STIFFNESS-GENERATED RIGID-BODY MODE SHAPES FOR LANCZOS EIGENSOLUTION WITH SUPORT DOF VIA A MSC/NASTRAN DMAP ALTER

Ayman A. Abdallah, Alan R. Barnett, and Timothy W. Widrick
Analex Corporation
3001 Aerospace Parkway
Brook Park, Ohio 44142

Richard T. Manella and Robert P. Miller
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

Abstract

When using all MSC/NASTRAN eigensolution methods except Lanczos, the analyst can replace the coupled system rigid-body modes calculated within DMAP module READ with mass orthogonalized and normalized rigid-body modes generated from the system stiffness. This option is invoked by defining MSC/NASTRAN r-set degrees-of-freedom via the SUPORT Bulk Data card. The newly calculated modes are required if the rigid-body modes calculated by the eigensolver are not "clean" due to numerical roundoffs in the solution. When performing transient structural dynamic load analyses, the numerical roundoffs can result in inaccurate rigid-body accelerations which affect steady-state responses. Unfortunately, when using the Lanczos method and defining r-set degrees-of-freedom, the rigid-body modes calculated within DMAP module REIGL are retained. To overcome this limitation and to allow MSC/NASTRAN to handle SUPORT degrees-of-freedom identically for all eigensolvers, a DMAP Alter has been written which replaces Lanczos-calculated rigid-body modes with stiffness-generated rigid-body modes. The newly generated rigid-body modes are normalized with respect to the system mass and orthogonalized using the Gram-Schmidt technique. This algorithm has been implemented as an enhancement to an existing coupled loads methodology.

1994 MSC World Users' Conference
Lake Buena Vista, Florida
June 20-24, 1994

STIFFNESS-GENERATED RIGID-BODY MODE SHAPES FOR LANCZOS EIGENSOLUTION WITH SUPORT DOF VIA A MSC/NASTRAN DMAP ALTER

Ayman A. Abdallah, Alan R. Barnett, and Timothy W. Widrick
Analex Corporation
3001 Aerospace Parkway
Brook Park, Ohio 44142

Richard T. Manella and Robert P. Miller
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

Nomenclature

Abbreviations

DOF	Degrees-of-freedom
DMAP	Direct Matrix Abstraction Program
LeRC	Lewis Research Center
NASA	National Aeronautics and Space Administration

Matrices

F	Constraint forces
I	Identity
K	Stiffness
M	Mass
X	Orthogonality check matrix as computed using new r-set rigid-body modes
Y	Orthogonality check matrix as computed using new x-set rigid-body modes
Z	Cross-orthogonality check matrix
α	Scaling factor for normalizing mode shapes
ϕ and Φ	Mode shape vector and matrix
ω^2	System eigenvalue

Set Notation

a	a-set (analysis DOF)
r	r-set (DOF defined by SUPORT card)
x	x-set (DOF in a-set having non-zero masses)
y	y-set (complement of r-set in x-set)

Superscripts

el	elastic
rb	rigid-body

Introduction

In general, to perform a modal transient loads analysis on a space structure, the system frequencies and mode shapes are needed; and hence, the solution of the eigenvalue problem is required. For undamped systems, the computed eigenvalues correspond to the square of the natural circular frequencies (ω^2), and the computed eigenvectors correspond to the normal modes of vibration (mode shapes ϕ). The Givens, Householder, Inverse Power, and Lanczos methods are real eigensolution methods provided by MSC/NASTRAN [1]. While the Givens, Householder, and Inverse Power methods are executed within the DMAP module READ, the Lanczos method is executed within the DMAP module REIGL. Although the Givens and Householder methods are suitable for small size problems, they are not efficient for medium to large size problems which do not fit into the computer memory. In these cases a reduction procedure, such as Guyan reduction [2] or generalized dynamic reduction (GDR) [1], can be used prior to solving the eigenvalue problem. When combined with these reduction methods, the Givens or Householder eigensolvers are very efficient in solving the eigenproblem of large models. Unfortunately, they can produce poor results and can miss modes [3]. The Inverse Power methods are very suitable for large problems if only a small number of modes are required; however, they are not efficient if more than a few modes are needed [3]. The Lanczos method is the most accurate and efficient method that can be used to solve the eigenvalue problem for medium to large size models. For most aerospace applications, the finite element model of a substructure (e.g. a payload or a launch vehicle) is composed of a large number of DOF. Thus, the Lanczos method is the preferred eigensolver for these types of structures.

Generally, aerospace systems are unconstrained (e.g. a launch vehicle flying after liftoff) and are considered free-free systems. For a free-free system, there are r number of rigid-body mode shapes which have zero value frequencies. The number of rigid-body mode shapes, r, is equal to the number of statically determinate constraints required to prevent all rigid-body motion. Due to numerical roundoffs, the frequencies computed by any eigensolver are not perfect 0.0 Hz frequencies, and the computed mode shapes are not "clean" rigid-body modes. This can result in inaccurate rigid-body modal accelerations which affect the steady-state responses in a transient solution.

