 and F2 specify the frequency range of interest within which Simcenter Nastran will search for modes. Simcenter Nastran attempts to find all of the modes in the range F1, F2 or the number specified by ND, whichever is less. If searching stops because ND modes are found, there is no guarantee that they will be the lowest eigenvalues. Iterative methods only guarantee the solution for modes nearest the shift point. If ND modes are not found in the range of interest, SINV will usually find one mode outside the range F1, F2 (or possibly more) before stopping the search.

The inverse power method is particularly efficient when only a small number of eigenvalues and eigenvectors are wanted. Very often only the lowest mode is of interest. The following example illustrates an EIGR entry which will extract only the lowest nonzero eigenvalue.

| EIGR | SID | METHOD | F1 | F2 | NE | ND | | | |
|---|---|---|---|---|---|---|---|---|---|
| EIGR | 13 | SINV | 0.0 | .011 | | | | | |

It is assumed in the example that the frequency of the lowest mode is greater than 0.01 cycles per unit time. The program will find one eigenvalue outside the range F1, F2 and stop the search. The eigenvalue found is guaranteed to be the lowest non zero eigenvalue (or a member of the lowest closely spaced cluster of eigenvalues in cases with pathologically close roots) provided that there are no negative eigenvalues and that the SUPORT entry has been used to specify the correct number of zero eigenvalues.

The fields of the EIGRL entry are designed to select and set parameters for the Lanczos method as well as request diagnostics.

| EIGRL | SID | V1 | V2 | ND | MSGL | VL | MAXSET | SHFSCL | |
|---|---|---|---|---|---|---|---|---|---|
| EIGRL | 1 | 0.1 | 3.2 | | 10 | | | | |

The V1 field defines the lower frequency bound, the V2 field defines the upper frequency bound, and the ND field defines the number of eigenvalues and eigenvectors desired in the region. The V1 and V2 are expressed in units of cycles per unit time. Examples of the results of using explicit or default values for the V1, V2, and ND fields are shown in **Table A-8**. The defaults on the EIGRL entry are designed to provide the minimum number of roots in cases where the input is ambiguous.

**Table A-8. Model Outputs from EIGRL Input Options**

| Case | V1 | V2 | ND | Number and Type of Roots Found |
|---|---|---|---|---|
| 1 | V1 | V2 | ND | Lowest ND or all in range; whichever is smaller |
| 2 | V1 | V2 | | All in range |
| 3 | V1 | | ND | Lowest ND in range [V1, +00] |
| 4 | V1 | | | Lowest root in range [V1, +00] |
| 5 | | | ND | Lowest ND roots in [-00, +00] |
| 6 | | | | Lowest root |
| 7 | | V2 | ND | Lowest ND roots below V2 |
| 8 | | V2 | | All below V2 |

The MSGLVL field of the EIGRL entry is used to control the amount of diagnostic output provided by Lanczos. The default value of 0 produces no diagnostic output. The values 1, 2, or 3 provide more output with the higher values providing increasingly more output. Typically, the default value is sufficient. In some cases, higher diagnostics levels may be in order to help resolve difficulties with special modeling problems.

The MAXSET field is used to control the block size, the number of appropriate eigenvectors computed in the outermost iteration loop. The default value of 7 is recommended for most applications. There may be special cases where a larger value may result in quicker convergence of highly multiple roots or a lower value may result in more efficiency when the structure is lightly coupled. However, the default value has been chosen after reviewing results from a wide range of problem types on several different computer types with the goal of minimizing the sum of CPU and I/O cost.

Often, the block size may be reset by Simcenter Nastran during the run because there is insufficient memory for a block size of 7. Computational efficiency often degrades as the block size decreases. Therefore, you should examine the EIGENVALUE ANALYSIS SUMMARY output to determine whether the software has sufficient memory to use an efficient block size. A smaller block size may be more efficient when only a few roots are requested. The minimum recommended block size is 2.

The SHFSCL field allows a user-designated shift to be used when all other shifting strategies fail. This entry is only used under special circumstances. However, when used, its value should be set to the expected first natural frequency.

The Lanczos method normalizes the computed eigenvectors using the MASS method. The other normalization methods are not available.

## Solution Control For Normal Modes

When used as an independent solution, normal modes analysis is available in SOL 103 of the Structured Solution Sequences. All eigenvalue extraction methods are available in all of these solutions, with one exception. The Executive Control Section can also contain diagnostic DIAG 16 which will print the iteration information used in the SINV method.

