NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Report 32-1240

ELAS—A General-Purpose Computer Program for the Equilibrium Problems of Linear Structures

Volume II. Documentation of the Program

Senol Utku

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

September 15, 1969

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Report 32-1240

ELAS—A General-Purpose Computer Program for the Equilibrium Problems of Linear Structures

Volume II. Documentation of the Program

Senol Utku

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

September 15, 1969

Prepared Under Contract No. NAS 7-100
National Aeronautics and Space Administration

Preface

The work described in this report was performed by the Engineering Mechanics Division of the Jet Propulsion Laboratory.

The ELAS program was developed by Dr. Senol Utku and Dr. Fevzican A. Akyuz, and is dedicated to the memory of Professor M. Inan of the Technical University of Istanbul.

Acknowledgment

The author is indebted to Vivia Crew for her help in editing all documents related with the ELAS program.

Contents

I. Introduction 1

II. Theoretical Background 3

A. Mathematical Formulation 3

B. Numerical Solution Method Based on the Extremum Formulation 4

C. The Program 6

III. COMMON Variables and COMMON Blocks of the Program 11

Appendix. Corrigenda for Volume I 21

References 23

Bibliography 23

Tables

III-1. Sequence and descriptions of COMMON blocks 13

III-2. Alphabetical listing of the constituents of COMMON block group 1 15

III-3. Meanings of the entries of important vectors 17

III-4. Alphabetical list of additional COMMON variables for Link 4 19

III-5. List of additional COMMON variables for Link 4 19

Figures

II-1. Definition of E_{ltj}, augmented matrix 7

II-2. Flow diagram corresponding to the summations implied by Eq. (25) 8

III-1. Memory organization for the four links of ELAS 12

Abstract

A general-purpose digital computer program (named ELAS) for the in-core solution of linear equilibrium problems of structural mechanics is described for potential and actual users in Volume I of this report, and is documented in Volume II. The program requires minimum input for the description of the problem. The solution is obtained by means of the displacement method and the finite element technique. Almost any geometry and structure may be handled because of the availability of lineal, triangular, quadrilateral, tetrahedral, hexahedral, conical, triangular torus, and quadrilateral torus elements. The assumption of piecewise linear deflection distribution insures monotonic convergence of the deflections from the stiffer side with decreasing mesh size. The stresses are provided by the best-fit strain tensors in the least-squares sense at the mesh points where the deflections are given. The selection of local coordinate systems whenever necessary is automatic. The core memory is efficiently used by means of dynamic memory allocation, an optional mesh-point relabelling scheme, imposition of the boundary conditions during the assembly time, and the straight-line storage of the rows of the stiffness matrix within variable bandwidth and the main diagonal. The number of unsuppressed degrees of freedom that can be handled in a given problem is 500 to 600 for a typical structure, but might far exceed these average values for special types of problems; the execution time of such problems is about four minutes in 32K IBM 7094 Model I machines. The program is written in FORTRAN II language. The source deck consists of about 8000 cards and the object deck contains about 1400 binary cards. The physical program (standard ELAS) is available from COSMIC, the agency for the distribution of NASA computer programs.

I. Introduction

Volume I, *User's Manual*, of this report gives a general description of ELAS,* a general-purpose digital computer program for the in-core solution of linear equilibrium problems of structural mechanics, and contains the information necessary for input preparation, arrangement of the physical program, and interpretation of output and error messages.

Volume II, *Documentation of the Program*, is published in two parts: the present volume—the basic Volume II—which gives the theoretical background of the program and contains tables and figures describing the COMMON variables, their meanings, and their arrangement in COMMON; and an Addendum to Volume II, which contains program descriptions, flowcharts, and source program listings for all program elements of ELAS/Level 3. (The original version of the ELAS program made available from COSMIC** in April 1968 is designated ELAS/Level 0. Subsequent program corrections made in January 1969, March 1969, and May 1969 updated the program to ELAS/Level 1, ELAS/Level 2, and ELAS/Level 3, respectively.)

In addition to the list of references cited in the text, a list of documented works related with the development of the ELAS program is given in the bibliography. A corrigenda for Volume I is given in the Appendix.

*First two syllables of the word Elasticity.

**Computer Software Management and Information Center, Computer Center, University of Georgia, Athens, Georgia, 30601, telephone 404-542-3265.