In order to overcome this problem, an analyst can use SUPORT Bulk Data cards with the Givens, Householder, and Inverse Power methods. The SUPORT card entry forces the solver to replace the computed rigid-body frequencies with "perfect" 0.0 Hz values. Also, the r-set DOF for the residual structure, defined via the SUPORT card, are used to generate rigid-body modes from the system stiffness matrix. The stiffness-generated rigid-body modes are orthogonalized and normalized with respect to the system mass matrix, and since they are "cleaner", they replace the rigid-body modes computed from the system eigenvalue problem. Unfortunately, *when using SUPORT cards with the Lanczos method, the rigid-body modes calculated within DMAP module REIGL are retained.* The only effect of the SUPORT card in the Lanczos method is that it attempts to output 0.0 Hz frequencies for the computed rigid-body modes without changing the originally computed eigenvectors [3]. When a SUPORT card is defined, the Lanczos solver inspects the values of the computed frequencies for the first r modes. If any frequency value is near 0.0, it is replaced by a "perfect" 0.0 value; otherwise, the computed frequency is retained.

The objective of this work is to allow MSC/NASTRAN to handle SUPORT DOF similarly for all eigensolvers. In order to meet this objective, a DMAP Alter has been written which replaces Lanczos-calculated rigid-body modes with stiffness-generated rigid-body modes. The newly generated rigid-body modes are normalized with respect to the system mass and orthogonalized using the Gram-Schmidt technique. Also, the frequencies of the first r modes output by the Lanczos solver are replaced with "perfect" 0.0 Hz values. This DMAP Alter has been implemented as an enhancement to an existing coupled loads methodology at the NASA Lewis Research Center.

The theory behind calculating system level rigid-body modes using the system stiffness, normalizing the modes with respect to the system mass, and orthogonalizing the modes via the Gram-Schmidt method is presented in the following section. In the subsequent section, implementation of the method within a DMAP Alter is described, as are the internal checking procedures available in the new Alter. Lastly, the new Alter is tested through a real-world engineering problem. The results of this example problem and the checking procedure results are presented in the last section.

Theory

The Alter to generate system rigid-body modes using the system stiffness and enable the consistent utilization of SUPORT DOF with the MSC/NASTRAN Lanczos eigensolver is relatively straight forward. In Solution 63 of MSC/NASTRAN (normal modes for superelements), the system eigensolution is performed on the x-set DOF. These DOF are either the system analysis (a-set) DOF if all a-set DOF have mass associated with them, or the x-set DOF are a subset of the a-set DOF where only those DOF with mass associated with them have been retained. Given this, the coupled system eigenproblem is

$$\left[[K_{xx}] - \omega_i^2[M_{xx}]\right]\{\phi_x^i\} = \{0_x\} \tag{1}$$

where $[M_{xx}]$ and $[K_{xx}]$ are the system x-set DOF mass and stiffness matrices, and ω_i^2 and $\{\phi_x^i\}$ are the ith system eigenvalue and normal mode. Assuming for a particular cutoff frequency the system has *r* number of rigid-body (rb) modes and *e* number of elastic (el) modes (*h* number of total modes), the matrix of calculated system modes, $[\Phi_{xh}]$, is written as

$$[\Phi_{xh}] = \left[\{\phi_x^{rb_1}\}\ \{\phi_x^{rb_2}\} \cdots \{\phi_x^{rb_r}\}\ \{\phi_x^{el_1}\}\ \{\phi_x^{el_2}\} \cdots \{\phi_x^{el_e}\}\right] \tag{2}$$

or

$$[\Phi_{xh}] = \left[[\Phi_{xr}^{old}]\ [\Phi_{xe}]\right] \tag{3}$$

where the first *r* modes, $[\Phi_{xr}^{old}]$, are the system rigid-body modes calculated via the eigensolver. As mentioned in the introduction, generally these modes as calculated by the eigensolver have non-zero frequencies due to small numerical roundoffs. In all eigensolvers except Lanczos, these rigid-body modes are replaced by "clean" rigid-body modes if a SUPORT card defining r-set DOF is present in the Bulk Data deck. Hence, the system modes including the new rigid-body modes are written as

$$[\Phi_{xh}] = \left[[\Phi_{xr}^{new}]\ [\Phi_{xe}]\right] \tag{4}$$

where $[\Phi_{xr}^{new}]$ are the system rigid-body modes generated using the system stiffness, normalized and orthogonalized with respect to the system mass. The purpose of the new Alter is to replace the rigid-body modes calculated by Lanczos with "clean" rigid-body modes generated using the system stiffness and corresponding to *r* number of r-set DOF defined via a SUPORT card.

To generate new rigid-body modes with respect to the r-set DOF using the system stiffness, the system stiffness matrix $[K_{xx}]$ is first partitioned according to r-set and complement (hereafter, referred to as y-set) DOF

$$[K_{xx}] = \begin{bmatrix} [K_{rr}] & [K_{ry}] \\ [K_{yr}] & [K_{yy}] \end{bmatrix} \tag{5}$$