One of the most important entries in the Case Control Section is the METHOD entry. This entry is required. The Set ID referenced by the METHOD entry refers to the Set ID of an EIGR entry or an EIGRL entry in the Bulk Data.

When you perform a modal extraction, the Simcenter Nastran output file contains various diagnostic messages and an Eigenvalue Table. Optional grid and element output are available using standard Case Control output requests. These requests are summarized in Table A-9.

**Table A-9. Eigenvalue Extraction Output Requests**

| Grid Output | |
|---|---|
| DISPLACEMENT | Requests the eigenvector (mode shape) for a set of grid points |
| GPFORCE | Requests the Grid Point Force Balance Table to be computed for each mode |
| GPSTRESS | Requests grid point stresses to be computed for a set of grid points. This request must be accompanied by the ELSTRESS Case Control request and the definition of Stress Surfaces and/or Stress Volumes in the OUTPUT(POST) section of the Case Control. |
| SPCFORCE | Requests Forces of Single Point Constraint to be computed for a set of grid points for each mode. |
| **Element Output** | |
| ELSTRESS(or STRESS) | Requests the computation of modal stresses for a set of elements for each mode |
| ESE | Requests the computation of modal element strain energies for a set of elements for each mode |
| ELFORCE(or FORCE) | Requests the computation of modal element forces for a set of elements for each mode |
| STRAIN | Requests the computation of the modal element strains for a set of elements |
| MODES | A special Case Control request which permits selective output requests to be processed on selective modes |

In addition to Bulk Data entries required to define the structural model, the only required Bulk Data entry is the eigenvalue selection entry, EIGR or EIGRL. The EIGR entry is used to select the modal extraction

parameters for the Householder, Modified Householder, and Automatic Householder methods. The EIGRL entry is used to select the modal extraction parameters for the Lanczos method.

# B. Set Notations

## B.1 Degree-of-Freedom Sets

Structural matrices are initially assembled in terms of all structural points, which excludes only the extra points introduced for dynamic analysis. These matrices are generated with six degrees-of-freedom for each geometric grid point and a single degree-of-freedom for each scalar point. These degrees-of-freedom are partitioned into sets, based on user inputs. The sets are used to successively eliminate variables during the solution process, as described in “Data Processing and Matrix Operations by Functional Module” in the *Simcenter Nastran User’s Guide*

### Degree-of-Freedom Set Definitions

Each degree-of-freedom is a member of one mutually exclusive "set". Set names have the following definitions:

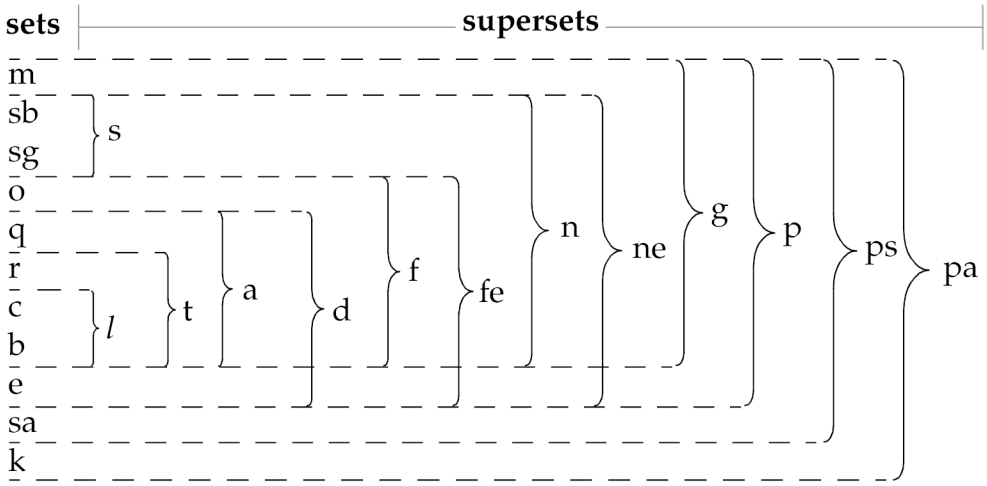
| Set Name | Meaning |
|---|---|
| m | degrees-of-freedom eliminated by multipoint constraints. |
| sb* | degrees-of-freedom eliminated by single-point constraints that are included in boundary condition changes and by the AUTOSPC feature. |
| sg* | degrees-of-freedom eliminated by single-point constraints that are specified on the PS field on GRID Bulk Data entries. |
| o | degrees-of-freedom omitted by structural matrix partitioning. |
| q | generalized degrees-of-freedom for dynamic reduction or component mode synthesis. |
| r | reference degrees-of-freedom used to determine free body motion. |
| c | degrees-of-freedom which are free during component mode synthesis or dynamic reduction. |
| b | degrees-of-freedom fixed for component mode analysis or dynamic reduction. |
| e | extra degrees-of-freedom introduced in dynamic analysis. |
| sa | permanently constrained aerodynamic degrees-of-freedom. |
| k | aerodynamic degrees-of-freedom. |

