T. S. Butler
321

# FINAL REPORT FOR NASTRAN PROJECT

Prepared for
GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

2 MARCH 1970

CONTRACT NO. NAS5-10049

Prepared by
COMPUTER SCIENCES CORPORATION
Tishman Airport Center 9841 Airport Boulevard
Los Angeles, California 90045

# FINAL REPORT FOR NASTRAN PROJECT

Prepared for
GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

2 MARCH 1970

CONTRACT NO. NAS5-10049

Prepared by
COMPUTER SCIENCES CORPORATION
Tishman Airport Center 9841 Airport Boulevard
Los Angeles, California 90045

Approved by:

L. L. PHILIPSON
Deputy Director, Systems Engineering Center

# SUMMARY

This final report describes the results of Computer Sciences Corporation's (CSC) participation in the NASTRAN Project (Contract NAS5-10049) and summarizes in detail how the objectives of that Project were either met, exceeded, or missed.

NASTRAN is a NASA project monitored by the Goddard Space Flight Center at Greenbelt, Maryland. Its purpose is to provide all NASA Centers, NASA Contractors, Universities, and Industry with a comprehensive computer program that solves a wide range of problems encountered in the field of structural analysis.

To accomplish this purpose the program was required to combine the best of the state-of-the-art in analytical mechanics, numerical methods, and computer programming, and was to be standardized to permit interchange of inputs and outputs between the NASA Centers and their contractors. In addition, the program had to be structured to permit, without major redevelopment, future modifications and extensions of capability into new problem areas and new computer configurations.

The NASTRAN program was also to incorporate both the Force and Displacement approaches to Structural Analysis by the finite element method; establish computer independence; and achieve maximum user convenience.

The requirement to incorporate the Force approach was abandoned, after partial implementation, because of development problems. Restart capability internal to mathematical routines was deferred for further study. Other major objectives, however, have been met or exceeded. For example:

- The program can have over 65 thousand degrees of freedom with over 40 thousand elements.

- There is essentially a single version of the program that runs on a variety of configurations of computer systems: IBM S/360, UNIVAC 1108, CDC 6600, and IBM Direct Couple System.
- The input data and the final output are computer-independent.
- User convenience has been stressed by incorporating, for the most part, such features as nonredundant input and order-independent data decks.

The completed NASTRAN program contains many significant advances in the field of computerized structural analysis. In particular, the sparse matrix routines for decomposition and multiplication of large matrices provide high efficiency in all cases. The eigenvalue techniques provide a set of tools for attacking large order eigenvalue problems effectively. The numerical integration algorithm is stable for a wide spectrum of practical problems without sacrificing either accuracy or efficiency. By choosing to use double-precision arithmetic in key areas, accurate solutions can be obtained from models including thousands of grid points.

The general-purpose nature of the NASTRAN program makes it usable for:

- Any size, shape, or class of structure
- Geometric representations referred to convenient coordinate systems
- Elastic relations ranging from isotropic to general anisotropy
- Restricted nonlinear behavior that can be represented by piecewise linear approximations
- Structural modeling with one- and two-dimensional elements in general, and special three-dimensional cases
- Vibration frequency and mode determination

- Synthesis of parts of the structure by experimentally determined static or dynamic properties
- Loading conditions embracing:
    - static surface and body forces
    - buckling factors
    - static thermal profiles
    - enforced deformations
    - steady sinusoidal excitations
    - time-varying surface and body forces
    - stationary random excitations
- Dynamic combination with scalar force-producing systems
- Solutions by most large second- and third-generation batch mode or multiprocessing digital computers
- Solution results plotted on any of five types of plotters
- Complex as well as real matrix operations

During the Project a comprehensive set of documents were produced including:

The NASTRAN Theoretical Manual (500 pages)
The NASTRAN User's Manual (800 pages)
The NASTRAN Programmer's Manual (1800 pages)
The NASTRAN Demonstration Problems Manual (200 pages)

In addition to this documentation numerous seminars, courses, etc., were given to train users of the NASTRAN program.

The body of this report will describe in a general manner the history of the NASTRAN Project, the features of the program, and the physical aspects of the NASTRAN system. Previously published documentation describes in detail the equations used, problem types solvable, and the program structure.

- Synthesis of parts of the structure by experimentally determined static or dynamic properties
- Loading conditions embracing:
    - static surface and body forces
    - buckling factors
    - static thermal profiles
    - enforced deformations
    - steady sinusoidal excitations
    - time-varying surface and body forces
    - stationary random excitations
- Dynamic combination with scalar force-producing systems
- Solutions by most large second- and third-generation batch mode or multiprocessing digital computers
- Solution results plotted on any of five types of plotters
- Complex as well as real matrix operations

During the Project a comprehensive set of documents were produced including:

The NASTRAN Theoretical Manual (500 pages)
The NASTRAN User's Manual (800 pages)
The NASTRAN Programmer's Manual (1800 pages)
The NASTRAN Demonstration Problems Manual (200 pages)

In addition to this documentation numerous seminars, courses, etc., were given to train users of the NASTRAN program.

The body of this report will describe in a general manner the history of the NASTRAN Project, the features of the program, and the physical aspects of the NASTRAN system. Previously published documentation describes in detail the equations used, problem types solvable, and the program structure.

The NASTRAN program is designed to be modular and modifiable. In particular, as the state-of-the-art advances in computing techniques or structural analysis, the plan is to continously bring NASTRAN up to date. For example, in the development of the program a number of areas of the NASTRAN program have come to light that can and should be exploited to improve its overall efficiency and usefulness to NASA. A symmetric decomposition option could be added to the Determinant Method of Real Eigenvalue Extraction; the matrix subroutine ADD could be rewritten to handle multiple (at least 3) files; the Coupled Transient Analysis module could be reprogrammed to eliminate repeated file operations; a sparse matrix transposition technique could be added using nonzero packing and sorting; a core I/O version of GINO, the General Input/Output utility, could be added for small dynamics problems; the ability to concatenate Old Problem Tapes from several runs can be established; the CDC 6400/6000 version of the program should be converted to single-precision; a non-FORTRAN read/write routine for the IBM 360 version should be provided; and internal restart should be added to the Matrix Mathematical modules.

## TABLE OF CONTENTS

Section 1 - Project History

Section 2 - General Description of the NASTRAN Program

Section 3 - Significant Features of the NASTRAN Program

Section 4 - Program Organization

Section 5 - Problem Size and Solution Times

## TABLE OF CONTENTS (Continued)

Page

Section 6 - Operating Systems and Plotters

Section 7 - Program Documentation

Section 8 - Recommendation for NASTRAN's Further Development

Section 9 - Bibliography

# LIST OF ILLUSTRATIONS

# LIST OF TABLES

# SECTION 1 - PROJECT HISTORY

NASTRAN's history can be broken into three stages: the definition stage and the Phase I and Phase II development stages. The definition stage consisted of the development of program requirements by NASA, and the generation of preliminary program design by two teams of contractors. These preliminary designs, documented in the form of Technical Evaluation Reports (TER), were completed in April of 1966.

The Phase I development stage saw the initial statics and dynamics program delivered to GSFC in April 1969. The Phase II development stage terminated in February 1970. During Phase II the documentation and program installations at six NASA Centers were completed. Specific milestones are listed in Table 1.

## Definition Stage

During the annual review of NASA's research program in January 1964, the area of structural dynamics, headed by Douglas Michel of NASA Headquarters, was discussed. It became apparent that there was considerable effort by many of the NASA Centers to develop computer programs for structural analysis, designed to meet each Center's particular needs. It was suggested that perhaps a single program could meet all the Centers' needs. Accordingly, the Office of Advanced Research and Technology appointed an Ad Hoc Committee with representatives from eight NASA Centers to study this possibility.

After six months of investigation, the Ad Hoc Committee on Computer Methods in Structural Analysis reported to Headquarters that there was no digital program in existence that had broad, uniform capabilities in the three interdependent disciplines of analytical mechanics, numerical methods, and computer programming. The Ad Hoc Committee did observe; however, that there was considerable capability dispersed throughout the aerospace industry which had

Table 1. NASTRAN Milestones (1 of 3)

| Date | Occurrence | Comments |
|---|---|---|
| January 5, 1966 | NASTRAN Definition Contract (Technical Evaluation Report) | Design starts |
| June 29, 1966 | NAS5-10049 awarded to CSC | NASTRAN development starts |
| May 3, 1967 | Initial Program to GSFC | Static Solutions of Rods and Beams (IBM 7094 DCS) |
| August 1967 | Real Eigenvalue Analysis to GSFC | Eigenvalues on IBM 7094 DCS |
| November 1967 | Complete Statics to GSFC | Rigid Formats 1 and 2 on the IBM 7094 DCS |
| May 1968 | Complete Statics to GSFC | Rigid Formats 1 and 2 on the IBM 360 |
| May 1968 | Statics Training Course | Training for NASA representatives from all Centers |
| July 1968 | Complete Statics to JPL | IBM 7094 |
| | Complete Statics to MSC | UNIVAC 1108 |
| | Complete Statics to Lewis | IBM 7094 |
| | Complete Statics to ARC | IBM 7094 |
| August 1968 | Real Eigenvalue Analysis to GSFC | Rigid Format 3 on IBM 360 |

Table 1. NASTRAN Milestones (2 of 3)

| Date | Occurrence | Comments |
|---|---|---|
| February 1969 | Skeleton Program to GSFC | Rigid Formats 1 throu 12 on the IBM 360 |
| April 1969 | Phase I Program to GSFC | Rigid Formats 1 throu 12. Demonstration problems |
| April 1969 | Static Force Method to GSFC | Martin effort delivere |
| May 1969 | User Training at GSFC | Two week User Training Course |
| May 1969 | Installation and Training at JPL | UNIVAC 1108 (EXEC 8) |
| June 1969 | Installation and Training at MSFC | UNIVAC 1108 (EXEC 8) |
| August 1969 | Installation and Training at ARC | IBM 7094 DCS |
| August 1969 | Phase I NASTRAN at LRC | CDC 6600 |
| September 1969 | Theoretical Seminars | East and West Coast Theoretical Seminars |
| October 1969 | Phase I Maintenance Installation at ARC | IBM 360/67 |
| November 1969 | Installation and Training at MSC | UNIVAC 1108 |
|  | Installation and Training at LRC | CDC 6600 |
|  | Phase II deliveries to all Centers |  |

Table 1. NASTRAN Milestones (3 of 3)

| Date | Occurrence | Comments |
|---|---|---|
| January 1970 | Phase I Documentation Printed<br>Programmer Course | 3000 pages of manuals |
| February 1970 | Final Phase II Documentation Delivered to GSFC | |

not been collected into a single program. They found also that there was a tendency toward proprietary secrecy which inhibited exchange of information and that communication was further hindered by the lack of compatibility between (any) two companies' structural analysis programs. Thus, the Ad Hoc Committee recommended that NASA sponsor an entirely new program aimed at consolidating all the best information in the fields of analytical mechanics, numerical methods, and computer programming.