Next, unit motion is applied to the partitioned stiffness matrix at the r-set DOF. Since the r-set is statically determinate, the system should move in a rigid-body manner when unit displacements are applied sequentially to each r-set DOF while holding the rest and allowing the y-set DOF to move freely [4]. The system equilibrium equation is

$$\begin{bmatrix} [K_{rr}] & [K_{ry}] \\ [K_{yr}] & [K_{yy}] \end{bmatrix} \begin{bmatrix} [I_{rr}] \\ [\bar{\Phi}_{yr}^{new}] \end{bmatrix} = \begin{bmatrix} [F_{rr}] \\ [0_{yr}] \end{bmatrix} \tag{6}$$

where $[F_{rr}]$ are constraint forces required to move the system in a rigid-body manner (these forces should be computed zeros). From the lower partition of Eq. (6), the un-normalized, un-orthogonalized rigid-body modes for the y-set DOF are calculated as constraint modes [4]

$$[\bar{\Phi}_{yr}^{new}] = -[K_{yy}]^{-1}[K_{yr}] \tag{7}$$

Hence, the un-normalized and un-orthogonalized x-set system rigid-body modes are

$$[\bar{\Phi}_{xr}^{new}] = \begin{bmatrix} [\ I_{rr}\ ] \\ [\bar{\Phi}_{yr}^{new}] \end{bmatrix} \tag{8}$$

where $[I_{rr}]$ is an identity matrix of order r which represents the unit translations (and/or rotations) applied at the r-set DOF to generate the rigid-body modes.

To orthogonalize and normalize the computed rigid-body modes with respect to the mass matrix, first the system rigid-body mass matrix is calculated as

$$[M_{rr}^{rb}] = [\bar{\Phi}_{xr}^{new}]^T[M_{xx}][\bar{\Phi}_{xr}^{new}] \tag{9}$$

For each rigid-body mode, the approach taken is to first orthogonalize the mode to all previous rigid-body modes and then normalize it with respect to the system mass given in Eq. (9). Obviously, the first rigid-body mode needs only to be mass normalized. All other modes are orthogonalized with respect to the previously calculated modes. The procedure for orthogonalizing and normalizing is performed on the r-set partition of the rigid-body modes. This is much more efficient than performing the operations on the full x-set DOF modes. After orthogonalization and normalization of the r-set DOF modes, the rigid-body modes are expanded to the x-set DOF.

Let the r-set partition of the un-orthogonalized and un-normalized system rigid-body mode shapes of Eq. (8) (an identity matrix) be written as

$$[\bar{\Phi}_{rr}^{new}] = [I_{rr}] = \left[\{\bar{\phi}_r^1\}\ \{\bar{\phi}_r^2\} \cdots \{\bar{\phi}_r^r\}\right] \tag{10}$$

For the first rigid-body mode, only mass normalization is required. Hence,

$$\{\phi_r^1\} = \alpha_1\{\bar{\phi}_r^1\} \tag{11}$$

where the scaling factor α_1 is calculated as

$$\alpha_1 = \frac{1}{\sqrt{\{\bar{\phi}_r^1\}^T[M_{rr}^{rb}]\{\bar{\phi}_r^1\}}} \tag{12}$$

Therefore, $\{\phi_r^1\}$ is the first rigid-body mode normalized with respect to the system mass.

For each of the remaining *r-1* number of rigid-body modes, the following procedure is performed. For a given jth mode, the mode is first orthogonalized to the previous *j-1* number of modes using the Gram-Schmidt orthogonalization scheme [5], or

$$\{\tilde{\phi}_r^j\} = \{\bar{\phi}_r^j\} - \sum_{k=1}^{j-1}\left(\{\bar{\phi}_r^j\}^T[M_{rr}^*]\{\phi_r^k\}\right)\{\phi_r^k\} \tag{13}$$

After orthogonalization of the jth mode, it is normalized with respect to the system mass as

$$\{\phi_r^j\} = \alpha_j\{\tilde{\phi}_r^j\} \tag{14}$$

where the scaling factor α_j is calculated as

$$\alpha_j = \frac{1}{\sqrt{\{\tilde{\phi}_r^j\}^T[M_{rr}^*]\{\tilde{\phi}_r^j\}}} \tag{15}$$

Given that all system rigid-body modes have been orthogonalized and normalized with respect to the system mass for all r-set DOF, the final modes corresponding to the r-set DOF are written as

$$[\Phi_{rr}^{new}] = \left[\{\phi_r^1\}\ \{\phi_r^2\} \cdots \{\phi_r^r\}\right] \tag{16}$$

This matrix can be considered a scaling matrix required to linearly transform the rigid-body modes given by Eq. (8) to form the final orthogonalized and mass normalized x-set DOF rigid-body modes. The final new system rigid-body modes expanded to the x-set DOF are

$$[\Phi_{xr}^{new}] = [\bar{\Phi}_{xr}^{new}][\Phi_{rr}^{new}] = \begin{bmatrix}[\Phi_{rr}^{new}]\\ [\Phi_{yr}^{new}]\end{bmatrix} \tag{17}$$

where

$$[\Phi_{yr}^{new}] = [\bar{\Phi}_{yr}^{new}][\Phi_{rr}^{new}] \tag{18}$$

To prove that the final computed x-set rigid-body modes are mass orthogonalized and normalized, consider the following triple matrix multiplication

$$[\Phi_{xr}^{new}]^T\ [M_{xx}]\ [\Phi_{xr}^{new}] \tag{19}$$

Substituting Eq. (17) into (19)

$$\begin{aligned}[\Phi_{xr}^{new}]^T\ [M_{xx}]\ [\Phi_{xr}^{new}] &= \left([\bar{\Phi}_{xr}^{new}][\Phi_{rr}^{new}]\right)^T [M_{xx}] \left([\bar{\Phi}_{xr}^{new}][\Phi_{rr}^{new}]\right)\\ &= [\Phi_{rr}^{new}]^T\left([\bar{\Phi}_{xr}^{new}]^T[M_{xx}][\bar{\Phi}_{xr}^{new}]\right)[\Phi_{rr}^{new}]\end{aligned} \tag{20}$$