Note:

* Strictly speaking, sb and sg are not exclusive with respect to one another. Degrees-of-freedom may exist in both sets simultaneously. Since these sets are not used explicitly in the solution sequences, this need not concern the user. However, those who use these sets in their own DMAPs should avoid redundant specifications when using these sets for partitioning or merging operations. That is, a degree-of-freedom should not be specified on both a PS field of a GRID entry (sg set) and on a selected SPC entry (sb set). Redundant specifications will cause UFM 2120 in the

VEC module and behavior listed in the *N Nastran DMAP Programmer's Guide* for the UPARTN module. These sets are exclusive, however, from other mutually exclusive sets.

Each degree-of-freedom is also a member of one or more combined sets called "supersets." Supersets have the following definitions:




| Set Name | Meaning (+ Indicates Union of Two Sets) |
|---|---|
| *s=sb+sg* | All degrees-of-freedom eliminated by single-point constraints. |
| *l=b+c* | The structural coordinates remaining after the reference coordinates are removed (degrees-of-freedom left over). |
| *t=l+r* | The total set of physical boundary degrees-of-freedom for superelements. |
| *a=t+q* | The set assembled in superelement analysis. |
| *d=a+e* | The set used in dynamic analysis by the direct method. |
| *f=a+o* | Unconstrained (free) structural degrees-of-freedom. |
| *fe=f+e* | Free structural degrees-of-freedom plus extra degrees-of-freedom. |
| *n=f+s* | All structural degrees-of-freedom not constrained by multipoint constraints. |
| *ne=n+e* | All structural degrees-of-freedom not constrained by multipoint constraints plus extra degrees-of-freedom. |
| *g=n+m* | All structural (grid) degrees-of-freedom including scalar degrees-of-freedom. |
| *p=g+e* | All physical degrees-of-freedom. |
| *ps=p+sa* | Physical and constrained aerodynamic degrees-of-freedom. |
| *pa=ps+k* | Physical set for aerodynamics. |
| *fr=o+l* | Statically independent set minus the statically determinate supports (*fr=f–q–r*). |
| *v=o+c+r* | The set free to vibrate in dynamic reduction and component mode synthesis. |

The a-set and o-set are created in the following ways:

1. If only OMITi entries are present, then the o-set consists of degrees-of-freedom listed explicitly on OMITi entries. The remaining f-set degrees-of-freedom are placed in the b-set which is a subset of the a-set.

2. If ASETi or QSETi entries are present, then the a-set consists of all degrees-of-freedom listed on ASETi entries and any entries listing its subsets, such as QSETi, SUPORTi, CSETi, and BSETi entries. Any OMITi entries are redundant. The remaining f-set degrees-of-freedom are placed in the o-set.

3. If there are no ASETi, QSETi, or OMITi entries present but there are SUPORTi, BSETi, or CSETi entries present then the entire f-set is placed in the a-set and the o-set is not created.

4. There must be at least one explicitly ASETi, QSETi, or OMITi entry for the o-set to exist, even if the ASETi, QSETi, or OMITi entry is redundant.

In dynamic analysis, additional vector sets are obtained by a modal transformation derived from real eigenvalue analysis of the a-set. These sets are as follows:

$\xi_0$ = rigid body (zero frequency) modal degrees-of-freedom
$\xi_f$ = finite frequency modal degrees-of-freedom
$\xi_i$ = $\xi_0 + \xi_f$, the set of all modal degrees-of-freedom

One vector set is defined that combines physical and modal degrees-of-freedom:

$u_h$ = $\xi_i + u_e$, the set of all modal degrees-of-freedom

The membership of each degree of freedom can be printed by use of the Bulk Data entries PARAM,USETPRT and PARAM,USETSEL.