II. Theoretical Background

This section summarizes the mathematical formulation, the numerical method of solution, and the design features of the program.

A. Mathematical Formulation

Let V denote the material volume of the structure within the closed boundary S. Let $x_\alpha, \alpha = 1, 2, 3,$ denote a fixed right-handed Cartesian coordinate system. The Greek subscripts always refer to these axes; therefore, $\sigma_{\alpha\beta}$ is the stress tensor described in such a coordinate system. Let u_α denote the displacement vector; $\tilde{p}_\alpha$ the body force; m the unit mass; double dots above, the second time derivative; and comma in the subscript the partial differentiation with respect to the space variable represented or implied by the subscript following the comma. If it is assumed that repeated subscripts imply summation over the range, the equilibrium of any particle within V may be expressed as

$$\sigma_{\beta\alpha,\beta} + \tilde{p}_\alpha = m\ddot{u}_\alpha \tag{1}$$

In the equilibrium problems, the loading is such that

$$\ddot{u}_\alpha = 0 \tag{2}$$

Therefore, substitution of $\ddot{u}_\alpha$ from Eq. (2) into Eq. (1) yields

$$\sigma_{\beta\alpha,\beta} + \tilde{p}_\alpha = 0 \qquad \text{in } V \tag{3}$$

Let S' denote the portion of S where the tractions are prescribed. The equilibrium condition on S' is

$$\sigma_{\alpha\beta}\, n_\beta + p_\alpha = 0 \tag{4}$$

where n_β is the unit normal vector and p_α is the prescribed traction. The stress–strain relationship of the material is

$$\sigma_{\alpha\beta} = D_{\alpha\beta\gamma\delta}\,(\epsilon_{\gamma\delta} - \epsilon^0_{\gamma\delta}) \tag{5}$$

where $\epsilon^0_{\gamma\delta}$ is the prescribed strain tensor, $\epsilon_{\gamma\delta}$ is the strain tensor, and $D_{\alpha\beta\gamma\delta}$ is the material matrix, which is positive-definite and symmetrical, so that

$$D_{\alpha\beta\gamma\delta} = D_{\gamma\delta\alpha\beta} = D_{\beta\alpha\gamma\delta} = D_{\alpha\beta\delta\gamma}$$

The strain displacement relationships are

$$\epsilon_{\alpha\beta} = \frac{1}{2}(u_{\alpha,\beta} + u_{\beta,\alpha}) \tag{6}$$

Finally, the displacement boundary conditions may be stated as

$$u_\alpha = u_\alpha^0 \qquad \text{on } S'' \tag{7}$$

where u_α^0 denotes the prescribed displacements on S''. It should be noted that

$$S' + S'' = S \tag{8}$$

In an equilibrium problem, usually V, S', S'', p_α, u_α^0, $\tilde{p}_\alpha$, $D_{\alpha\beta\gamma\delta}$, n_β, and $\epsilon_{\gamma\delta}^0$ are given and u_α, $\epsilon_{\alpha\beta}$, and $\sigma_{\alpha\beta}$ are requested.

Equations (3) through (8) constitute the differential equation formulation of the equilibrium problem in three-dimensional continuum. A finite difference solution based on this formulation may be set up as follows. A regular mesh is placed in V such that S' and S'' are determined by the mesh points. If S is not defined by coordinate surfaces, such representation of S' and S'' is only approximate. The displacements u_α at the mesh points in V and on S are taken as the primary unknowns. The prescribed u_α^0 displacements are assigned to the mesh points of S''. With the use of Eq. (6), $\epsilon_{\alpha\beta}$ at the mesh points in V and on S are approximated by the first differences of u_α and u_α^0. Then, by the use of Eq. (5), the values of $\sigma_{\alpha\beta}$ are expressed at the mesh points. Finally, depending upon the mesh point in V or on S', Eq. (3) or Eq. (4), respectively, is used to write the difference equations for the unknown displacements. After the unknown displacements from these equations have been computed, the strains and the stresses may be computed from the finite difference approximations of Eq. (6) and Eq. (5). Such a solution method has the following drawbacks:

(1) To minimize the truncation errors, a regular mesh in V is used; however, this causes approximate representation of boundary S and, therefore, increases the truncation errors in the finite difference approximations of Eqs. (4) and (7). Since the errors in the finite difference approximations of Eqs. (4) and (7) dominate in the solution more (Ref. 1) than the errors in the finite difference approximation of Eq. (3), either an irregular mesh in V may be considered to represent S more accurately, or higher-order formulas for the boundary conditions are used, although neither scheme is desirable in a general-purpose digital computer program.