The middle term between parentheses in Eq. (20) is the rigid-body mass matrix given by Eq. (9), hence

$$[\Phi_{xr}^{new}]^T \, [M_{xx}] \, [\Phi_{xr}^{new}] \; = \; [\Phi_{rr}^{new}]^T \, [M_{rr}^{*}] \, [\Phi_{rr}^{new}] \tag{21}$$

Since the r-set rigid-body modes given by Eq. (16) were orthogonalized and normalized with respect to the rigid-body mass matrix using the Gram-Schmidt orthogonalization scheme, the right hand side of Eq. (21) is equal to an identity matrix. Therefore Eq. (21) becomes:

$$[\Phi_{xr}^{new}]^T \, [M_{xx}] \, [\Phi_{xr}^{new}] \; = \; [\, I_{rr}] \tag{22}$$

This proves the final system x-set rigid-body modes given by Eq. (17) are mass orthogonalized and normalized.

Implementation

In order to replace system rigid-body modes calculated using the Lanczos eigensolver with rigid-body modes generated using the system stiffness, a new DMAP Alter has been implemented in the NASA LeRC coupled loads methodology. This Alter to MSC/NASTRAN Solution 63 is included in the Appendix.

The following are the alterations made to Solution 63:

- Prepare required data blocks and replace the solved-for rigid-body eigenvalues with perfect zeros. This is performed first by transforming the LAMA table to a matrix and then replacing the rigid-body frequency entries with zero values. The discarded rigid-body frequencies output by Lanczos are printed for inspection.
- Use residual USET table to form a partition vector, VXCOMPRN, for partitioning out r-set DOF from x-set DOF.
- Calculate system rigid-body modes from system stiffness as given by Eqs. (7) and (8). Generate system rigid-body mass matrix, as defined by Eq. (9), using new modes and print as a check.
- Normalize first new system rigid-body mode with respect to system mass as shown in Eqs. (11) and (12).
- Use the DO WHILE DMAP statement to loop over the rest of the new system rigid-body modes. Orthogonalize each subsequent mode with all previous modes using Gram-Schmidt method (Eq. (13)) and normalize each mode with respect to system mass (Eqs. (14) and (15)).
- Expand new system rigid-body modes from r-set DOF to x-set DOF (Eqs. (17) and (18)).
- Check orthogonality of new system rigid-body modes if required by the analyst via new parameter CHECKRBM. The checks performed are for the r-set DOF rigid-body modes:

  $$[\Phi_{rr}^{new}]^T [M_{rr}^{*}] [\Phi_{rr}^{new}] \; = \; [X_{rr}] \tag{23}$$

  and for the expanded x-set DOF rigid-body modes:

  $$[\Phi_{xr}^{new}]^T [M_{xx}] [\Phi_{xr}^{new}] \; = \; [Y_{rr}] \tag{24}$$

where $[X_{rr}]$ and $[Y_{rr}]$ matrices are printed for inspection. Theoretically, these matrices are equal to the identity matrix $[I_{rr}]$.

Replace old system rigid-body modes in PHIX datablock, generated by REIGL DMAP module, with new system rigid-body modes.

With that, the system modes are ready for further operations within Solution 63.

Numerical Example

For a launch vehicle/spacecraft coupled system, rigid-body mode shapes were computed using three different methods with the same SUPORT DOF defined. The three methods used were Lanczos, Inverse Power, and Lanczos with the new Alter. The order of the x-sized matrices for which the eigensolutions were performed was equal to 15,616. The purpose of solving the eigenproblem with three different methods was to illustrate the following:

1- Rigid-body mode shapes output by the Lanczos solver (without the new Alter) are different than those output by other solvers (Inverse Power without the new Alter) when the same SUPORT DOF are defined.

2- Stiffness-generated mass orthogonalized and normalized rigid-body modes are accurately computed by the new Alter when used with the Lanczos solver with SUPORT DOF defined.

3- Rigid-body mode shapes output by the Lanczos solver (with the new Alter) are similar to those output by other solvers (Inverse Power without the new Alter) when the same SUPORT DOF are defined.

To show that the rigid-body mode shapes computed by Lanczos (without the Alter) are different than those computed by Inverse Power (without the Alter) with the same SUPORT DOF defined, cross-orthogonality checks were used. The cross-orthogonality matrix $[W_{rr}]$ was computed as:

$$[\Phi_{xr}^{SINV}]^T [M_{xx}] [\Phi_{xr}^{\text{Lanczos without Alter}}] = [W_{rr}] \tag{25}$$

where the SINV superscript denotes the Inverse Power method. As shown in Table (1), the result of this check is not an identity matrix which demonstrates the difference between the rigid-body modes output by both eigensolvers.

To determine whether or not the new Alter properly mass normalized and orthogonalized the SUPORTed rigid-body modes, the orthogonality checks shown by Eqs. (19) and (20) were performed. Results were excellent and are shown in Tables (2) and (3).