## Degree of Freedom Set Bulk Data Entries

Degrees-of-freedom are placed in sets as specified by the user on the following Bulk Data entries:

| Name | Bulk Data Entry Name |
|---|---|
| m | MPC, MPCADD, RBAR, RBE1, RBE2, RBE3, RROD, RSPLINE, RTRPLT |
| sb | SPC, SPC1, SPCADD, BNDGRID, (PARAM,AUTOSPC,YES) |
| sg | GRID, GRDSET (PS field) |
| o | OMIT, OMIT1, GRID (SEID field), SESET |
| q | QSET,QSET1 |
| r | SUPORT, SUPORT1 |
| c | CSET,CSET1 |
| b | BSET, BSET1 |
| e | EPOINT |

| Name | Bulk Data Entry Name |
|---|---|
| sa | CAEROi |
| k | CAEROi |
| a | ASET, ASET1, Superelement exterior degrees-of-freedom, CSUPEXT |

In superelement analysis, the appropriate entry names are preceded by the letters SE, and have a field reserved for the superelement identification number. This identification is used because a boundary (exterior) grid point may be in one mutually exclusive set in one superelement and in a different set in the adjoining superelement. The SE-type entries are internally translated to the following types of entry for the referenced superelement:

| Entry Type | Equivalent Type |
|---|---|
| SEQSETi | QSETi |
| SESUP | SUPORT |
| SECSETi | CSETi |
| SEBSETi | BSETi |

## B.2 Multipoint Constraints

Each multipoint constraint is described by a single equation that specifies a linear relationship for two or more degrees-of-freedom. In static analysis, multiple sets of multipoint constraints can be provided in the Bulk Data Section, with selections made at execution time by using the subcase structure in Case Control. Multipoint constraints are also discussed in Section 5.4 of *The NASTRAN Theoretical Manual*.

Multipoint constraints are defined on MPC and MPCADD entries. MPC is the basic entry for defining multipoint constraints. The first component specified on the entry is the dependent degree-of-freedom, i.e., that degree-of-freedom that is removed from the equations of motion. Dependent degrees-of-freedom may appear as independent terms in other equations of the set; however, they may appear as dependent terms in only a single equation. The MPCADD entry defines a union of multipoint constraints. Some uses of multipoint constraints are:

- To describe rigid elements and mechanisms such as levers, pulleys, and gear trains. In this application, the degrees-of-freedom associated with the rigid element that are in excess of those needed to describe rigid body motion are eliminated with multipoint constraint equations. Treating very stiff members as rigid elements eliminates the ill-conditioning associated with their treatment as ordinary elastic elements.

- To enforce zero motion in directions other than those corresponding with components of the global coordinate system. In this case, the multipoint constraint will involve only the degrees-of-freedom at a single grid point. The constraint equation relates the displacement in the direction of zero motion to the displacement components in the global system at the grid point.

- To be used with scalar elements to generate nonstandard structural elements and other special effects.

- To describe parts of a structure by local vibration modes. This application is treated in Section 14.1 of *The NASTRAN Theoretical Manual*. The general idea is that the matrix of local eigenvectors represents a set of constraints relating physical coordinates to modal coordinates.

In general, the user must provide the coefficients in the multipoint constraint equations. However, several rigid elements have been introduced which will generate the MPC equations for some applications:

| | |
|---|---|
| **RROD** | A pin-ended rod which is rigid in extension. |
| **RBAR** | A rigid bending element with six degrees-of-freedom at each end. |
| **RTRPLT** | A rigid triangular plate with six degrees-of-freedom at each vertex. |
| **RBE1** | A rigid body connected to an arbitrary number of grid points. The independent and dependent degrees-of-freedom can be arbitrarily selected by the user. |
| **RBE2** | A rigid body connected to an arbitrary number of grid points. The independent degrees-of-freedom are the six components of motion at a single grid point. The dependent degrees-of-freedom are specified at an arbitrary number of grid points. |
| **RBE3** | Defines the motion at a reference grid point as the weighted average of the motions at a set of other grid points. |
| **RSPLINE** | Defines multipoint constraints for the interpolation of displacements at grid points. |

The rigid elements will always meet equilibrium and continuity requirements, whereas this is a user responsibility for MPC equations.