(2) Because of the symmetry and the positive-definiteness of $D_{\alpha\beta\gamma\delta}$, the formulation given by Eqs. (3) through (8) is self-adjoint and positive-definite. However, the coefficient matrix of the unknown displacements in the finite difference equilibrium equations is, in general, neither symmetric nor positive-definite. The loss of the two desirable qualities of the problem in the numerical formulation increases the storage requirements and solution time. Because of these setbacks, the mathematical formulation given by Eqs. (3) through (8) is modified slightly as explained in the following paragraph.

Consider the quantity π defined as

$$\pi = \frac{1}{2}\int_V \epsilon_{\alpha\beta}\,\sigma_{\alpha\beta}\,dV - \int_V u_\alpha \tilde{p}_\alpha\,dV - \int_{S'} u_\alpha p_\alpha\,dS \tag{9}$$

where dV is the volume element, dS the area element, and the other symbols are as previously defined. Consider the displacement fields satisfying Eq. (7). For each such displacement field, by means of Eqs. (9), (6), and (5), a scalar π may be computed. It can be shown that, for sufficiently smooth displacement fields satisfying Eq. (7), the stationary point of π, i.e., the point for which

$$\delta\pi = 0 \tag{10}$$

also satisfies Eqs. (3) and (4). In fact, by the methods of calculus of variations, Eq. (10) yields Eq. (3) as the Euler differential equation, and Eq. (4) as the additional boundary condition. Therefore, Eq. (10) is an equivalent statement of Eqs. (3) and (4). The quantity π is known as the "total potential energy" of the system. Thus, the formulation given by Eq. (10) reduces the problem to that of locating the stationary point of the total potential energy functional. How the numerical solution is set up from this formulation (which is sometimes referred to as the extremum formulation of the problem), and its advantages, are discussed in the next subsection.

B. Numerical Solution Method Based on the Extremum Formulation

A random mesh is placed in V such that the mesh elements are line segments, triangles, quadrilaterals, tetrahedrons, hexahedrons, conical segments, or triangular or quadrilateral tori. Some of the mesh elements are shown in Fig. III-1 (Vol. I). The types of mesh elements that may be used in different structures are given in Table III-2 (Vol. I). The randomness of the mesh enables the selection of mesh points that are exactly on the boundary S. For clarity, the mesh points are labelled sequentially, with integer numbers starting from 1. If there are $\underline{s}$ number of mesh points, there are $\underline{s}!$ different types of possible labelling. In the discussion that follows it will become obvious that some of these labelling systems are more desirable than others.

It is assumed that one of the possible $\underline{s}!$ systems is selected. Next, the mesh elements are labelled sequentially with integers. If there are $\underline{m}$ number of mesh elements, there are $\underline{m}!$ number of different labelling systems. It is supposed that one of the possible $\underline{m}!$ systems is selected. In what follows, superscript m indicates the element label, and subscripts t or s indicate the mesh point label. To solve the equilibrium problem formulated in Section II-A numerically, instead of computing u_α at every point of V and S, an attempt is made to find, at the mesh points, certain related quantities that define the distorted configuration of the structure in the same way as u_α. These quantities are called deflections, which are displacements/rotations at the mesh points. Given a mesh point, the total number of independent deflection components is the number of degrees of freedom of that mesh point. In Table III-1 (Vol. I), the deflection components at a mesh point of different structures are shown as referred to an overall coordinate system. Let $\underline{k}$ denote the number of degrees of freedom at a mesh point of a structure. The value of $\underline{k}$ for different structures is given in the last column of Table III-1 (Vol. I). It will be assumed that the deflection components at a mesh point are ordered as shown in the table. The subscripts k and l will be used to indicate the sequence number implied by this ordering. If a prime appears on k or l, this implies that a local coordinate system is used in defining the degree-of-freedom directions. Let q_{ks} denote the kth deflection component at mesh point s. A mesh element may be defined by the mesh points that are coincident with its vertices. For clarity in referencing, the convention of Table III-5 (Vol. I) is adopted in ordering the vertices of mesh elements. The type numbers shown in this table refer to the numbers shown on the shaded squares of Table III-2 (Vol. I). Subscripts g and h will be exclusively used to denote the sequence number of a vertex in the $\underline{g}$ number of vertices of an element.