NASA Headquarters endorsed the recommendations of the Ad Hoc Committee and commissioned it to draw up a set of specifications for a new program. To ensure interdisciplinary efforts, teaming was particularly encouraged.

Two teams were selected to prepare Technical Evaluation Reports. One team was headed by Computer Sciences Corporation and augmented by the MacNeal-Schwendler Corporation and the Martin Company. The other was headed by Douglas Aircraft and augmented by Bell Aerosystems, Philco Ford, and Computer Usage Corporation. The CSC team was selected to begin development on the program.

## Development Stage

The development stage proceeded in two distinct phases: Phase I, from July 1966, to April 1969, included general development and validation of the program. Phase II, from April 1969 to February 1970, included installation, training, and documentation of the final program.

Phase I. Early versions of the NASTRAN program embodying static analysis only were developed and delivered to five NASA Centers by July 1968. The JPL IBM 7094 DCS was chosen as a base machine for program development and then the program was converted to the IBM 360 and UNIVAC 1108 systems. The force method development lagged behind that of the displacement method so that the operational program released to the Centers in July of 1968 did not include a force method.

The complete program, including twelve problem solution types, and all elements was ready for validation in January of 1969. A decision to defer the force method development even further was made because budgetary pressures argued for getting the most available method--the displacement method--in the Centers' hands expeditiously. Between January and April of 1969 validation effort was performed to assure the reliability of the final program. Over 200 problems were run on the UNIVAC 1108 at JPL and the IBM 360/95 at the the Goddard Institute for Space Studies. The first complete (i.e., embodying static and dynamic problem capability) program was released to GSFC in April of 1969.

Phase II. The principal Phase II development activities were routines for encompassing nonlinear behavior and coupled mass effects in the plate elements; plotting capability on the incremental plotters; eigenvalue analysis by the Givens method; element labeling in structural plots; and the development of the CDC 6600 NASTRAN system. Three thousand pages of documentation were published during this period. In addition, numerous support and training services were performed, such as user's training at six Centers, two theoretical seminars, and a maintenance support course for installed versions of the program.

# SECTION 2 - GENERAL DESCRIPTION OF THE NASTRAN PROGRAM

The complete NASTRAN program delivered in November of 1969 provides a range of applications extending to almost every type of construction. Structural elements are provided for specifically representing the more common types of construction including rods, beams, shear panels, plates, and shells of revolution. More general types of construction are treated by combinations of some standardized elements and by the use of general elements that model experimental or other internally derived data. Control systems, aerodynamic transfer functions, and other nonstructural elements can also be incorporated into the structural problem in a number of ways.

The program also provides a wide range of analysis capability including static response to concentrated and distributed loads, to thermal expansion, and to enforced deformation; dynamic response to transient loads, to steady-state sinusoidal loads, and to random excitation; determination of real and complex eigenvalues for use in vibration analysis, dynamic stability analysis, and elastic stability analysis.

In addition, the program includes a limited capability for solving nonlinear problems, including piecewise linear analysis of nonlinear static response and transient analysis of nonlinear dynamic response.

The program is specifically designed to allow many degrees of freedom in treating large problems. Generally, economic considerations will limit the problem size well before the limitations of the algorithms are significant. Computational procedures have been selected to provide the maximum obtainable efficiency for large problems.

The foremost consideration in designing the program has been to provide the best obtainable structural analysis program within the existing state-of-the-art. The techniques and methods that have been employed were fully tested by members of the design team.

Two areas of program design that are sensitive to state-of-the-art considerations are program organization and numerical analysis. The organizational demands on the program design are severe in view of the multiplicity of problem types and user conveniences; the multiplicity of operating computer configurations; the requirement for large problem capability; the requirement for flexibility for future modification; and the requirement for responsiveness to improvements in programming systems and computer hardware. These problems were solved by applying techniques that are standard in the design of operating systems but have not, as yet, been extensively used in the design of scientific applications programs.

The main instrument of program organization is the Executive System that schedules the operating sequence of functional modules and plans and allocates storage files. An important aspect of the executive routine concept is that it greatly reduces the cost of program coding and checkout by eliminating most module interface problems and by reducing the remainder to a form that permits systematic treatment.

In implementing effective numerical analysis methods it has been recognized, during program development, that most difficulties arise in connection with three basic operations: matrix decomposition (or inversion), eigenvalue extraction, and integration of large systems of differential equations. The major difficulties that occur in the application of these operations to large problems are excessive computing time, error accumulation, and instability. Many methods that work well with small or moderate sized problems are not acceptable for large problems; therefore, particular care was exercised in the selection of numerical methods for the present program.

The method employed for matrix decomposition was especially important due to its extensive use as a base for the other two basic operations. The method that is employed in the NASTRAN program takes maximum advantage of matrix sparsity and bandedness. The latter aspect is particularly important due to the enormous gain in efficiency that accrues when banding techniques are properly employed in setting up problems for the displacement method.

In general, the solution time for a large structural analysis of any type can be greatly reduced by taking full advantage of the sparsity and bandwidth of the matrices that describe the structural problem. However, other means, in addition to the matrix decomposition routine mentioned above, have been used to improve the efficiency of NASTRAN for large problems. The means employed include storing sparse matrices in packed form, avoiding operations that reduce sparsity or destroy bandwidth, using well designed input/output strategies, and advanced techniques for eigenvalue extraction and numerical integration.

The needs of the structural analyst have been considered in all aspects of the design of the program. For this reason a high degree of flexibility and generality has been incorporated into certain areas of the program. For example, in addition to the usual library of structural elements that refer to specific types of construction, the structural analyst is provided with more general elements that may be used to construct any type of special element; to represent part of a structure by deflection influence coefficients; or to represent part of a structure by its vibration modes.

A number of other essentials is provided for the user, including routines for plotting conventional X-Y plots and for plotting parts of all of the three-dimensional undeformed or deformed structures. The ability to present output in plotted as well as listed form is necessary for any large problem.

The avoidance of errors in the preparation of input data is a major difficulty that the user faces in the solution of large problems. Therefore, in the NASTRAN program, card formats and card ordering are made as simple and flexible as possible to avoid errors caused by trivial violations of format rules. A number of aids for the detection of correctly formatted but incorrect data are also provided, such as specification of zero thickness for a plate or zero length for a bar.

## SECTION 3 - SIGNIFICANT FEATURES OF THE NASTRAN PROGRAM

The principal objective of NASTRAN was to unify the state-of-the-art in three areas: numerical analysis, scientific program organization, and structural analysis. In several areas of numerical analysis and program organization, significant forward steps were taken, particularly in the areas of:

1. Sparse matrix routines
2. Eigenvalue extraction
3. Numerical integration
4. Computer precision
5. The Executive System

In addition, NASTRAN has advanced computing techniques by the addition of a major piece of computer software, a linkage editor, whose application lies outside the bounds of NASTRAN or structural analysis. The linkage editor is discussed in detail in Section 6.

### Sparse Matrix Routines

Most of the matrices used in structural analysis are initially very sparse; however, they tend to fill to various degrees as the problem solution proceeds. Frequently, much of the sparseness, and so computational efficiency, can be retained if reasonable precautions are used in formulating the problem and in performing the matrix operations.

The speed of solutions in NASTRAN relies heavily on taking advantage of matrix sparsity. This is particularly true in matrix decomposition and matrix multiplication, since these are the most time-consuming matrix operations in structural analysis.

Efficient operation with large sparse matrices requires an effective matrix packing scheme in order to minimize the time required to transfer the nonzero elements from secondary storage devices to the working space in main memory.

The packing routine employed in NASTRAN stores the matrices by columns, and each column constitutes a logical record. The first nonzero term in a column is described by an integer that indicates its row position and by a floating point number that describes its value. If the following term is also nonzero, only its value is stored; in general, the position of only the first term in a sequence of nonzero terms is stored.

NASTRAN includes two methods of matrix multiplication, both of which are written for the general multiply-add form $[D] = [A] [B] + [C]$. The first method is used when [B] has a small number of columns. The second method, which is used in all other cases, may be described as a true sparse matrix routine in that the terms of both [A] and [B] are held in packed form in main memory. At any given time in the calculation by the second method, one column of [B] is held in core in packed form along with one column of the partially formed [D] matrix in unpacked form. The remaining storage is allocated to as many columns of [A] in packed form as can be stored. The [B] matrix is passed, column by column, forming partial answers for the current columns of [A] in main storage. The partial answers for each column of [B] are added to the corresponding column of [D]. At the end of one complete pass of the [B] matrix, the columns of the [A] matrix are replaced by new columns and the calculation is repeated. The number of repetitions depends on the size and sparsity of the [A] matrix.