Multipoint forces of constraint may be output with MPCFORCE Case Control command in SOLution Sequences 101 through 200.

## B.3 Single Point Constraints

A single point constraint (SPC) applies a fixed value to a translational or rotational component at a geometric grid point or to a scalar point. Common uses of single point constraints are to specify the boundary conditions of a structural model by fixing the appropriate degrees-of-freedom and to eliminate unwanted degrees-of-freedom with zero stiffness. Multiple sets of single point constraints can be provided in the Bulk Data Section, with selections made at execution time by using the subcase structure in the Case Control Section. This procedure is particularly useful in the solution of problems having one or more planes of symmetry.

The elements connected to a grid point may not provide resistance to motion in certain directions, causing the stiffness matrix to be singular. Single point constraints are used to remove these degrees-of-freedom from the stiffness matrix. A typical example is a planar structure composed of membrane and extensional elements. The translations normal to the plane and all three rotational degrees-of-freedom must be constrained since the corresponding stiffness matrix terms are all zero. If a grid point has a direction of zero stiffness, the single point constraint needs not be exactly in that direction, but only

needs to have a component in that direction. This allows the use of single point constraints for the removal of such singularities regardless of the orientation of the global coordinate system. Although the displacements will depend on the direction of the constraint, the internal forces will be unaffected.

One of the tasks performed by the Grid Point Singularity Processor (GPSP) is to examine the stiffness matrix for singularities at the grid point level. Singularities remaining at this level, following the application of the multipoint and single point constraints, are listed in the Grid Point Singularity Table (GPST), which is automatically printed. The GPST lists all singular degrees-of-freedom, in the global coordinate system, and the ratio of stiffness between the softest and stiffest degree-of-freedom for the grid point. The user may request that single point constraints be generated for all identified singularities by use of the Bulk Data entry PARAM,AUTOSPC,YES.

Single point constraints are defined on SPC, SPC1, and SPCADD entries. The SPC entry is the most general way of specifying single point constraints. The SPC1 entry is a less general entry that is more convenient when several grid points have the same components constrained to a zero displacement. The SPCADD entry defines a union of single point constraint sets specified with SPC or SPC1 entries.

Single point constraints can also be defined on the GRID entry. In this case, however, the points are constrained for all subcases. The default value for enforced displacement on points constrained on GRID entries is zero. The default value can be overridden at the subcase level with SPC entries.

The printed output for single point forces of constraints can be requested in Case Control. All nonzero forces are printed, whether they originate from SPCi entries, the PS field on GRID entries, or by PARAM,AUTOSPC,YES.

## B.4 Rigid Body Supports

In the following discussion, a free body is defined as a structure that is capable of motion without internal stress; i.e., it has one or more rigid body degrees-of-freedom. The stiffness matrix for a free body is singular with the defect equal to the number of stress-free, or rigid body modes. A solid three-dimensional body has up to six rigid body modes. Linkages and mechanisms can have a greater number. In order to permit the analysis of mechanisms, no restriction is placed in the program on the number of stress-free modes.

Free-body supports are defined with a SUPORT or SUPORT1 entry. Free-body supports must be defined in the global coordinate system. The SUPORT1 entry must be selected by the SUPORT1 Case Control command.

In static analysis by the displacement method, the rigid body modes must be restrained in order to remove the singularity of the stiffness matrix. The required constraints may be supplied with single point constraints, multipoint constraints, or free body supports. If free body supports are used, the rigid body characteristics will be calculated and a check will be made on the sufficiency of the supports. Such a check is obtained by calculating the rigid body error ratio and the strain energy as defined in the Rigid Body Matrix Generator operation. This error ratio and the strain energy are automatically printed following the execution of the Rigid Body Matrix Generator. The error ratio and the strain energy should be zero, but may be nonzero for any of the following reasons:

- Round-off error accumulation.
- Insufficient free body supports have been provided.
- Redundant free body supports have been provided.

The redundancy of the supports may be caused by improper use of the free body supports themselves or by the presence of single point or multipoint constraints that constrain the rigid body motions.

Static analysis with inertia relief is necessarily made on a model having at least one rigid body motion. Such rigid body motion must be constrained by the use of free body supports. These supported degrees-of-freedom define a reference system, and the elastic displacements are calculated relative to the motion of the support points. The element stresses and forces will be independent of any valid set of supports.