To determine whether or not the Alter generated SUPORTed rigid-body modes (Lanczos with Alter) equal to those generated using standard MSC/NASTRAN (Inverse Power without Alter), the former were compared with the latter via cross-orthogonality checks and direct comparisons. The cross-orthogonality matrix was computed as:

$$[\Phi_{xr}^{SINV}]^T [M_{xx}] [\Phi_{xr}^{\text{Lanczos with Alter}}] = [Z_{rr}] \tag{26}$$

The result of this check, shown in Table (4), is an identity matrix which shows the Alter properly generates SUPORTed rigid-body modes as compared to standard MSC/NASTRAN. The rigid-body modes were then subtracted from one another in a direct comparison as

$$[\Phi_{xr}^{SINV}] - [\Phi_{xr}^{\text{Lanczos without Alter}}] = [0_{xr}] \tag{27}$$

Again, the results were excellent, where the largest term in the difference was on the order of 10^{-13} (calculated zero). Given these comparison results on a real-world engineering problem, it has been shown that the new Alter performs as required and replicates the operations performed by MSC/NASTRAN for all other eigensolution methods when SUPORT DOF are defined.

Conclusion

In order to enable MSC/NASTRAN to handle SUPORT DOF with the Lanczos method in a similar manner to the other eigensolvers, a Solution 63 DMAP Alter has been written as an enhancement to the NASA LeRC coupled loads methodology. The new Alter replaces the rigid-body modes generated by DMAP module REIGL with rigid-body modes computed from the system stiffness matrix. The newly generated rigid-body modes are orthogonalized and normalized with respect to the system mass matrix using the Gram-Schmidt orthogonalization scheme. In addition, the rigid-body frequencies output by the Lanczos solver are replaced with perfect 0.0 Hz values. A real-world engineering problem was used to test the accuracy of the newly developed Alter. Rigid-body modes were calculated using Lanczos (with the Alter) and Inverse Power (without the Alter) methods with the same SUPORT DOF defined. It has been shown, through a comparison of the generated rigid-body modes, that using the Alter with the Lanczos method produces very accurate results.

References

[1] *MSC/NASTRAN Handbook for Numerical Methods,* Version 66, The MacNeal-Schwendler Corporation, Los Angeles, CA, 1990.

[2] Guyan, R. J.: "Reduction of Stiffness and Mass Matrices," *AIAA Journal,* Vol. 3, No. 2, February 1965, p. 380.

[3] MSC/NASTRAN: *"Common Questions and Answers for Version 67.5,"* The MacNeal-Schwendler Corporation, Los Angeles, CA, 1993.

[4] Craig, R. R., Jr. and Bampton, M. C. C.: "Coupling of Substructures for Dynamic Analysis," *AIAA Journal,* Vol. 6, No. 7, July 1968, pp. 1313-1319.

[5] Bathe, K. J.: *"Finite Element Procedures in Engineering Analysis,"* Prentice-Hall, 1982.

$[W_{\pi}]$ =

-6.9215e-01	2.0783e-03	5.8888e-03	1.9600e-03	7.2172e-01	-9.1999e-04
5.9411e-04	6.8867e-01	-2.4478e-03	7.2361e-01	-3.4169e-03	-4.5826e-02
7.6239e-03	1.5092e-03	9.9997e-01	1.8613e-03	-8.5851e-04	-1.1783e-03
4.1390e-02	2.0442e-02	7.8191e-04	4.3854e-02	4.0781e-02	9.9714e-01
-7.2033e-01	1.6575e-02	4.9701e-03	-1.4726e-02	-6.9079e-01	5.8456e-02
1.6714e-02	7.2459e-01	9.1000e-05	-6.8865e-01	1.5830e-02	1.4091e-02

Table 1. -- Cross-orthogonality check computed using SUPORTed rigid-body modes generated using Lanczos without the new Alter vs. Inverse Power

$[X_{\pi}]$ =

1.0000e+00	1.0842e-19	-2.7105e-20	-2.1684e-19	-3.5527e-15	-8.3267e-17
1.0842e-19	1.0000e+00	0.0000e+00	0.0000e+00	-5.4210e-20	1.1102e-16
-2.7105e-20	0.0000e+00	1.0000e+00	-1.0408e-17	5.5511e-17	6.1240e-19
-2.1684e-19	0.0000e+00	-1.0408e-17	1.0000e+00	1.3553e-19	-2.2204e-16
-3.5527e-15	-5.4210e-20	5.5511e-17	1.3553e-19	1.0000e+00	5.5511e-17
-8.3267e-17	1.1102e-16	6.1240e-19	-2.2204e-16	5.5511e-17	1.0000e+00

Table 2. -- Orthogonality check computed using r-set size SUPORTed rigid-body modes generated using new Alter.