With the preceding definitions, the method used to obtain the stationary point of the total potential energy functional may now be explained. This is the classical Ritz procedure (Ref. 2), where the undetermined parameters of the problem are the unknown components of q_{ks} deflections. Equation (9) is first written as

$$\pi = \frac{1}{2}\int_{V^m} \epsilon_{\alpha\beta}\,\sigma_{\alpha\beta}\,dV^m - \int_{V^m} u_\alpha \tilde{p}_\alpha\,dV^m - \int_{S'^m} u_\alpha p_\alpha\,dS'^m \tag{11}$$

where the repeated index implies the summation over the range. Next, an attempt is made to select a family of displacement fields that are sufficiently smooth, but otherwise arbitrary, ignoring for the time being the essential boundary conditions of Eq. (7). A piecewise linear displacement field is acceptable in this sense (Ref. 3). Of course there are other piecewise continuous fields that are acceptably smooth. However, to simplify the understanding of the procedure, it is assumed that the displacement fields are piecewise linear. Such a field may be described mathematically for the mth element in terms of the deflections of its vertices as

$$u_{\alpha'} \approx B_{\alpha'\beta'l'h}\,\tilde{Q}_{l'l}\,\mu^m_{ht}\,q_{lt}\,x_{\beta'} + \text{rigid body movement} \tag{12}$$

where the primes indicate the local coordinate system of the element. The coefficients μ^m_{ht} constitute a binary array such that, for a given m and h, it is zero throughout the range of t, but 1 at the value of t corresponding to the hth vertex of the mth element. In fact, $\mu^m_{ht}\,q_{lt}$ is the list of deflection components pertaining to the vertices of the mth element. The matrix $\tilde{Q}_{l'l}$ in Eq. (12) is the coordinate transformation matrix, where, for fixed l', it represents the direction cosines of the local axis related with l' degree-of-freedom direction in the coordinate system associated with l. The space variable $x_{\beta'}$ is the distance measured along the β'th local axis. The coefficients $B_{\alpha'\beta'l'h}$ may be computed from the local coordinates of the vertices of the mth element. In Table III-3 (Vol. I), for different mesh elements, the orientation of the local coordinate system relative to the overall coordinate system is given. It should be noted that Eq. (12) is an approximation of the true displacements in the mth element, even if the exact values of deflection components q_{lt} are known. However, it may be shown that the error decreases with decreasing mesh size. With the use of $u_{\alpha'}$ from Eq. (12) in Eq. (6), the strains in the mth element, as referred to the local coordinate system of the mth element, may be expressed as

$$\epsilon_{\alpha'\beta'} = B_{\alpha'\beta'l'h}\,\tilde{Q}_{l'l}\,\mu^m_{ht}\,q_{lt} \tag{13}$$

Let $D_{\alpha'\beta'\gamma'\delta'}$ denote the material constants of the mth element as referred to the local coordinate system of the mth element. If

$$K^m_{kglh} = \tilde{Q}_{k'k}\left(\int_{V^m} B_{\delta'\gamma'k'g}\,D_{\delta'\gamma'\alpha'\beta'}\,B_{\alpha'\beta'l'h}\,dV\right)\tilde{Q}_{l'l} \tag{14}$$

and

$$P_{kg}^{m} = \tilde{Q}_{k'k}\left(\int_{V^m} B_{\alpha'\beta'k'g}\, x_{\beta'}\, \tilde{p}_{\alpha'}\, dV + \int_{S'^m} B_{\alpha'\beta'k'g}\, x_{\beta'}\, p_{\alpha'}\, dS\right) \tag{15}$$

are defined, π of Eq. (11) may be expressed as

$$\pi = \frac{1}{2}\, q_{ks}\, \mu_{gs}^{m} K_{kglh}^{m}\, \mu_{ht}^{m}\, q_{lt} - q_{ks}\,(\mu_{gs}^{m} P_{kg}^{m} + Q_{ks}) \tag{16}$$