The matrix multiply routine includes a variation for premultiplication of a matrix by the transpose of another matrix, $[D] = [A]^T [B] + [C]$, where [A] is stored by columns. This is done to avoid transposing the [A] matrix, by no means a trivial operation. In fact, surprisingly enough, the multiply routine provides an efficient means for transposition of sparse matrices by setting [B] equal to the identity matrix with [C] null.

The factoring of a matrix into upper and lower triangular forms is, as previously mentioned, a central feature of structural analysis as performed with the NASTRAN program. For large problems, a substantial fraction of the computing time is associated with matrix decomposition. For many matrices used in structural analysis, the creation of nonzero terms during the decomposition is restricted to a relatively narrow band that completely fills along the principal diagonal, and to isolated rows or columns that begin to fill at some point in the decomposition. In other cases, the matrices are either initially full or tend to fill as the decomposition proceeds. In order to handle all situations effectively, the decomposition routines treat all matrices as partially banded; that is, terms clustered near the diagonal are treated inside a band of constant width, and scattered terms outside the band are treated separately. This procedure efficiently treats the general partially-banded case as well as the limiting cases of a full matrix and of a simple band matrix.

In NASTRAN, the diagonal terms are included with the upper triangular factor so that the lower triangular factor has a unit diagonal. For symmetric, positive definite matrices, all leading minors are nonsingular, and hence no pivoting is necessary to complete the triangular decomposition. For these conditions, the upper and lower triangular elements are related. Consequently, the decomposition can be performed by using only the upper triangular elements.

A preliminary pass is made over the original matrix in order to decide how many diagonal rows to include in the band. Terms outside the band give rise to "active columns" which tend to fill as the decomposition proceeds. If the bandwidth is made larger, some of the shorter active columns will disappear inside the band. The combination of bandwidth and active columns that results in minimum computing time is selected for use in the performance of the decomposition.

The procedures for the triangular decomposition of unsymmetrical partially banded matrices are similar to those for symmetrical matrices. The lack of symmetry means that the upper and lower triangular factors are not related and that the widths of the upper and lower bands may be different. However, it is still true that any partially banded portion existing in either the upper or lower triangle of the original matrix will be maintained in the triangular factors. With the matrices that occur in dynamic analyses, whether symmetric or nonsymmetric, there is no assurance that all leading minors are nonsingular. Thus, pivoting must be used to maintain the numerical stability of the triangular decomposition. Pivoting is restricted to take place within the lower band. This will increase the bandwidth of the upper triangular factor by the width of the lower band, but will not otherwise affect the partially banded character of the triangular factors.

The main purpose of using sparse matrix routines is to minimize computer running time. In NASTRAN, the running time for matrix decomposition is particularly short. For example, the time required to decompose a symmetric matrix with a bandwidth of 100 on a third-generation computer will vary from less than 0.5 minute to about 3 minutes per thousand rows, depending on the size of the computer. The running time increases with the square of the semiband and linearly with the order of the problem.

## Eigenvalue Extraction

Three methods of eigenvalue extraction are provided with NASTRAN because no single method or pair of methods has been found that is satisfactory with respect to efficiency, reliability, and generality of application in all situations.

Most methods of algebraic eigenvalue extraction belong to one of two groups: transformation methods and tracking methods. In a transformation method, the matrix of coefficients is first transformed, while preserving its eigenvalues, into a special form (diagonal, tridiagonal, or upper Hessenberg) from which eigenvalues may be easily extracted. In a tracking method, the roots are extracted, one at a time, by iterative procedures applied to the original dynamic matrix. One of the methods used in NASTRAN is a transformation method (the tridiagonal method); the other two (determinant method and inverse power method with shifts) are tracking methods.

The preliminary transformation procedure of the transformation methods requires that the major share of the total effort be expended prior to the extraction of the first eigenvalue. Thus, the total effort is not strongly dependent on the number of eigenvalues that are extracted. In marked contrast, the total effort in the tracking methods is linearly proportional to the number of extracted eigenvalues. Therefore, tracking methods are more efficient when only a few eigenvalues are required and are less efficient when a high proportion of all eigenvalues are required.

The general characteristics of the methods used in NASTRAN are compared in Table 2. The tridiagonal method, due to restrictions on matrix form, is available only for the evaluation of the modes of conservative systems. The other two methods are available for all real and complex eigenvalue problems currently solved by NASTRAN.

It may be noted from Table 2 that a narrow bandwidth as well as a small proportion of extracted roots tends to favor the tracking methods. An example of such a problem is the evaluation of the lowest few modes of a launch vehicle. When the bandwidth is relatively large, and/or when a high proportion of the eigenvalues is required, the tridiagonal method will probably be more efficient. The form of tridiagonalization used in NASTRAN is the Givens method (6) as modified by Wilkinson (7). The Q-R transformation of Francis (8) as modified by Ortega and Kaiser (9) is used to extract the eigenvalues of the tridiagonal matrix.

Table 2. Comparison of Methods of Eigenvalue Extraction

| | Method | | |
|---|---|---|---|
| Characteristic | Tridiagonal Method | Inverse Power Method With Shifts | Determinant Method |
| Most general form of matrix | $[A - pI]$ | $[Mp^2 + Bp + K]$ | $[A(p)]$ |
| Restrictions on matrix character | A real, sym., constant | M, B, and K constant | None |
| Obtains eigenvalues in order | All at once | Nearest to shift point | (Usually) nearest to starting points |
| Takes advantage of bandwidth | No | Yes | Yes |
| Number of calculations, order of | $0(n^3)$ | $0(nb^2 E)$ | $0(nb^2 E)$ |
| Notes: n = number of equations, b = semiband width, E = number of eigenvalues extracted. | | | |

In the determinant method, the determinant is numerically evaluated for a sequence of trial values of the eigenvalue. The iteration algorithm employs a quadratic approximating polynomial based on the values of the determinant in three preceding trials.

Although it is a version of the familiar power method, the inverse power method with shifts finds the eigenvalues in order of closeness to an arbitrarily selected shift point rather than in order of closeness to the origin. Difficulties with convergence and with rigid body modes are thereby avoided. The algorithm applied to a vibration modes problem is:

$$(K - \lambda_0 M) \left\{ w_n \right\} = [M] \left\{ u_{n-1} \right\} \tag{1}$$

$$\left\{u_n\right\} = \frac{1}{C_n}\left\{w_n\right\} \tag{2}$$

where:

$C_n$ is the largest value in $\left\{w_n\right\}$

$\frac{1}{C_n}$ converges to $\Lambda$

$\lambda_0$ is the shift point

The ratio of the elements of $\left\{u_{n-1}\right\}$ to the elements of $\left\{u_n\right\}$ converges to the value of the closest shifted eigenvalue $\Lambda$ and :

$$\lambda = \lambda_0 + \Lambda \tag{3}$$

while $\left\{u_n\right\}$ converges to the corresponding eigenvector.

Wilkinson (7, p. 622) regards the inverse power method with shifts chiefly as a powerful method for refining the eigenvectors of eigenvalues that have been found by other methods. In NASTRAN the inverse power method with shifts is used as a stand-alone method for obtaining both real and complex eigenvalues. As such, it requires sophisticated tracking procedures and convergence criteria. It is the most efficient of the three methods in NASTRAN for structural problems with moderately narrow bandwidths in which only a small fraction of the eigenvalues are desired.

## Numerical Integration

Two separate integration algorithms are provided, one for integrating uncoupled modal equations and the other for integrating coupled equations for either the direct or the modal formulation. The equations of the modal formulation become completely coupled whenever direct input matrix terms due to a control system, a fluid environment, and so forth, are added to the problem.

Since the uncoupled equation for a modal coordinate is a linear second-order differential equation with constant coefficients, its solution by analytical methods is straightforward. A closed form analytical solution is used to generate a numerical recursion formula for evaluating the solution at $t = t_{n+1}$ in terms of the applied loading and the known values of the modal coordinate and its velocity at $t = t_n$. The only approximation is that the applied load varies linearly between $t_n$ and $t_{n+1}$. The algorithm is extremely rapid, provided the time step is not changed at frequent intervals.

The method used in integrating coupled equations has been designed to comply with the special characteristics of structural problems. It is well known that the price paid for high efficiency in the numerical integration of differential equations is a tendency toward instability. Frequently, a choice exists between an efficient algorithm that is unstable for large time steps and an inefficient algorithm that is more than necessarily stable but relatively inaccurate.

In a structural dynamics problem, the stability limit of the integration algorithm may be expressed as the ratio of the maximum permissible time step to the period of the highest vibration mode of the system. For very large systems a limit of this kind is intolerable because the period of the highest mode of the system is generally not known and is, in fact, zero for the very practical case when the mass matrix is singular. Thus, in NASTRAN, every effort was made to provide a stable integration algorithm for the widest possible spectrum of practical problems without sacrificing either accuracy or efficiency.