$[Y_{\pi}]$ =

1.0000e+00	-1.4993e-14	-5.0910e-16	3.8540e-15	-2.1588e-12	-1.3701e-14
-1.4993e-14	1.0000e+00	-7.1370e-17	7.0031e-13	6.0921e-14	2.8727e-14
-5.0910e-16	-7.1370e-17	1.0000e+00	2.6143e-15	1.6721e-15	5.4753e-15
3.8540e-15	7.0031e-13	2.6143e-15	1.0000e+00	-4.0034e-14	1.9734e-14
-2.1588e-12	6.0921e-14	1.6721e-15	-4.0034e-14	1.0000e+00	9.0601e-14
-1.3701e-14	2.8727e-14	5.4753e-15	1.9734e-14	9.0601e-14	1.0000e+00

Table 3. -- Orthogonality check computed using x-set size SUPORTed rigid-body modes generated using new Alter.

$$
[Z_{rr}] = \begin{bmatrix}
1.0000e{+}00 & 1.1332e{-}14 & -9.5581e{-}17 & -1.6161e{-}14 & -2.0192e{-}12 & -2.1386e{-}14 \\
3.0538e{-}15 & 1.0000e{+}00 & -5.1448e{-}16 & 7.5368e{-}13 & 1.1361e{-}13 & -1.4128e{-}14 \\
-2.7188e{-}16 & -1.6059e{-}16 & 1.0000e{+}00 & 1.0776e{-}15 & -2.4310e{-}16 & 5.3789e{-}15 \\
-6.2457e{-}15 & 3.9230e{-}14 & -2.4279e{-}16 & 1.0000e{+}00 & -1.0624e{-}13 & 3.9857e{-}14 \\
-3.9616e{-}13 & -7.0599e{-}14 & -2.5558e{-}15 & 1.8957e{-}14 & 1.0000e{+}00 & 3.7061e{-}14 \\
3.6614e{-}14 & -3.6741e{-}14 & 2.2042e{-}16 & -8.1182e{-}15 & 4.4146e{-}14 & 1.0000e{+}00
\end{bmatrix}
$$

Table 4. -- Cross-orthogonality check computed using SUPORTed rigid-body modes generated using Lanczos with the new Alter vs. Inverse Power