Rigid body vibration modes are calculated by a separate procedure provided that a set of free body supports is supplied by the user. This is done to improve efficiency and, in some cases, reliability. The determinant method, for example, has difficulty extracting zero frequency roots of high multiplicity, whereas the alternate procedure of extracting rigid body modes is both efficient and reliable. If the user does not specify free body supports (or he specifies an insufficient number of them), the (remaining) rigid body modes will be calculated by the method selected for the finite frequency modes, provided zero frequency is included in the range of interest. If the user does not provide free body supports, and if zero frequency is not included in the range of interest, the rigid body modes will not be calculated.

Free body supports must be specified if the mode acceleration method of solution improvement is used for dynamic problems having rigid body degrees-of-freedom (see "**Rigid-body Modes** " in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*). This solution improvement technique involves a static solution, and although the dynamic solution can be made on a free body, the static solution cannot be performed without removing the singularities in the stiffness matrix associated with the rigid body motions.

## B.5 Sets for Dynamic Reduction

There are several methods for reducing the size of models in dynamic analysis. The method described here uses shapes derived as generalized functions by static analysis. The same entries may be used in static analysis with the same nomenclature being used internally as in superelement analysis.

The statically independent degrees-of-freedom remaining after constraint elimination are partitioned into two sets. The partitions are defined by listing the degrees-of-freedom for one of the partitions on the ASETi entries. These degrees-of-freedom are referred to as the analysis set. The remaining degrees-of-freedom in the omitted set with the remaining degrees-of-freedom being placed in the analysis set. This is easier if the analysis set is large.

On of the more important applications of partitioning is Guyan reduction, described in "**Overview** " in the *Simcenter Nastran Basic Dynamic Analysis User's Guide*. This technique is a means for reducing the number of degrees-of-freedom used in dynamic analysis with minimum loss of accuracy. Its basis is that many fewer grid points are needed to describe the inertia of a structure than are needed to describe it

elasticity with comparable accuracy. The error in the approximation is small provided that the set of displacements used for dynamic analysis is judiciously chosen. Its members should be uniformly dispersed throughout the structure, and all large mass items should be connected to grid points that are members of the analysis set.

The use is cautioned to consider the fact that the matrix operations associated with this partitioning procedure tend to create nonzero terms and to fill what were previously very sparse matrices. The partitioning option is most effectively used if the members of the omitted set are either a very large fraction or a very small fraction of the total set. In most of the applications, the omitted set is a large fraction of the total and the matrices used for analysis, while small, are usually full. If the analysis set is not a small fraction of the total, a solution using the large, but sparser, matrices may be more efficient.

The a-set is further partitioned into the q-, r-, c-, and b-sets. The q-set is used to store the coefficients for the generalized coordinates determined by dynamic reduction and/or component mode calculations. The r-set is again used to determine rigid body modes. The c-set contains coordinates considered free to move while computing the generalized coordinate functions. The b-set contains coordinates fixed during this process.

The set used to describe the generalized functions for dynamic reduction and component mode synthesis is the v-set, the union of the o-, r-, and c-sets.

The d-set (dynamic) for direct formulations is formed from the union of the structural degrees-of-freedom in the a-set and the extra points of the e-set. The d-set can be used to input unsymmetrical terms into the equations of motion using DMIG and TF Bulk Data entries. In the modal formulations, the h-set is composed of the union of the $\xi_i$ (modal set and the e-set. Again, the DMIG and TF Bulk Data entries are a means to introduce unsymmetrical matrix coefficients.

## B.6 Sets for Aerodynamics

Aerodynamic calculations are made in a Cartesian aerodynamic coordinate system. By convention, the flow is in the positive x direction. The basic coordinate system was not chosen, since it would place an undesirable restriction upon the description of the structural model. You can specify any Nastran Cartesian system. Flow will be defined in the direction of its positive x-axis. All element and aerodynamic grid point data, computed initially in the basic coordinate system, will be converted to the aerodynamic coordinate system. The global (displacement) coordinate system of the aerodynamic grid points will have its T1 direction in the flow direction. T3 is normal to the element for boxes, and parallel to the aerodynamic T3 in the case of bodies. Coordinate system data are generated for the aerodynamic grid points.

The grid points are physically located at the centers of the boxes and body elements. Permanent constraints are generated for the unused degrees-of-freedom. A second set of grid points, used only for undeformed plotting, is located at the element corners. All six degrees-of-freedom associated with each grid point in this second set are permanently constrained. Grid point numbers are generated based upon the element identification number. For any panel, the external grid point numbers for the boxes start with the panel identification number and increase consecutively.