The integration algorithm used in NASTRAN for coupled equations employs a relatively simple variation of the elementary central difference formula. The exact matrix differential equation:

$$[Mp^2 + Bp + K]\{u\} = \{F\} \tag{4}$$

is replaced by:

$$\frac{1}{\Delta t^2}[M]\left\{u_{n+2} - 2u_{n+1} + u_n\right\} + \frac{1}{2\Delta t}[B]\left\{u_{n+2} - u_n\right\}$$

$$+ [K]\left\{au_{n+2} + (1-2a)\,u_{n+1} + au_n\right\}$$

$$= a\left\{F_{n+2}\right\} + (1-2a)\left\{F_{n+1}\right\} + a\left\{F_n\right\} . \qquad (5)$$

The parameter "a" is selected on the basis of stability and accuracy. It is easily shown (10) that if the damping is uniform and positive (or zero) and if there are no nonlinear terms on the right-hand side, the solution to Equation 5 will be stable for any size of time step, provided $a \geq 1/4$. In NASTRAN, "a" is chosen equal to 1/3 to provide a margin of stability for nonlinear problems. Substituting this value into Equation 5 and collecting terms for the different time steps, we obtain the integration algorithm

$$\left[\frac{1}{\Delta t^2}M + \frac{1}{2\Delta t}B + \frac{1}{3}K\right]\left\{u_{n+2}\right\} = \frac{1}{3}\left\{F_{n+2} + F_{n+1} + F_n\right\}$$

$$+ \left\{N_{n+1}\right\} + \left[\frac{2}{\Delta t^2}M - \frac{1}{3}K\right]\left\{u_{n+1}\right\}$$

$$+ \left[\frac{-1}{\Delta t^2}M + \frac{1}{2\Delta t}B - \frac{1}{3}K\right]\left\{u_n\right\} . \qquad (6)$$

Note that the nonlinear load $\{N\}$ is evaluated at the (n + 1)st time step. The matrix premultiplying $\left\{u_{n+2}\right\}$ is decomposed into its triangular factors so that only the forward and backward passes are performed at each time step. If the stiffness matrix [K] and the damping matrix [B] are well banded, the integration time will be about twice as long per time step as that for a bootstrap formula in which the coefficient of $\left\{u_{n+2}\right\}$ is $[M]/\Delta t^2$. The latter procedure, however, has stability difficulties that can be overcome (if at all) only by reducing the time step to a value that is a fraction of the period of the highest mode of the system.

A measure of the error in Equation 6 is the distortion of the frequency and damping of its normal modes from those of the exact dynamic equations. Analysis of the uniformly damped case shows that error in frequency is given by:

$$\frac{\omega_f}{\omega_e} = 1 - \frac{1}{8} (\omega_e \Delta t)^2 \qquad (7)$$

where:

$\omega_f$ frequency of finite difference system

$\omega_e$ exact frequency

The error in damping is:

$$\frac{\zeta_f}{\zeta_e} = 1 - \frac{1}{3} (\omega_e \Delta t)^2 \qquad (8)$$

Computer Precision

One of the more important early decisions in the design of NASTRAN was the decision to use double-precision arithmetic (approximately 16 decimal digits) in the formation of structural matrices and in matrix decomposition. Single-precision is used primarily in data reduction tasks employing previously calculated vectors and in the transient analysis integration iterations. Since most existing structural analysis programs employ single-precision, an explanation of the reasons for the decision is in order. It should be emphasized that the decision was made on the assumption that a single-precision number has 27 binary bits (approximately 8 decimal digits) rather than the 21 bits that are available with the IBM 360 computer.

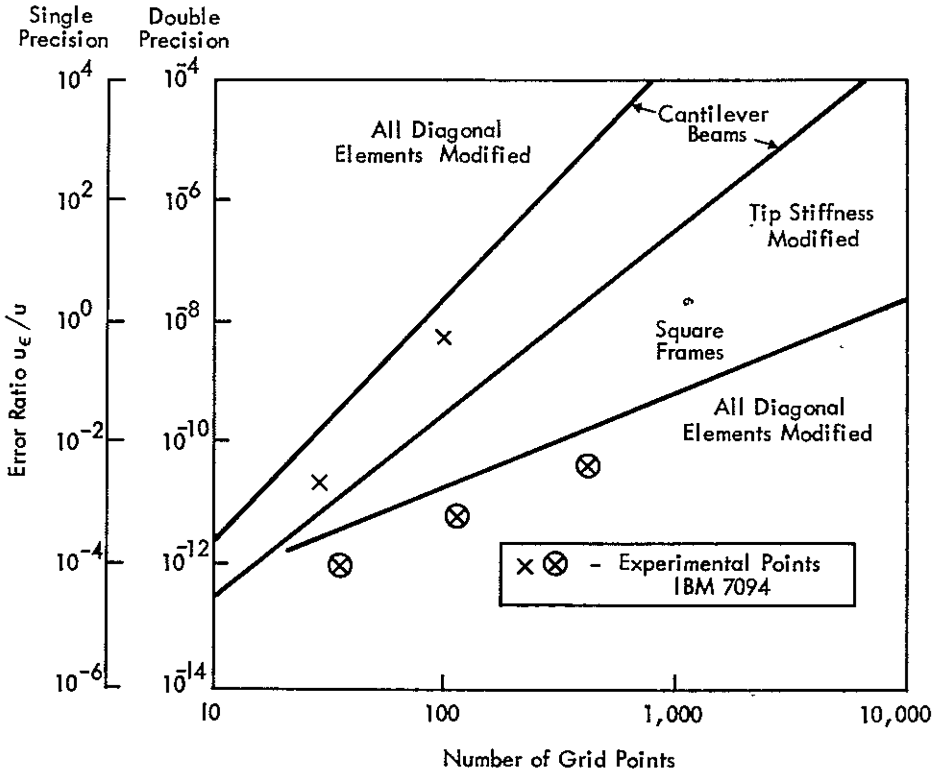
It was known at the time the decision was made that NASTRAN would be called upon to solve structural problems that were large according to the then current standards. Previous experience with single-precision arithmetic was not, therefore, regarded as a reliable guide, and a brief analytical and experimenta investigation was made of the effects of round-off errors in beam and framework problems. Two different kinds of errors were considered: errors arisin in the calculation of the elements of a stiffness matrix; and errors arising in the subsequent triangular decomposition of a stiffness matrix. One of the more interesting results was that errors of the first kind have about the same magnitude as those of the second.

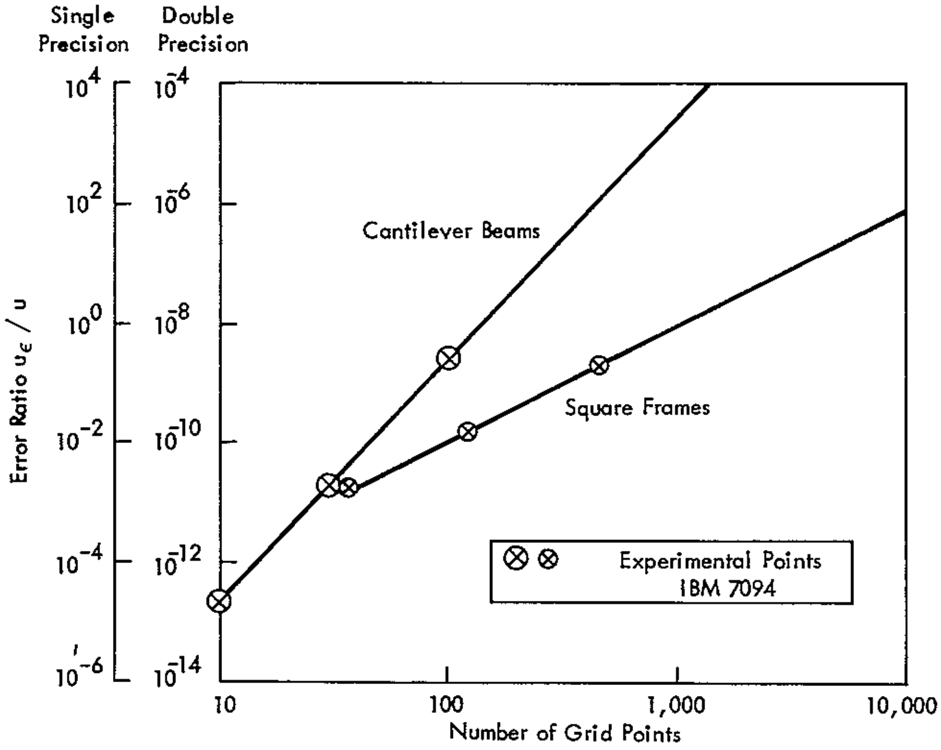
Figure 1 shows the results of analysis and of arithmetic experiments for the errors due to the truncation of the stiffness matrix in uniform beam and framework problems. The ordinate is the ratio of the error in displacement at the free end to the exact displacement. Note that if three-place accuracy in the solution is required, fewer than 100 grid points can be used with single-precision arithmetic, whereas many thousands of grid points can be used with double-precision arithmetic.

Figure 2 shows similar results due to the accumulation of round-off errors in the triangular decomposition of the stiffness matrix. Somewhat poorer results can be expected in cases where the stiffness varies widely throughout the structure. The conclusions that were derived from the studies are that double-precision arithmetic is required for large problems and that it will give accurate answers in the vast majority of cases.

## User Convenience

Many of the special programming features of NASTRAN are related either directly or indirectly to user convenience. Every effort has been made to relieve the user of all tasks other than those associated with engineering judgment and structural modeling. The procedures for input preparation


Single Precision
Double Precision
Error Ratio $u_\epsilon/u$
$10^4$
$10^2$
$10^0$
$10^{-2}$
$10^{-4}$
$10^{-6}$
$10^{-4}$
$10^{-6}$
$10^{-8}$
$10^{-10}$
$10^{-12}$
$10^{-14}$
All Diagonal Elements Modified
Cantilever Beams
Tip Stiffness Modified
Square Frames
All Diagonal Elements Modified
× ⊗ - Experimental Points IBM 7094
10
100
1,000
10,000
Number of Grid Points


Single Precision
Double Precision
Error Ratio $u_\epsilon/u$
$10^4$
$10^2$
$10^0$
$10^{-2}$
$10^{-4}$
$10^{-6}$
$10^{-4}$
$10^{-6}$
$10^{-8}$
$10^{-10}$
$10^{-12}$
$10^{-14}$
Cantilever Beams
Square Frames
⊗ ⊗ Experimental Points IBM 7094
10
100
1,000
10,000
Number of Grid Points

have been simplified in order to minimize the chances of human error in problem preparation. All items of input are checked for consistency and correctness of format. In case of errors, detailed diagnostic messages are provided to assist the user in making corrections.

## The Executive System

The main instrument of program organization in NASTRAN is an Executive System that controls the sequence of operations in the problem solution, manages the secondary storage files, maintains full restart capability, and generally relieves the user of any concern for the operation of the program. The problem solution functions are performed by modules that are separated strictly along functional lines. These functional modules operate as independent subprograms and may not call, or be called by, other modules. They may be entered only from the Executive System. Modules may interface with other modules only through auxiliary storage files and indirectly through parameter tables that are prepared and managed by the Executive System. This separation of system and problem solution functions permits the modification or addition of functional capability with relatively minor modifications to the Executive System. The existing functional modules are, in general, not affected by such changes.