where Q_{ks} denotes the prescribed concentrated loads at the mesh points. The deflection components q_{ks} (or q_{lt}) should be such that, on S', they satisfy the essential boundary conditions of Eq. (7). Let d_j denote the portion of q_{lt}, which is unknown. The essential boundary conditions may be expressed as

$$q_{lt} = e_{ltj} d_j + e_{lt}^{0} \tag{17}$$

where the coefficients e_{ltj} and e_{lt}^{0} are quantities that may easily be determined from Eq. (7). For example, if there are no prescribed deflections in the problem, e_{ltj} is a binary array containing only one 1 in the whole range of j for a given lt, and e_{lt}^{0} is zero throughout. Actually, the d_j are the true undetermined parameters of the problem. If q_{lt} is substituted from Eq. (17) into Eq. (16), the values of d_j corresponding to the stationary point of π may be obtained from the set of linear equations

$$\pi_{,d_j} = 0 \tag{18}$$

since

$$\delta\pi = \pi_{,d_j}\, \delta d_j \tag{19}$$

The equations given by Eq. (18) may be rewritten as

$$\mathcal{K}_{ij}\, d_j = \mathcal{P}_i \tag{20}$$

where

$$\mathcal{K}_{ij} = e_{ksi}\, \mu_{gs}^{m} K_{kglh}^{m}\, \mu_{ht}^{m} e_{ltj} \tag{21}$$

and

$$\mathcal{P}_i = e_{ksi}\,(\mu_{gs}^{m} P_{kg}^{m} + Q_{ks}) - e_{ksi}\, \mu_{gs}^{m} K_{kglh}^{m}\, \mu_{ht}^{m} e_{lt}^{0} \tag{22}$$

The coefficients K_{kglh}^{m} constitute the element stiffness matrix of the mth element and P_{kg}^{m} is called the mth element load vector. In Eq. (20), the coefficient matrix $\mathcal{K}_{ij}$ is the stiffness matrix associated with the directions of d_j deflections, and the right-hand-side vector $\mathcal{P}_i$ lists the loads in these directions. Equations (21) and (22) indicate how the coefficient matrix and the right-hand-side vector of the governing equations can be systematically generated from the element stiffness matrices and load vectors.

The operation implied by Eqs. (21) and (22) is referred to as the assembling of the elemental matrices. Because of the positive-definiteness and the symmetry of $D_{\alpha\beta\gamma\delta}$, Eq. (21) shows that

$$\mathcal{K}_{ij} = \mathcal{K}_{ji} \tag{23}$$

and $\mathcal{K}_{ij}$ is also positive-definite. Once the unknown deflections d_j are solved from Eq. (20), the complete deflections q_{lt} are obtained by substituting d_j into Eq. (17). After deflections q_{ks} have been computed, the strains and the stresses at the mesh points may be computed as described in Ref. 4.

The method of solution described in the preceding has the following advantages:

(1) Since the mesh is random, the boundaries S' and S'' may be closely approximated, and thus minimize truncation errors.

(2) Any *a priori* knowledge about the variation of u_α in V or on S may be used advantageously by varying the mesh size accordingly to minimize the truncation errors.

(3) The self-adjoint character, as well as the positive-definiteness of the problem, is preserved, since $\mathcal{K}_{ij}$ is always symmetric and positive definite.

C. The Program

1. Criteria for Storage Allocation. All the input data previously mentioned (see Fig. IV-1, Vol. I) are stored permanently in COMMON after their validity is checked. No fixed-length block is assigned to these diversified data. The data are compactly stored as a string of variable length. This enables the program to compete in obtaining priorities efficiently in a multiprogramming environment. After reading the control card, the program determines the pointers of each of the data blocks relative to the beginning of COMMON. When the other input data become available, they are placed in the proper place in COMMON by the pointers. Although the locations of the data blocks vary from one job to another, the locations of the pointers and the control information provided by the control card have fixed locations at the beginning of COMMON. The remainder of the core is assigned for the program instructions and temporary storage for the coefficient matrix and the right-hand-side vector of the governing equations. The program consists of four links, and since all the program instructions are not required simultaneously, only the instructions for each link in turn as needed are retained in the core.