Aerodynamic degrees-of-freedom, along with the extra points, are added after the structural matrices and modes have been determined. This introduces the following displacement sets:

| | |
|---|---|
| **k** | Aerodynamic box and body degrees-of-freedom |
| **sa** | Permanently constrained degrees-of-freedom associated with aerodynamic grid points |
| **ps** | Union of the p-set (physical) and the sa-set |
| **pa** | Union of k-set and ps-set (physical and aerodynamic) |

## B.7 Rules of Sets for Undefined Degrees-of-Freedom

Most of the Bulk Data entries used to list set membership specify mutually exclusive sets. This means that degrees-of-freedom listed on more than one such entry will result in a fatal error. For example, a degree-of-freedom may not be in the m-set on two RBAR entries, or on a selected SPC entry and a SUPORT entry.

The a-set, by contrast, defines a combined set. Degrees-of-freedom listed on Bulk Data entries that define its subsets, the q-, r-, c-, and b-set, may also be listed on ASETi Bulk Data entries, although this is a redundant specification and is not necessary. However, automatic restarts are more reliable if ASETi entries are present for all degrees-of-freedom listed on SUPORT entries.

Bulk Data entries used to define mutually exclusive sets automatically place them in their specified set and any combined set to which they belong. Since the ASETi entries define a combined set, the subsets of the a-set, if not defined, must be placed in a mutually exclusive set by an arbitrary convention. The conventions described below are used because they allow the user to specify these sets with the minimum input for most cases. In superelement analysis, exterior (boundary) degrees-of-freedom are automatically determined and placed in the a-set of the superelement. Exterior degrees-of-freedom for superelements are therefore equivalent to degrees-of-freedom listed on ASETi entries in the discussion below.

The f-set is the union of the mutually exclusive o-, q-, r-, c-, and b-sets and includes the a-set. The o-set exists under two conditions only:

1. If OMITi entries are present, the o-set consists of degrees-of-freedom listed explicitly on OMITi entries. The remaining f-set degrees-of-freedom are placed in the b-set. They are then also members of the a-set.

2. If ASETi or QSETi entries are present, the a-set consists of all degrees-of-freedom listed on ASETi entries and any entries listing its subsets, such as QSETi, SUPORTi, CSETi, and BSETi entries. The remaining f-set degrees-of-freedom are placed in the o-set. Note, however, there must be at least one explicit ASETi or QSETi entry type present for the o-set to exist, even if it is a redundant specification. If there are no ASETi or QSETi entries present, and SUPORT, BSETi, or CSETi are present, the entire f-set is placed in the a-set; that is, the o-set is not formed.

If both OMITi and ASETi entries are present, the OMITi entries are ignored, so that condition 2, above, applies. Similarly, if ASETi entries are used in conjunction with entries defining its subsets, undefined

degrees-of-freedom in the a-set are placed in mutually exclusive sets as shown in the following table (0 = no entry types present; 1 = entry type present):

| Set Defined by Entry Type | Case | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| c | 0 | 0 | 1 | 1 |
| b | 0 | 1 | 0 | 1 |
| Undefined a-set placed in | b | c | b | b |

## B.8 Output Selection via Set Specification

The membership of grid points and their components and scalar points in the various sets is stored in the USET table as digits in a binary number. Some modules use input parameters that are translations of these binary numbers as a shorthand notation. Three formats are used:

### Set Name

(DEFUSET, USET and USET1 Bulk Data entries, MATGPR, UPARTN, VEC, UMERGE, UMERGE1, and PARAML (OP = USET) Modules)

The name of the set from the table below (which may be redefined by the DEFUSET entry).

### Decimal Equivalent

(TABPRT, MATMOD (Option 17), and MATGEN (Option 11) Modules)

The decimal equivalent of the set is listed below. Several sets can be described by a single number that is the sum of the listed equivalents. For example, to print the m and s sets, the value of 3(1+2) is input to the TABPRT module. The value of -1 will cause the output of all sets.

### Bit Position

(VEC Module)

The location of the digit corresponding to the set in the binary number, starting with the left-most digit, can be used to identify one set, as listed in **Table B-10**.