NASTRAN has been designed to run on several different computing systems and to permit extension to other machines. It is operational on the following computing systems:

1. IBM 7094/7040 (44) Direct Coupled system under the IBSYS operating system.
2. IBM System/360, Model 65 and up, under the 360 Operating System.
3. UNIVAC 1108 under the EXEC 8 operating system.
4. CDC 6600 under the SCOPE 3.0 operating system.

A high degree of machine independence has been achieved in NASTRAN by writing in FORTRAN IV the complete code for the functional modules and almost all of the code for the Executive System. The machine-dependent routines are restricted to operations where a direct interface with the operating system of the computer must be made. The main operations in this category are the identification and use of secondary storage files and the identification of the part of main memory that is available to the functional modules for working space.

Although modern computers have sufficient secondary storage to accommodate the needs of NASTRAN, the number of separate files into which the storage can be divided is usually limited. In general, the number of data blocks required for the solution of a problem far exceeds the number of files available. Consequently, the dynamic allocation of secondary storage files is an important function of the Executive System. The Executive System maintains tables that indicate when the information on a particular file is no longer needed and hence when the file is available for assignment. If the number of files is severely limited, it may be necessary for the Executive System to temporarily pool on a single file the several data blocks that are not currently needed. These data blocks are unpooled and assigned to separate files at some later stage in the solution when they are needed.

The main memory of the computer is also used in a dynamic manner. Since all intermediate results are stored on secondary devices, that portion of main memory assigned to NASTRAN that is not occupied by instructions, local data, or input/output buffers is available as working space during calculations. Working space in core is dynamically assigned to subroutines by the NASTRAN Executive System.

If the main memory working space is insufficient to complete an operation, the subroutines are programmed to use secondary storage devices as an extension of main memory. These "spill" operations increase the running time but permit the solution of large problems with smaller blocks of main memory than would otherwise be the case.

The most general way of using NASTRAN to solve a problem is for the user to instruct the Executive System directly as to the sequence of operations that he wishes performed. Such a sequence of operations is called a Direct Matrix Abstraction Program (DMAP). A DMAP sequence may be used to manipulate matrices or solve structural problems. However, to relieve the user of the necessity for constructing such a sequence of operations for the more common types of structural analyses, a set of 12 rigid formats has been included in NASTRAN. The user may make additions or deletions to the sequence of operations specified in a particular rigid format. Provision is also made for the addition of new rigid formats.

Each rigid format contains a set of restart tables that enables the Executive System to automatically modify the specified sequence of operations to account for any changes that are made in the input when making a restart after having previously run all or a part of the problem. One of the tasks of the Executive System is to maintain a copy of all data blocks that are needed for restart purposes. This copy is made on one or more physical tapes so that the tapes can be dismounted and saved at the conclusion of the run. With the assistance of the restart tables, the Executive System is able to determine which parts of the previous solution are still valid, with due regard for any changes that may have been made in the structural model or in the requests for output. This determination can be made by the Executive System even though the previous solution was made with a different rigid format.

### Structural Analysis Features

NASTRAN employs the finite element approach to structural analysis, i.e., the distributed physical properties of structures are represented by idealized lumped element models. The idealized models consist of "grid points" to which loads are applied and at which degrees of freedom are defined, and "elements" that are connected to or between the grid points. The elements define the elastic, inertial, and damping properties of the structure.

The geometrical location of grid points and the coordinate axes used to measure displacements may be specified by the user either in a basic rectangular coordinate system or in "local" coordinate systems that may be rectangular, cylindrical or spherical. All derived geometrical properties (such as the side lengths and surface areas of elements) are computed internally by the program. A special class of grid points is provided called "scalar points", which do not have geometrical locations, and for each of which, only one degree of freedom is defined. These points are useful for the representation of control system variables and of abstract quantities such as the generalized coordinates of vibration modes.

Material properties may be isotropic or anisotropic, and may be temperature dependent.

Mass properties of the structure are specified in four ways: by structural mass density obtained from material properties tables; by a nonstructural mass density specified for each element; by mass concentrated at grid points and defined by a six-by-six matrix; and by representing mass as having an influence at more than one grid point (coupled).

Structural Elements. Much of the individuality of a structures program is exhibited in the structural elements that it employs. Since the intended range of applications of NASTRAN is broad, the number of different structural elements is large and their properties are less specialized than in most other programs. In addition, the modular design of NASTRAN facilitates the addition of new elements.

The structural elements that are currently available in NASTRAN are the following:

1. Beam and Rod Elements - For these elements, a number of special features are included, such as the ability to offset the neutral axis of a beam from the grid points to which it is attached; and the ability to remove the connections between the grid points and any of the six motions at either end of the beam.

2. Shear and Twist Panels - These panels may have a general quadrilateral shape and the corners need not lie in a plane. Garvey's formulation (1) is used.

3. Plates - A total of nine different plate elements include: those with triangular and general quadrilateral shapes; those with solid and sandwich cross sections; and those with membrane, bending, or combined membrane and bending action. Material properties may be isotropic or anisotropic. A uniform state of stress is assumed for membrane action. For bending action, the stiffness matrix is derived from an assumed deflection function consisting of a truncated double power series expansion in the planform (X,Y) coordinates. Transverse shear flexibility is included (optionally) in all bending elements. One of the forms is the Clough bending triangle (2). Another is the simpler "basic" bending triangle, which has a linear variation of normal slope along one edge only.

   The quadrilateral bending element consists of four basic bending triangles arranged so that each is connected to a different set of three corner points, and so that the edge of each triangle with the linear variation of normal slope lies along a diagonal of the quadrilateral. Since the surface area of the quadrilateral is covered twice by the triangles, the resulting stiffness matrix is divided by two.

4. Axisymmetric Conical Shell - The axisymmetric conical shell element includes transverse shear flexibility and permits nonaxisymmetric loading.

5. Doubly Curved (Toroidal) Shell - The toroidal shell element (3) includes higher-order displacement functions that are accommodated by adding the first derivative of tangential displacement and the second derivative of normal displacement along the meridian as degrees-of-freedom at the grid points. The loading must be axisymmetric.

6. Solid of Revolution Elements - The solid of revolution elements (4) and (5) are rings with triangular and trapezoidal cross sections. They are used in the analysis of axisymmetric solids with axisymmetric loads.

7. Scalar Elements - The scalar spring and the scalar damper can be connected between a single degrees-of-freedom and "ground", or between any pair of degrees-of-freedom.

8. General Element - The general element can be connected to any number of degrees-of-freedom. Its properties are defined by deflection influence coefficients rather than by stiffness coefficients because the former are more directly related to test results and because the latter require extremely accurate values to avoid violation of free body properties.

9. Constraint Element - This element, a special feature of the program that implies a linear relationship among the degrees-of-freedom to which it is attached. It has a variety of uses including

the introduction of enforced motions, simulation of very stiff members or rigid bodies, and the generation of nonstandard structural elements.

Material properties for elements are specified by reference to material property tables which include density, elastic coefficients, thermal expansion coefficients, and damping coefficients for each type of material.

## SECTION 4 - PROGRAM ORGANIZATION

The NASTRAN program is organized into two distinctly separate parts: the functional, or problem solution, modules and the Executive System.

The mathematical computations required to solve problems are performed by the various functional modules as directed by the Executive System. Sequences of module executions for various classes of structural problems are stored in rigid formats which are embedded in the Executive System. Execution of any problem may proceed to a final solution in one run, or at the option of the user, to an intermediate point.

### The Executive System

The essential functions performed by the (NASTRAN) Executive System are:

- Establish and control the sequence of module executions according to options specified by the user.
- Establish and communicate values of parameters to each module.
- Allocate system files for all data blocks generated during program execution, and perform input/output to auxiliary files for each module (in NASTRAN functional modules interface only with the Executive System and not with each other).
- Maintain a full recovery capability for restoring a program execution after either a scheduled or unscheduled interruption.

Each of these functions is essentially independent of any particular feature of structural analysis, and applies to the operational control of any complex multimodule, multifile application program. The Executive System is open-ended in the sense that it can accommodate an essentially unlimited number of functional modules, files, and parameters. Modification of the Executive System necessary for changing or extending of functional capability is restricted to changes in entries in control tables stored within the executive routines.

### Rigid Formats

The functional, or problem solution, modules are grouped into a logical sequence to form rigid formats that provide a complete solution to each class of structural problem. There are four general classes of problems for which solutions are provided, they are:

1. Static Structural Problems
    - Basic Static Analysis
    - Static Analysis with Inertia Relief
    - Static Analysis with Differential Stiffness
    - Piecewise Linear Analysis
2. Elastic Stability Problems
    - Linear Buckling Analysis
3. Vibration Problems
    - Normal Modes
4. Dynamic Structural Problems
    a. Direct Method
        - Complex Eigenvalue Analysis
        - Frequency and Random Response
        - Transient Response

    b. Modal Method
        - Complex Eigenvalue Analysis
        - Frequency and Random Response
        - Transient Response

Static Structural Problems. The basic steps followed in obtaining a solution for a static analysis by the displacement method in NASTRAN are as follows:

- Assemble the stiffness matrix at the grid points.
- Impose constraints to eliminate dependent degrees of freedom.
- Decompose the reduced stiffness matrix into triangular factors.
- Form the load vectors.
- Solve for the independent displacements using the triangular factors of the reduced stiffness matrix.
- Recover the dependent displacements and stresses.

Each of the above steps is performed by one or more functional modules. Optional features of basic static analysis include: solution by matrix partitioning; checks on the consistency of determinate reactions (f any are provided); and checks on the residual vectors of the static solutions.