A sketch of the governing equations in Eq. (20) is given in Fig. II-1 (Vol. I). Since the coefficient matrix is symmetric, the program allows storage only for the shaded area shown in the figure. From Eq. (20) it may be observed that, for a fixed j, $\mathcal{K}_{ij}$ represents a vector listing the forces that may develop when a unit deflection is applied in the fixed j direction, while keeping all the other degree-of-freedom directions with zero deflections. The nonzero entries of this vector coincide with the deflection directions of only those mesh points that are connected with the disturbed mesh point by means of mesh elements and deflection boundary conditions. This shows that the $\mathcal{K}_{ij}$ matrix is sparse and usually has a large zero area in the upper right-hand corner.

Before generating the coefficient matrix and the right-hand side of Eq. (20), the program computes a pointer for each of the rows of the coefficient matrix, and a pointer for the right-hand-side vector so that the coefficients shown in the shaded area in Fig. II-1 (Vol. I) can be stored compactly in COMMON as a string. Actually, the pointers of the rows are the addresses of the words immediately preceding the diagonal elements. As discussed in Ref. 5, by the proper ordering of d_j unknowns in Eq. (20), the zero area may be increased in the upper right-hand corner of $\mathcal{K}_{ij}$. If the user chooses to assign zero into the ISHUF field of the control card, the unknowns are ordered as implied by the mesh-point labels; e.g., the unknown deflection components of the first mesh point are placed first, those of the second mesh point are placed second, etc. If the user assigns ISHUF = 1 or 2, the program first tries to find a better labelling system with the method given in Ref. 5, and uses these new mesh-point labels in ordering the unknowns d_j. For example, if mesh point with label 25 is the first mesh point in the new labelling system, the unknowns of this mesh point are listed first in d_j. If the user assigns ISHUF = 3, the better labelling system is required by the program from input data cards (number 17 in Fig. IV-1, Vol. I). The method for relabelling described in Ref. 5 requires the generation of the mesh-point connectivity matrix N_{st}, which is a binary matrix. If mesh point s is connected to mesh point t by a mesh element or by a deflection boundary condition, $N_{st} = N_{ts} = 1$; otherwise, $N_{st} = N_{ts} = 0$. It is always assumed that a mesh point is connected to itself.

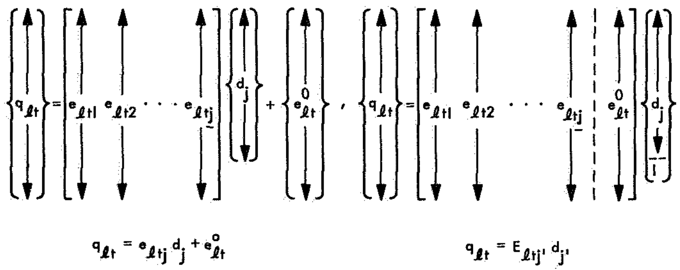

Fig. II-1. Definition of $E_{ltj'}$ augmented matrix

If point s is completely constrained by the deflection boundary conditions, $N_{st} = N_{ts} = 0$ for all t, except $t = s$. The program generates N_{st} from the information provided by the element description data and deflection boundary conditions. The connectivity matrix N_{st} is always generated, since it is also used in determining the pointers of the rows of $\mathcal{K}_{ij}$.

2. *Method of Assembly.* To obtain the coefficient matrix and the right-hand-side vector of the governing equations, the mesh elements are processed, one at a time, first to obtain the element stiffness matrix and the element load vector for each, and then to assemble these according to Eqs. (21) and (22) and the allocated storage. Let A^r denote the vector in COMMON and R_i denote the pointer of the ith equation in Eq. (20). Let us assume that the right-hand-side vector is stored after the coefficient matrix. Let $E_{ksi'}$ (or $E_{ltj'}$) denote the augmented matrix composed of e_{ksi} (or e_{ltj}) and e^0_{ks} (or e^0_{lt}), as shown in Fig. II-1. Let a prime on the index indicate that the range of unprimed index is increased by 1, and let an underlined index indicate the largest value within the range. With this notation, the assembly procedure may be summarized as

$$A^r = \gamma^r_{i'} E_{ksi'} Q_{ks} + \gamma^r_{i'} E_{ksi'} \mu^m_{gs} P^m_{kg} + \delta^r_{i',j'} E_{ksi'} \mu^m_{gs} K^m_{kglh} \mu^m_{ht} E_{ltj'} \tag{25}$$