**Table B-10. USET Table: Set Name, Decimal Equivalent, and Bit Position**

| Set Name | Decimal Equivalent | Bit Position |
|---|---|---|
| U1 | -- | 1 |
| U2 | -- | 2 |

| Set Name | Decimal Equivalent | Bit Position |
|---|---|---|
| U3 | -- | 3 |
| U4 | -- | 4 |
| U5 | -- | 5 |
| U6 | -- | 6 |
| V | 33554432 | 7 |
| FR | 16777216 | 8 |
| T | 8388608 | 9 |
| O | 4194304 | 10 |
| B | 2097152 | 11 |
| C | 1048576 | 12 |
| PA | 524288 | 13 |
| K | 262144 | 14 |
| SA | 131072 | 15 |
| PS | 65536 | 16 |
| D | 32768 | 17 |
| FE | 16384 | 18 |
| NE | 8192 | 19 |
| P | 4096 | 20 |
| E | 2048 | 21 |
| SB | 1024 | 22 |
| SG | 512 | 23 |
| L | 256 | 24 |
| A | 128 | 25 |
| F | 64 | 26 |
| N | 32 | 27 |
| G | 16 | 28 |
| R | 8 | 29 |
| O | 4 | 30 |
| S | 2 | 31 |
| M | 1 | 32 |
| H,J,... | -- | -- |

The data block naming convention in the solution sequences for matrices related to the sets is based on the set names. For example, the system stiffness matrix $[K_{gg}]$ has the name KGG. Some sets are used that do not appear in the USET table. The h-set, for example, is the set used for modal formulations. Its

size is not known until after the USET table is formed, so it does not appear in the table. The j-set, a set synonymous with the g-set, is used in superelement analysis. There are several other sets used that also do not appear in the USET table.

Some names may be used for other purposes. The name H, for example, may be input to the MATGPR module when one wants to label rows or columns of matrices sequentially, rather than by their external sequence numbers. This feature may be used to number vectors in load matrices or in other instances not related to modal analysis, even though H is defined for modal analysis.

# C. Selected Bibliography


1. Miura, H., *MSC/NASTRANHandbookforStructuralOptimization*, The MacNeal-Schwendler Corporation, 1988.

2. Moore, G.J. *MSC/NASTRANDesignSensitivityandOptimizationUser'sGuide*, Version 67, The MacNeal-Schwendler Corporation, 1992.

3. Moore, G.J. and Nagendra, G.K., "Dynamic Response Sensitivities in MSC/NASTRAN" Proceedings of the MSC 1991 World User's Conference, Paper No. 4, March 1991.

4. Patel, H., "Multidisciplinary Design with Superelements in MSC/NASTRAN," AIAA CP-92-4732, Fourth AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, Cleveland, Ohio, September1992.

5. Johnson, E.H., and Reymond, M.A., "Multidisciplinary Aeroelastic Analysis and Design using MSC/NASTRAN," AIAA CP-91-1097, 32nd SDM Conference, Baltimore, MD, April 1991.

6. Kodiyalam, S., Vanderplaats, G., Miura, H., Nagendra, G. and Wallerstein, D. "Structural Shape Optimization with MSC/NASTRAN," AIAA CP 90-1007, 31st SDM Conference, Long Beach, CA, April 1990.

7. Johnson, E.H., "Disjoint Design Spaces in the Optimization of Harmonically Excited Structures," AIAA Journal, Vol. 14, No. 2, February 1976, pp 259-261.

8. "Introduction to Version 67 of MSC/NASTRAN," The MacNeal-Schwendler Corporation, September 1991.

9. Cassis, J.H., "Optimum Design of Structures Subjected to Dynamic Loads," PhD. Thesis UCLA-ENG-7451, June 1974.

# Siemens Digital Industries Software


## Headquarters
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

## Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

## Europe
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

## Asia-Pacific
Suites 4301-4302, 43/F
AIA Kowloon Tower, Landmark East
100 How Ming Street
Kwun Tong, Kowloon
Hong Kong
+852 2230 3308


## About Siemens Digital Industries Software

Siemens Digital Industries Software is a leading global provider of product life cycle management (PLM) software and services with 7 million licensed seats and 71,000 customers worldwide. Headquartered in Plano, Texas, Siemens Digital Industries Software works collaboratively with companies to deliver open solutions that help them turn more ideas into successful products. For more information on Siemens Digital Industries Software products and services, visit **www.siemens.com/plm**.