Additional capabilities are provided by the alternate case types that include inertia-relief effects for free bodies; a first approximation to the effects of large displacements and rotations (static analysis with differential stiffness); and the inclusion of the solution of certain types of problems with nonlinear stress-strain relationships by piecewise linear analysis. All of these features are treated automatically without user intervention.

Static loads are introduced into the analysis in the following ways:

- Load components at individual grid points.
- Pressures on panels
- Body force loads (by specifying magnitude and direction of field vector, gravity, centrifugal and steady acceleration, and mass distribution).

- Enforced displacements and deformations.
- Thermal loads at grid points (by specifying temperatures and thermal expansion coefficients of materials).

An important feature of static analysis by the displacement method is that solution time can be greatly reduced by numbering the grid points in a numerical order that will enhance the "banded" character of the stiffness matrix. In order to facilitate the use of this feature, a grid point sequence table is generated that correlates an unordered grid point list with a list that the user supplies for improving matrix banding.

Elastic Stability Problems. NASTRAN includes the capability for automatic calculation of buckling modes using linear elastic stability theory. The steps involved in the calculation by the displacement approach are as follows:

1. Perform a complete static analysis for a specified primary loading condition.
2. Calculate and assemble the differential stiffness matrix from knowledge of the displacements of the elements. Differential stiffness is automatically computed for rods, beams, shear panels, and triangular and quadrilateral membrane elements and the axisymmetric conical shell.
3. Solve the eigenvalue problem:

$$([K_s] + \lambda[K_d]) \{u\} = 0 \tag{9}$$

where:

$[K_s]$ is the structural stiffness matrix

$[K_d]$ is the differential stiffness matrix

$\lambda$ is the factor of proportionality between the buckling load and the specified primary load

An implied assumption is that the magnitudes and directions of load components in the primary loading condition are independent of the displacements of the structure. More general elastic stability problems can be solved by means of externally generated matrix elements that are inserted into the program.

The differential stiffness matrix can be added to the structural stiffness matrix to obtain an improved solution to the static loads problem, which amounts to a first-order correction for the effects of large displacements.

Vibration Problems. The methods of eigenvalue extraction used in the calculation of vibration modes are described in Section 3. Some of the special features provided in NASTRAN are:

- Separate calculation of, and check on the consistency of, the rigid body modes.
- Mass-orthogonality check.
- Automatic calculation and orthogonalization of eigenvectors associated with multiple roots.
- Preparation of modal properties for subsequent dynamic analyses. Either the number of modes or the frequency range of the modes may be specified.

A general feature of the program that is important for the calculation of vibration modes and for all other types of dynamic analyses is the ability to reduce the number of degrees-of-freedom from a very large number that may have been used in a previous static analysis to a smaller number that is more appropriate for dynamic analysis. In order to accomplish this, the user specifies the degrees-of-freedom that will be retained, and the stiffness and mass matrices are automatically recomputed for the reduced set in such a manner that the error in kinetic energy is minimized.

Dynamic Structural Problems. Dynamic problems are separated in NASTRAN into the following types:

- Complex eigenvalue problems, such as flutter problems.
- Frequency and random response problems.
- Transient response problems.

Each of these problem types can be solved either by a direct or by a modal approach.

The basic steps followed in obtaining a solution for a dynamic analysis by the direct approach are as follows:

1. Form the structural stiffness, mass, and damping matrices. These matrices are generated in the static portion of the program, and need not be regenerated if a static analysis had previously been performed.
2. Add additional matrix terms that describe unconservative effects, such as aerodynamic or hydrodynamic terms. The additional terms may be directly input by the user. The program includes a provision for the automatic generation of matrix terms due to control system transfer functions.
3. Add a nonviscous structural damping effect. This effect may be specified at the element level or in terms of a single structural damping coefficient for the entire structure.
4. In the case of transient response problems, generate nonlinear terms. The concept of nonlinear elements, such as those employed in analog computation, is used for the generation of nonlinear terms. The nonlinear elements in the current program include the multiplier and the arbitrary function generator. Any desired relationship can be simulated by means of these devices.

5. In the case of frequency response and transient response problems, calculate load vectors.

6. Perform the implicit operations required to obtain a solution for the independent degrees-of-freedom.

7. Recover dependent displacements and stresses by the same modules that are used in static analysis.

Additional steps are required for solution by the modal approach. These involve transformation of load vectors, solution vectors, and the added matrix terms via the eigenvector matrix. The modes that are used are the orthogonal vibration modes for the conservative structural system.

The response to random excitation is calculated in frequency space as a data reduction task on the results of frequency response analysis, over the portions of the spectrum so selected. Conversion to the autocorrelation function is obtained by Fourier transformation.

## Output

In all of the above problem classes different output quantities may be selected. These quantities include displacements (and velocities and accelerations in dynamic analyses) of grid points, loads applied to grid points, internal forces in structural elements, and stresses in structural elements. All quantities may be printed. Two types of plot format are available. X-Y plots are used for recording transient and frequency responses. Structures plots (in orthographic, perspective, or stereographic projection) are used to display the deformed structure (superimposed on the undeformed structure if desired) for the results of static analysis, vibration modes, and transient analysis (at specified times). The scale of the deformations is applied according to user request.

# SECTION 5 - PROBLEM SIZE AND SOLUTION TIMES

In designing NASTRAN as a user-oriented, general-purpose program many factors were taken into consideration involving problem size and solution times. These factors are discussed here to provide information with which problem sizes and solution times can be estimated. Two general limitations on feasible problem size are that the decomposition routine requires sufficient main storage to hold the nonzero terms of two columns of the stiffness matrix, and that all vector operations require space for at least one full vector. The decomposition limitation is of no consequence, since the computing times tend to become excessive before the size limitation is reached. The limitation on vector operations is not likely to be serious except in computers having the smallest main memories. There are additional limitations on problem sizes for dynamics solutions in that more vectors must be held in core; however, the sizes of matrices required for dynamics problems tend to be considerably smaller than those required for static analyses of comparable accuracy. In all cases, however, there must be sufficient capacity on secondary storage devices to hold all of the data blocks generated during the problem solution.

## Solution Times

Since many third-generation computing systems operate in a multiprogramming mode, solution times are discussed in terms of central processor time required for execution. The wallclock times will vary, depending on the number and types of programs executing simultaneously, as well as the characteristics of the secondary storage system.

Solution times for problems in static analysis can be described in terms of the time required for the triangular decomposition of the stiffness matrix which increases with the square of the bandwidth, the time required for multiplications of a matrix times a vector varies linearly with the bandwidth, and the time to generate the stiffness matrix which is dependent only on the size of the problem. For a statics problem with a semiband of 100, the time to generate the stiffness matrix will vary from less than one-half minute to two minutes per thousand rows, and the time to decompose this matrix will be almost the same. The vector operations consume another one-half to one minute per thousand rows, for a total solution time of from one and one-half to five minutes per thousand rows.

For problems having large bandwidths, the decomposition times are substantially greater than for smaller problems. The additional time is required to store and retrieve intermediate results of the decomposition from secondary storage devices when main memory is of insufficient size to hold all of the intermediate results. For a 65, 000-word core, the dividing line between large and small problems is at a semiband of about 200, and with a 1, 000, 000-word core, the dividing line is at a semiband of about 1, 000.

The central processor time required to extract real eigenvalues by the inverse power method with shifts, for problems with semibands of 100, varies from five to twenty minutes per thousand rows per eigenvalue extracted. Since these times are dominated by the execution of the decomposition routine, they can be expected to increase with the square of the bandwidth.

### Demonstration Problems

When NASTRAN was completed, twenty-five problems were used for demonstration and acceptance testing. These problems were run on all four NASTRAN computers and time comparisons were made. Table 3 lists the problems and Table 4 gives the running times. An engineering report on these tests is described in Report on the Demonstration Problems (NASA SP224).

Table 3. NASTRAN Demonstration Problems

| Number | Description | Analysis (Rigid Format) | Size (Number Grid Points) |
|---|---|---|---|
| 1-1<br>1-1A<br>1-1B | Delta Wing | Static Analysis<br>Same, New Load (Restart)<br>Normal Modes | 93 |
| 1-2<br>1-2A | Spherical Shell, Pressure Loading | Static Analysis<br>Same, New Boundary Condition | 26 |
| 1-3 | Free Rectangular Plate, Thermal Loading | Static Analysis | 247 |
| 1-4<br>1-4A | Long, Narrow Orthotropic Plate | Static Analysis<br>Same, New Load | 306 |
| 1-5 | Nonsymmetric Bending of a Cylinder of Revolution | Static Analysis | 51 rings |
| 1-6 | Solid Disk, Radially Varying Thermal Load | Static Analysis | 26 |
| 1-7 | Shallow Spherical Shell, External Pressure Loading | Static Analysis | 26 |
| 2-1 | Circular Ring, Concentrated and Centrifugal Loads | Inertia Relief | 25 |
| 3-1 | Vibration of 10 x 20 or 20 x 40 Plate | Real Eigenvalue Analysis | 231 |
| 4-1 | 100 Cell Beam Under Axial and Bending Loads | Differential Stiffness | 101 |
| 5-1 | Symmetric Buckling of Cylinder | Buckling Analysis | 33 |
| 7-1 | 500 Cell String with Damping | Complex Eigenvalue | 500 deg-of-freedom |
| 8-1 | 10 x 10 or 20 x 20 Plate | Frequency Response | 111 or 421 |
| 9-1 | Direct Matrix Input | Transient Analysis | 4 x 4 matrices |
| 9-2 | 1000 Cell String, Traveling Wave | Transient Analysis | 1001 deg-of-freedom |
| 10-1 | Rocket Guidance and Control Problem | Complex Eigenvalue Analysis | -- |
| 11-1<br>11-2 | Ten Cell Beam<br>500 Cell String | Frequency and Random Analysis<br>Frequency Response | 11<br>501 deg-of-freedom |
| 12-1 | Free 100 Cell Beam | Transient Analysis | 101 |