Appendix

```
$  LANCZOS EIGENSOLUTION WITH PROPER SUPORT OF R-B MODES       (RF63D401)
$-----------------------------------------------------------------------
$  RIGID FORMAT 63 - NORMAL MODES WITH SUPERELEMENTS
$  MSC/NASTRAN VERSION 67
$
$
$  THIS DMAP ALTER IS USED TO GENERATE RIGID-BODY MODES USING SUPORT
$  DOF WITH THE LANCZOS EIGENSOLUTION METHOD.  THE CALCULATED RIGID-
$  BODY MODES GENERATED WITHIN REIGL ARE DISCARDED AND REPLACED WITH
$  RIGID-BODY MODES GENERATED FROM THE SYSTEM STIFFNESS.  THESE NEW
$  MODES ARE MASS NORMALIZED AND ORTHOGONALIZED USING THE GRAM-SCHMIDT
$  METHOD.
$
$
$  REQUIREMENTS TO USE THIS DMAP SEQUENCE -
$-----------------------------------------------------------------------
$  FILE MANAGEMENT SECTION -
$
$    NO SPECIAL REQUIREMENTS
$-----------------------------------------------------------------------
$  EXECUTIVE CONTROL DECK -
$
$    DIAG 8,14,20 RECOMMENDED
$    SOL 63
$    COMPILE SOL63,SOUIN=MSCSOU,NOLIST,NOREF
$    INCLUDE THIS ALTER
$-----------------------------------------------------------------------
$  CASE CONTROL DECK -
$
$    NO SPECIAL REQUIREMENTS
$-----------------------------------------------------------------------
$  BULK DATA DECK -
$
$    THE FOLLOWING PARAMETERS MAY BE DEFINED -
$
$    PARAM,CHECKRBM - IF .GE. 0, RIGID-BODY MASS AND ORTHOGONALITY
$     (OPTIONAL)      MATRICES ARE GENERATED AND PRINTED AND CAN BE USED
$                     TO CHECK THE NEW RIGID-BODY MODES.
$                     (DEFAULT = -1 - DO NOT PERFORM CHECKS)
$-----------------------------------------------------------------------
$  EXAMPLE NASTRAN DECK -
$
$    ID        LANCZOS,SUPORT
$    SOL       63
$    TIME      30
$    DIAG 8,14,20
$    COMPILE SOL63,SOUIN=MSCSOU,NOLIST,NOREF
$    INCLUDE RF63D401.V67
$    CEND
$    TITLE    = SYSTEM MODES
$    SUBTITLE = LANCZOS WITH SUPORT CARDS
$    $
$    SET 1000 = 0
$    SEALL    = 1000
$    $
$    DISP(PLOT) = ALL
$    $
$    SUBCASE 1
$      SUPER = 200
$      LABEL = SPACECRAFT
$      METHOD = 100
```

```
$     SUBCASE 2
$       SUPER = 100
$       LABEL = LAUNCH VEHICLE
$       METHOD = 100
$     SUBCASE 3
$     $ SUPER = 0
$       LABEL = COUPLED SYSTEM
$       METHOD = 50
$     $
$     BEGIN BULK
$     .
$     .  STANDARD BULK DATA INCLUDING SUPORT CARDS FOR RIGID-BODY MODES
$     .
$     $
$     $  PERFORM CHECKS ON NEW RIGID-BODY MODES
$     $
$     PARAM,CHECKRBM,+1
$     $
$     ENDDATA
$-----------------------------------------------------------------------
$   HISTORY DOCUMENTATION -
$
$     VERSION 1.0   01-OCT-93   A. ABDALLAH, A. BARNETT, T. WIDRICK
$       -  ORIGINAL VERSION
$=======================================================================
$23456789012345678901234567890123456789012345678901234567890123456789012
$        1         2         3         4         5         6         7
$
ALTER    982,982 $ V67                           REPLACE REIGL
$
$  DEFINE DEFAULT VALUES
$
TYPE     PARM,,I,Y,CHECKRBM=-1 $                 DO NOT PRINT CHECKS
$
$  RENAME LAMA AND PHIX FROM REIGL
$
REIGL    KXX,MXX,DYNAMICS,CASES,,MR,DMX,VXCOMPR/
         LAMAO,PHIXO,MI,EIGVMAT,OUTVEC/V,N,READAPP/S,N,NEIGV $
$
$  GENERATE RIGID-BODY AND ELASTIC MODES PARTITIONING VECTOR
$
LAMX     ,,LAMAO/LAMAOM/-1 $                     FORM MATRIX FROM LAMAO
MATMOD   LAMAOM,,,,,/FREQ,/1/3 $                 EXTRACT FREQUENCIES
MATMOD   LAMAOM,,,,,/GNMASS,/1/4 $               EXTRACT GEN MASSES
PARAML   LAMAOM//'TRAILER'/2/S,N,NFREQ $         NFREQ = NO. OF FREQ
MATGEN   ,/VHRBEL/6/NFREQ/NORSET/NFREQ $         H = RB / EL
$
$  SET RIGID-BODY FREQUENCIES TO ZERO
$
PARTN    FREQ,,VHRBEL/FREQRB,FREQEL,,/1 $        PARTITION RB/EL FREQ
MESSAGE  //'  '/ $
MESSAGE  //'RIGID-BODY FREQUENCIES AS OUTPUT BY LANCZOS'/ $
MESSAGE  //'  '/ $
MATPRN   FREQRB// $                              PRINT LANCZOS RB FREQ
MERGE    ,,FREQEL,,,,VHRBEL/FREQEL2/1 $          ROW MERGE WITH NULLS
MATGEN   ,/VLAM1/4/1/2//1//2 $                   VLAM1 = 0 1
MATGEN   ,/VLAM2/4/1/3//1//3 $                   VLAM2 = 0 0 1
MERGE    FREQEL2,,,,VLAM1,/LAMA1/1 $             COL MERGE WITH NULLS
MERGE    LAMA1,,GNMASS,,VLAM2,/LAMA2/1 $         COL MERGE W/ GEN MASS
TRNSP    LAMA2/LAMA3 $                           TRANSPOSE
LAMX     LAMA3,/LAMA $                           FORM NEW LAMA TABLE
$
ALTER 984 $ V67                                  AFTER OFP OF LAMA
$
```

```
$  REPLACE RIGID-BODY MODES IN PHIX BY RIGID-BODY MODES GENERATED
$  FROM STIFFNESS MATRIX ABOUT THE R-SET DOF.  THE RIGID-BODY MODES
$  ARE NORMALIZED AND ORTHOGONALIZED WITH RESPECT TO THE MASS MATRIX