where

$$\left.\begin{aligned} \delta^r_{i',j'} &= 1 && \text{if } i' \leq j',\ j' \leq \underline{i}, \text{ and } r = R_{i'} + j' - i' \\ \delta^r_{i',j'} &= -1 && \text{if } j' > \underline{i},\ i' \leq \underline{i}, \text{ and } r = R_{\underline{i}} + i' \\ \delta^r_{i',j'} &= 0 && \text{for all other possibilities} \end{aligned}\right\} \tag{26}$$

and

$$\left.\begin{aligned} \gamma^r_{i'} &= 1 && \text{if } i' \leq \underline{i} \text{ and } r = R_{\underline{i}} + i' \\ \gamma^r_{i'} &= 0 && \text{for all other possibilities} \end{aligned}\right\} \tag{27}$$

In the program, only the nonzero $E_{ksi'}$ constants are computed by the deflection boundary condition input units and the connectivity matrix. For each ks the nonzero entries of $E_{ksi'}$ are stored with their values and i' indices. The values and the indices of nonzero Q_{ks} entries are directly provided by the concentrated load input cards. The binary coefficients $\delta^r_{i',j'}$ and $\gamma^r_{i'}$ are not stored, but determined from Eqs. (26) and (27). If m and g are given, the s value of the nonzero entry of μ^m_{gs} is obtained from the element description data. Let $\tilde{e}_a$ and i'_a denote the nonzero values and corresponding indices in $E_{ksi'}$ for a given ks, and let $\underline{a}$ denote the maximum value of a, so that $1 \leqq \underline{a} \leqq a$ (b and $\underline{b}$ are alternate symbols). Let $\underline{d}$ denote the number of concentrated load input units. This notation is used in the flow diagram corresponding to Eq. (25) given in Fig. II-2. In the ELAS program, the summations implied by the first term in the right-hand side of Eq. (25) are implemented in Link 1 and the remainder in Link 2.

3. Method of Solution of the Governing Equations. Since $\mathcal{K}_{ij}$ is a symmetric and positive-definite and band-limited matrix for the solution of Eq. (20), the Cholesky algorithm may be applied. In this method, the decomposed stiffness matrix B_{ij} from $\mathcal{K}_{ij}$ is first computed as

$$B_{ij'} B_{j'j} = \mathcal{K}_{ij} \tag{28}$$

where the range of j' equals that of i. Then, from

$$B_{ij} d'_j = \mathcal{P}_i \tag{29}$$

with a forward sweep, the auxiliary unknowns d'_j can be solved. Finally,

$$B_{ij} d_i = d'_j \tag{30}$$

yields the unknowns with a backward sweep. In Fig. II-1 (Vol. I) the border of the zero area in the upper right-hand corner is not always defined by the last nonzero coefficient in each equation. This is because the shaded areas of $\mathcal{K}_{ij}$ and B_{ij} are identical only when the border is selected as shown in Fig. II-1 (Vol. I). In the ELAS program, $\mathcal{K}_{ij}$ coefficients in the shaded area of the figure are first modified to those of the coefficients of B_{ij}, then $\mathcal{P}_i$ constants are changed to d'_i, and finally, d'_i values are converted to the numerical values of the d_j unknowns. Then, from Eq. (17), q_{lt} is computed on the same area as d_j. These operations are carried out in Link 3 of ELAS.

4. Computation of Stresses. The computation of stresses in displacement methods poses a harder problem in structures of two- or three-dimensional continuum than that in truss and frame structures, which truly have a finite num-