Table 4. Summary of Computer Times (Units are Minutes)

| Problem Number | IBM 360/95 | | IBM 360/67 | CDC 6600 | | UNIVAC 1108 | | IBM 7094 |
|---|---|---|---|---|---|---|---|---|
| | CPU | Wall | CPU | CPU | Wall | CPU | Wall | Wall |
| 1-1 | 1.0 | 5.0 | 3.0 | 0.6 | 4.7 | 1.2 | 14.4 | 10.0 |
| 1-1A | 0.5 | 2.9 | 1.7 | 0.4 | 1.8 | 0.4 | 8.8 | 8.0 |
| 1-1B | 1.8 | 16.0 | 4.9 | 1.5 | 5.6 | 1.2 | 14.4 | 13.0 |
| 1-2 | 1.4 | 8.0 | 4.4 | 1.2 | 4.5 | 1.0 | 13.3 | 14.0 |
| 1-2A | 0.8 | 4.5 | 2.4 | 0.6 | 3.6 | 0.5 | 10.8 | 11.0 |
| 1-3 | 1.9 | 7.2 | 5.6 | 2.0 | 6.6 | 1.6 | 17.2 | 18.0 |
| 1-4 | 17.0 | 52.0 | 24.4 | 11.1 | 36.8 | 17.0 | 125.0 (5 x 50)[1] | 60.0 (5 x 40)[1] |
| 1-4A | 1.0 | 6.0 | 2.9 | 0.5 | 3.4 | 0.5 | 12.1 | 6.5 |
| 1-5 | 11.0 | 19.0 | 30.0 | N/A | N/A | 18.0 | 50.25 (20)[2] | 38.0 (10)[2] |
| 1-6 | 0.8 | 10.5 | 2.2 | 0.5 | 2.2 | 0.4 | 14.8 | 11.0 |
| 1-7 | 1.0 | 4.1 | 4.5 | 2.0 | 3.5 | 1.5 | 16.7 | 17.0 |
| 2-1 | 1.2 | 12.0 | 2.8 | 0.5 | 2.8 | 0.4 | 7.5 | 13.0 |
| 3-1 | 12.0 | 38.6 | 65.0 | 13.8 | 31.4 | 11.0 | 51.2 (3)[3] | 26.0 (1)[3] |
| 4-1 | 2.8 | 16.9 | 8.1 | 2.4 | 9.3 | 1.5 | 40.5 (3)[4] | 27.0 (1)[4] |
| 5-1 | 6.8 | 16.0 | 18.9 | 8.1 | 22.6 | 5.3 | 50.0 (4)[3] | 28.0 (1)[3] |
| 6-1 | N/A | N/A | 86.9 (25)[4] | 32.6 | 109.0 | 4.5 | 15.2 (3)[4] | N/A |
| 7-1 | 3.8 | 12.0 | 10.1 | 3.6 | 8.8 | 3.4 | 17.8 | 20.0 |
| 8-1 | 12.2 | 31.6 (20 x 20)[1] | 7.78 (10 x 10)[1] | 2.1 | 7.3 | 2.3 | 25.6 | 22.0 |
| 9-1 | 1.5 | 7.0 | 6.2 | 0.7 | 5.5 | 0.6 | 11.6 | 17.0 |
| 9-2 | 6.6 | 16.0 | 17.1 | 6.3 | 16.8 | 3.5 | 31.7 (1000)[5] | 26.0 (500)[5] |
| 10-1 | 3.4 | 22.5 | 9.5 | 2.0 | 8.4 | 1.1 | 9.2 | 45.0 |
| 11-1 | 2.7 | 18.5 | 7.0 | 1.6 | 8.7 | 1.4 | 25.4 | 23.0 |
| 11-1A | 0.6 | 3.0 | 1.9 | 0.3 | 2.1 | 0.3 | 6.7 | 9.0 |
| 11-2 | 15.7 | 68.0 | 41.4 | 14.7 | 48.5 | 8.5 | 33.8 (20)[3] | 62.0 (10)[3] |
| 12-1 | 5.8 | 30.5 | 17.2 | 5.6 | 24.4 | 4.9 | 23.9 | 39.0 |

1. Mesh size
2. Number of harmonics
3. Number of modes
4. Number of loading increments
5. Number of degrees-of-freedom

In regard to Table 4, several sizes of some problems were run. This was necessary due to the wide disparity of the computer capabilities. Problem sizes for these problems are shown to the right of the running times. When a problem size restriction appears all computers to the left used this size. Thus, for example, problem 9-2 had 1000 degrees-of-freedom on the IBM 360/95, the IBM 360/67, the CDC 6600, and the UNIVAC 1108 but had only 500 degrees-of-freedom on the IBM 7094.

The following gives the core storage used and the number of peripheral files for each computer:

| Computer | Core (Words) | Number of Files |
|---|---|---|
| IBM 360/95 | 125K | 43 |
| IBM 360/67 | 125K | 43 |
| CDC 6600 | 54K | 35 |
| UNIVAC 1108 | 65K | 35 |
| IBM 7094 | 32K | 31 |

## SECTION 6 - OPERATING SYSTEMS AND PLOTTERS

The size of NASTRAN and its ability to handle very large matrices necessitate the specification of a minimum amount of main core memory and secondary (peripheral) storage. The minimum number of uniquely addressable secondary files is thirty. At least twenty or more of these should be random-access devices. Two or three, in general, but at certain times five or more, must be magnetic tape drives. The NASTRAN Executive System has been adapted to the computer configurations shown in Table 5.

Table 5. Computer Configurations

| Computer | Operating System | Minimum Main Core Assignment | Peripheral Storage |
|---|---|---|---|
| IBM 7094/7040 Direct Coupled | IBSYS/DCOS | 32K words | 111301 disc module and tapes |
| UNIVAC 1108 | EXEC 8 | 64K words | At least one FASTRAND drum |
| IBM 360 Model 50 | OS 360/PCP | 235K bytes of a 256K-byte core | At least one 2314 disc |
| IBM 360 Models 50, 65, 67 | OS 360/PCP or MFT<br>HASP permitted | 300K bytes of a 512K-byte core | At least one 2314 disc |
| IBM 360 Models 50, 65, 67, 75, 85, 91, 95 | OS 360/PCP or MFT or MVT | 400K bytes of a 1000K-byte core | At least one 2314 disc |
| CDC 6400<br>6500<br>6600 | SCOPE 3.2<br>RUN compiler with nonstandard returns | 130K octal | 3 million words of 6803 disc |

The plotting capability is supported on three general types of plotters: table, microfilm, and drum. The design of the plotting capability allows for minor local differences to accommodate specific characteristics of various plotters. For structural plots, the user has the options of calling for deformed and/or undeformed structure plots, the width of lines, the vantage point, a portion of the structure, and a choice of orthographic, perspective, or stereoscopic projections. For curve plots, the user can choose in terms of time or frequency for the history of any output quantity and for any degree-of-freedom. Currently, the plotting capability is supported on these plotters:

| Manufacturer | Type | Model |
|---|---|---|
| Electronics Associates | Table | EAI 3500 |
| Benson Lehner | Table | LTE or STE |
| Stromberg Carlson | Microfilm | SC 4020 |
| Display Data | Microfilm | DD 80 |
| CalComp | Drum | 563 |
| CalComp | Drum | 565 |
| CalComp | Drum | 763 |
| CalComp | Drum | 765 |

## The CDC 6400/6600 Linkage Editor

Complex problems require complicated programs which grow in size as problem-solving capability advances. In a multiprogramming environment large problems utilize memory most efficiently by partitioning and overlaying those program portions which need not reside in memory simultaneously. The loader is responsible for calling to memory the overlays as requested.

While there is some probability that a large portion will compile successfully on different computers, the probability that it will load and execute is almost nil. Any programmer-analyst who has implemented an overlaid program on more than one computer can testify to rearrangements that ought not be necessary.

In order to adapt the NASTRAN program to the CDC operating system it became necessary to augment the CDC-supplied software by writing the CDC 6400/6600 NASTRAN linkage editor. This linkage editor has been designed to provide an efficient and effective means of utilizing core storage for medium to large programs. The existing loader for the CDC 6400/6600 systems has the following disadvantages:

1. Only two levels of overlay are provided beyond the root segment.
2. An overlay segment must be explicitly called as opposed to being automatically supplied by the operating system. Consequently, the overlay structure must be known when the program is coded.
3. An overlay segment may be entered at one point only. Consequently, downward calls are extremely limited.
4. No facility exists to explicitly position labeled common blocks.
5. Loading of overlay segments is accomplished from a sequential rather than a direct access file, thus providing unnecessary search time.

The CDC 6400/6600 NASTRAN linkage editor in conjunction with its partner, the segment loader, overcomes these disadvantages in the following ways:

1. An unlimited number of overlay levels is provided.
2. The programmer describes the overlay structure to the linkage editor after the program is coded. The linkage editor provides implicit segment loading.
3. Complete communication between all levels of overlay is maintained.

4. Linkage editor control statements may be used to explicitly position subprograms and named common blocks.

5. The overlay segments are maintained in an indexed file. Consequently, every segment is immediately available to the segment loader.

This linkage editor provides compatibility between other larger-scale systems (IBM 7094 IBSYS, IBM 360 OS, UNIVAC 1108 EXEC 8) overlay techniques and those now available on the CDC 6000-series computers. Using the NASTRAN linkage editor conversion times for programs to the CDC-6000 series machines should be substantially reduced. Further details on the CDC-6600 NASTRAN linkage editor can be found in the NASTRAN Thirteenth Quarterly Report.

# SECTION 7 - PROGRAM DOCUMENTATION

The NASTRAN documentation consists of four manuals: The NASTRAN Theoretical Manual (NASA SP221), The NASTRAN User's Manual (NASA SP222), The NASTRAN Programmer's Manual (NASA SP223), and The Report of Demonstration Problems (NASA SP224).