$  USING THE GRAM-SCHMIDT ORTHOGONALIZATION SCHEME.
$
FILE      V6 = OVRWRT $                              ALLOW OVERWRITE
$
$  GENERATE R-SET AND COMPLEMENT OF R-SET PARTITIONING VECTOR
$
IF (NOARED = -1) THEN $
  VEC     USET/VXCOMPRN/'A'/'COMP'/'R' $             A = COMP / R
ELSE $
  PARTN   VACOMPR,,VAXW/VXCOMPRN,,,/1 $              A = COMP / R
ENDIF $
$
$  CALCULATE RIGID-BODY MODES FROM KXX
$
PARTN     KXX,VXCOMPRN,/KCRCR,,KCRRF,/-1 $           SYMMETRIC PARTITION
SOLVE     KCRCR,KCRRF/DMCR//-1 $                     FORM RIGID-BODY TRANS
MATGEN    ,/DMRF/1/NORSET $                          GENERATE IDENTITY MTX
MERGE     DMCR,DMRF,,,,VXCOMPRN/DMXN/1 $             ROW MERGE
SMPYAD    DMXN,MXX,DMXN,,,/MRIG/3////1////6 $        MRIG = RIGID-BODY MASS
MESSAGE   //'  '/ $
MESSAGE   //'SYSTEM RIGID-BODY MASS MATRIX ABOUT R-SET DOF'/ $
MESSAGE   //'  '/ $
MATPRT    MRIG// $                                   PRINT MRIG
$
$  NORMALIZE FIRST RIGID-BODY MODE (OPERATING ON R-SET ONLY)
$
MATGEN    ,/VFIRST/6/NORSET/1/NORSET $               ZERO FOR 1ST MODE
PARTN     DMRF,VFIRST,/INPHI1,,,/1 $                 COLUMN PARTITION
SMPYAD    INPHI1,MRIG,INPHI1,,,/GNM1/3////1////6 $ GNM1= GEN MASS 1ST MD
PARAML    GNM1//'DMI'/1/1/S,N,GNM2 $                 CHANGE TO SCALAR
PARAMR    //'SQRT'/S,N,GNM3/GNM2 $                   GNM3 = SQRT(GNM2)
PARAMR    //'DIV'/S,N,GNM4/1.0/GNM3 $                GNM4 = INV(GNM3)
PARAMR    //'COMPLEX'//GNM4/0.0/S,N,GNM4CX $         CHANGE TO COMPLEX
ADD       INPHI1,/INPHI2/GNM4CX $                    MASS NORMALIZE
MERGE     INPHI2,,,,VFIRST,/PHI/1 $                  COLUMN MERGE W/ NULLS
PARAM     //'SUB'/S,N,VCOUNT/NORSET/2 $              VCOUNT = NORSET - 2
MATGEN    ,/V6/7/NORSET/NORSET/1 $                   GENERATE NULL MATRIX
$
$  MASS NORMALIZE AND ORTHOGONALIZE EACH SUCCESSIVE RIGID-BODY MODE
$  WITH THE PREVIOUS MODES USING GRAM-SCHMIDT METHOD
$
DO WHILE (VCOUNT >= 0) $                             TOP OF LOOP
  PARAM     //'SUB'/S,N,INDX/NORSET/VCOUNT $         INDX = NORSET - VCOUNT
  PARAM     //'SUB'/S,N,INDX1/INDX/1 $               INDX1 = INDX - 1
  MATGEN    ,/VINDX/6/NORSET/0/INDX1/1/VCOUNT $      GEN PARTITION VECTOR
  PARTN     DMRF,VINDX,/V1,,,/1 $                    COLUMN PARTITION
  SMPYAD    PHI,MRIG,V1,,,/V2/3////1 $               V2 = (PHI)T * MRIG * V1
  MPYAD     PHI,V2,V1/V3//-1 $                       V3 = V1 - PHI*V2
  SMPYAD    V3,MRIG,V3,,,/SM1/3////1////6 $          SM1 = (V3)T * MRIG * V3
  PARAML    SM1//'DMI'/1/1/S,N,SM2 $                 CHANGE TO SCALAR
  PARAMR    //'SQRT'/S,N,SM3/SM2 $                   SM3 = SQRT(SM2)
  PARAMR    //'DIV'/S,N,SM4/1.0/SM3 $                SM4 = INV(SM3)
  PARAMR    //'COMPLEX'//SM4/0.0/S,N,SM4CX $         CHANGE TO COMPLEX
  ADD       V3,/V4/SM4CX $                           MASS NORMALIZE
  MERGE     V4,,,,VINDX,/V5/1 $                      COL MERGE WITH NULLS
  EQUIVX    PHI/V6/NEVER $                           BREAK EQUIVALENCE
  ADD       PHI,V5/V6 $                              V6 = PHI + V5
  EQUIVX    V6/PHI/ALWAYS $                          MAKE EQUIVALENT
  VCOUNT    = VCOUNT - 1 $                           RESET VCOUNT
ENDDO $                                              BOTTOM OF LOOP
$
```

```
$  EXPAND RIGID-BODY MODES FROM R-SET TO X-SET
$
MPYAD    DMCR,PHI,/PHICR $                      PHICR = DMCR * PHI
MERGE    PHICR,PHI,,,,VXCOMPRN/DMRGFIN/1 $      ROW MERGE
$
$  CHECK ORTHOGONALITY OF COMPUTED RIGID-BODY MODES IF REQUIRED BY
$  PARAM,CHECKRBM
$
IF (CHECKRBM > -1) THEN $
  MESSAGE  //'  '/ $
  MESSAGE  //'RIGID-BODY SCALING MATRIX TO ORTHOGONALIZE R-B MODES'/ $
  MESSAGE  //'  '/ $
  MATPRT   PHI// $                              PRINT RB MD SCALING MTX
  SMPYAD   PHI,MRIG,PHI,,,/MRBRSET/3////1////6 $ MRBRSET=(PHI)T*MRIG*PHI
  MESSAGE  //'  '/ $
  MESSAGE  //'ORTHOGONALIZED RIGID-BODY MASS MATRIX COMPUTED USING'/ $
  MESSAGE  //'  R-SET RIGID-BODY MODES AND MASS'/ $
  MESSAGE  //'  '/ $
  MATPRT   MRBRSET// $                          PRINT RB MASS MATRIX
  SMPYAD   DMRGFIN,MXX,DMRGFIN,,,/MRBXSET/3////1////6 $ TRIPLE MULTIPLY
  MESSAGE  //'  '/ $
  MESSAGE  //'ORTHOGONALIZED RIGID-BODY MASS MATRIX COMPUTED USING'/ $
  MESSAGE  //'  X-SET RIGID-BODY MODES AND MASS'/ $
  MESSAGE  //'  '/ $
  MATPRT   MRBXSET// $                          PRINT RB MASS MATRIX
ENDIF $
$
$  REPLACE RIGID-BODY MODES IN PHIXO WITH COMPUTED RIGID-BODY MODES
$
PARTN    PHIXO,VHRBEL,/,,PHIXOEL,/1 $           COLUMN PARTITION
MERGE    DMRGFIN,,PHIXOEL,,VHRBEL,/PHIX/1 $     COLUMN MERGE
$-----------------------------------------------------------------------
```