The style of the manuals has been designed to accommodate future additions and modifications. Each major subsection stands alone with its own page numbers, equation numbers and figure numbers, so that new sections can be added without significant disruption.

## Theoretical Manual

The Theoretical Manual (500 pages), written for the serious structural analyst, explains the analytical and numerical procedures that underlie the program by presenting derivations and explanations to help him assess NASTRAN's suitability for the needs of his particular problem. Chapter titles are:

Executive System
Matrix Operations
Static Analysis
Elastic Elements
Differential Stiffness and Buckling
Dynamic Analysis Organization
Eigenvalue Extraction
Transient Analysis
Frequency Response and Random Analysis
Computer Graphics
Special Modeling Techniques
Error Analysis

The selection of material for the Theoretical Manual was not an easy task because not everyone has the same concept of what the word "theory" means when it is applied to a computer program. For some, theory is restricted to include only the formulation of the equations that will be solved; for others, theory also includes the development of the procedures, or algorithms, that will be used in the solution; still others regard the organization of the program and the flow of data through the computer as important theoretical topics.

A broad view concerning the selection of material was adopted, and the reader will find that all of the above aspects of the program are treated. Some structural analysts may be surprised at the emphasis on program organization and data processing, particularly in the early sections of the manual. These subjects are emphasized because they are vitally important to the success of a large computer program and should not be taken for granted.

In regard to the more mathematical subjects, such as the derivation of the equations for structural elements, and the development of eigenvalue extraction procedures, the reader will find that the level of sophistication is geared to the difficulty of the subject matter. Thus, it is assumed that a reader with an interest in an advanced topic (such as shell elements) will have the necessary theoretical background. In most cases the derivations are intended to be complete and rigorous. For a few of the structural elements, the reader is referred to the Programmer's Manual for the detailed expression of matrix coefficients that are regarded as too cumbersome to have general interest.

The Theoretical Manual is divided into fifteen major sections and numerous subsections. The first two major sections deal with some of the organizational aspects of NASTRAN and with utility matrix routines. Sections 3 through 8 deal with static structural analysis. It will be noted that no material has been included in Sections 4, 6 and 8, which are reserved for static analysis by the matrix force method, when it has been completed and incorporated into

NASTRAN. Sections 9 through 12 deal with dynamic structural analysis. Sections 13 through 15 deal with miscellaneous topics, including computer graphics, special structural modeling techniques, and error analysis.

The Programmer's Manual

The Programmer's Manual is divided into six major sections: Section 1, NASTRAN Programming Fundamentals; Section 2, Data Block and Table Descriptions; Section 3, Subroutine Descriptions; Section 4, Module Functional Descriptions; Section 5, NASTRAN - Operating System Interfaces; and Section 6, Modifications and Additions to NASTRAN. This manual describes rhetorically, the design and features of the program that are quite distinct from the source coding statements. These statements may be obtained by printing source library.

Section 1 is a general overview of the program, and as such it should be read as background material for all sections which follow.

Section 2 contains descriptions of the data blocks, which are the principal means of data communication between the program's functional modules (a module is defined to be a group of subroutines which perform a specific function) and the NASTRAN Executive System. Two indexes for the data block descriptions, one sorted alphabetically on data block names and the other sorted alphabetically on the names of the modules from which the data blocks are output, are given in Sections 2.2.1 and 2.2.2 respectively. Section 2 also contains (a) descriptions of tables, both core and noncore resident, maintained by the NASTRAN Executive System and (b) descriptions of miscellaneous tables which are accessed by a class of modules. Alphabetical indexes for these tables are given at the beginning of Sections 2.4 and 2.5 respectively.

Sections 3 and 4 contain descriptions of the (utility or general purpose) subroutines and modules of NASTRAN respectively. The reader is directed to the alphabetical indexes, sorted on entry point names, in Sections 3.2 and 4.1.3 respectively for these sections. An index to the Module Functional Descriptions, sorted alphabetically on module names, is given in Section 4.1.2. The reader is urged to read the introductory material to Sections 3 and 4 before using these sections.

Section 5 treats computer- and operating-system-dependent matters such as operating system control cards and generation of the absolute (executable) NASTRAN system.

Section 6 describes the means by which modifications and additions to NASTRAN are implemented.

### The User's Manual

The procedures for defining and loading a structural model are described in Section 1, which contains a functional reference for every card that is used for structural modeling.

The NASTRAN Data Deck, including the details for each of the data cards, is described in Section 2. This section also discusses the NASTRAN control cards that are associated with the use of the program.

NASTRAN contains twelve separate problem solution sequences, called rigid formats. Each of these rigid formats is associated with the solution of problems for a particular type of static or dynamic analysis. Section 3 contains a general description of rigid format procedures, along with specific instructions for the use of each rigid format.

The procedures for using the NASTRAN plotting capability are described in Section 4. Both deformed and undeformed plots of the structural model are available. Response curves are also available for transient response and frequence response analyses.

In addition to the rigid format procedures, the user may choose to write his own Direct Matrix Abstraction Program (DMAP). This procedure permits the user to execute a series of matrix operations of his choice along with any utility modules or executive operations that he may read. The rules governing the creation of DMAP programs are described in Section 5.

The NASTRAN diagnostic messages are documented and explained in Section 6. The NASTRAN dictionary, in Section 7, contains descriptions of mnemonics, acronyms, phrases, and other commonly used NASTRAN terms.

Sample problems are not included in the User's Manual. However, a set of twenty demonstration problems, at least one for each of the twelve rigid formats, are described in the Report of the Demonstration Problems. The data decks are available on tape, in the form of a User's Master File, for each of the computers on which NASTRAN has been implemented. NASTRAN control cards required to retrieve these problems from the tape are also available.

Samples of the printer output and of structure plots and response plots can be obtained by executing these demonstration problems.

## Demonstration Problem Report

The Demonstration Problem Report consists of twelve sections, one for each of the twelve NASTRAN rigid formats. Within each section the problems pertaining to that rigid format are described. For each problem there is a brief description of the model and comparisons between either analytic or published solutions and the NASTRAN solution. There is a table at the front of the report relating NASTRAN features (such as nonlinear materials) to the demonstration problem that demonstrates this feature. As indicated above, the actual input data for these problems is not published here but resides on a separate magnetic tape available with the NASTRAN program.

# SECTION 8 - RECOMMENDATION FOR NASTRAN'S FURTHER DEVELOPMENT

If NASTRAN is to continue to meet the objectives of a standardized, convenient, production tool, certain provisions for its future development should be considered. Obviously a program of this size requires a continuing and orderly maintenance system. In addition, however, it is also sensible to consider numerous enhancements to the program.

There are many areas where immediate upgrades to NASTRAN could be made to improve its overall efficiency and usefulness to NASA. A partial list along with potential savings in computer time is given below. This list does not include new features such as hydroelastic elements, solid elements, or three dimensional elements, since these are extensions of capability, rather than immediate enhancements.

- A symmetric decomposition option could be added to the determinant method of real eigenvalue extraction. This should improve determinant method performance by a factor of four.
- The matrix subroutine ADD could be rewritten to handle multiple (at least 3) input files. This would speed up certain dynamic problems where the operation of a triple sum must be repeatedly computed as in modules CEAD (Determinant Method) and FRRD.
- The Coupled Transient Analysis module repeatedly opens and closes several files. Substantial savings in time could be realized by reprogramming to eliminate these file operations.
- Addition of a sparse matrix transposition technique using nonzero packing and sorting would result in substantial time savings for sparse transposition problems.

- A core version of GINO, the General Input/Output utility, would speed up small dynamics problems by factors of ten or more.
- The ability to concatenate old problem tapes from several runs would allow the user more flexibility. This would be particularly important for modal formulations.
- The CDC 6400/6000 version of the program should be converted to single-precision. This conversion would result in a reduction by a factor of two in core usage and a factor of six in problem execution times.
- The IBM 360 version needs a non-FORTRAN read/write routine. This should reduce the running times by as much as a factor of four.

## SECTION 9 - BIBLIOGRAPHY

1. S. J. Garvey, "The Quadrilateral Shear Panel," Aircraft Engineering (May 1951), 134.
2. R. W. Clough and J. L. Tocher, "Finite Element Stiffness Matrices for Analysis of Plate Bending." Proc. of Conf. on Matrix Methods in Structural Mechanics, Air Force Flight Dynamics Lab. Report AFFDL-TR-66-80, December 1965.
3. R. H. Mallett and E. Helle, "Formulation and Evaluation of a Toroidal Ring Discrete Element." Bell Aerosystems Corp. Report No. 9500-941-003, June 1966.
4. E. Helle, "Formulation and Evaluation of a Triangular Cross-Section Ring Discrete Element." Bell Aerosystems Corp. Report No. 9500-941001, May 1966.
5. R. H. Mallett and S. Jordan, "Formulation and Evaluation of a Trapezoidal Cross-Section Ring Discrete Element." Bell Aerosystems Corp. Report No. 9500-941-004, November 1966.
6. W. Givens, "Numerical Computation of the Characteristic Values of a Real Symmetric Matrix." Oak Ridge National Lab., ORNL-1574, 1954.
7. J. W. Wilkinson, "The Algebraic Eigenvalue Problem," New York: Oxford Univ. Press, 1965, pp. 506-510.
8. J. G. F. Francis, "The QR Transformation, a Unitary Analogue to the LR Transformation," Computer J., Vol. 4, No. 3 (October 1961) and No. 4 (January 1962).
9. J. M. Ortega and H. F. Kaiser, "The $LL^T$ and QR Methods for Symmetric Tridiagonal Matrices," Computer J., Vol. 6, No. 1 (January 1963), 99-101.

10. R. H. MacNeal, "NASTRAN Theoretical Manual."

11. C. W. McCormick, "NASTRAN User's Manual."

12. F. J. Douglas, "NASTRAN Programmer's Manual."