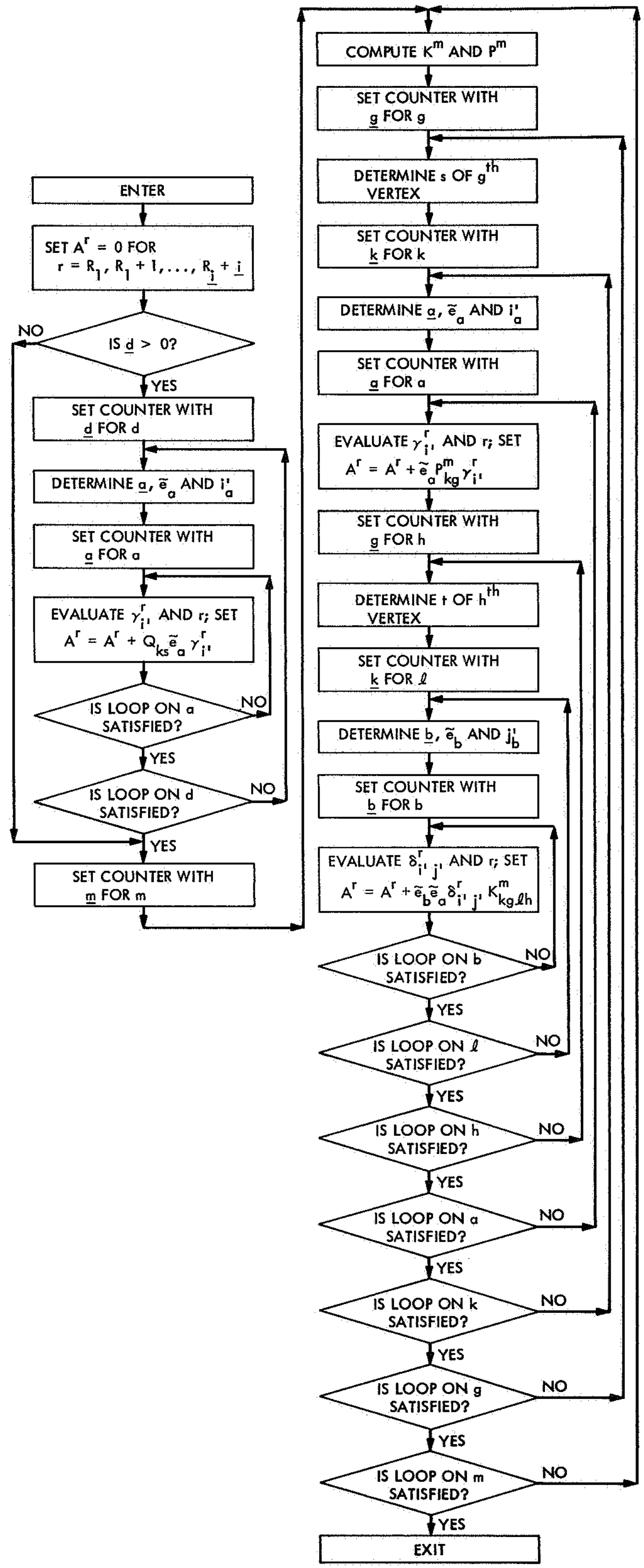

Fig. II-2. Flow diagram corresponding to the summations implied by Eq. (25)

ber of deflection components for the determination of their distorted configuration. The difficulty arises from the fact that the structures of two- or three-dimensional continuum actually have infinitely many deflection components, and the relations of the type of Eq. (12) are only approximate.

After computing the deflections as mesh functions, the problem of stress computation with acceptable accuracy in reasonable machine times still remains. Experience has shown that the use of Eq. (13) and then Eq. (6) presents the following drawbacks: (1) the exact location of the point for which the stresses are computed is not known, and (2) the computed stresses may be largely different from the actual stresses. Despite these setbacks, stress computation of this type is being widely used because it is modular in elements, just as is the generation of the governing equations in Eq. (20), a feature that facilitates automation. In the ELAS program, the best-fit stress computation method of Ref. 4 is used for structures of continuum. This method is just as easy to automate and has the following advantages: (1) stresses are computed at the points where the deflections are obtained, (2) the accuracy in stresses is comparable with that of deflections, and (3) the stress boundary conditions of Eq. (4) may be satisfied during the computation of the stresses of boundary points. This scheme was initially devised for triangular finite elements (Ref. 6).

In the following paragraphs the stress computation in structures of two- and three-dimensional continua is explained. The computation of stresses in structures composed of elements of one-dimensional continuum is performed by multiplying element stiffness matrices with computed deflections.

Mesh Line Set. Suppose that the deflections at the mesh points of a structure of three-dimensional continuum are available and that the stresses at mesh point s are requested. The question of how much deflection data should be included in the computation is of practical importance because the computation time rapidly increases with this quantity. Experience with the method of computing stresses in the element indicates that deflections of the set of elements meeting at mesh point s are sufficient for the computation of its stresses with acceptable accuracy. The mesh points of the element set are called "mesh-point set" and the mesh lines meeting at the common mesh point s are called "mesh-line set." The scheme adopted in ELAS is modular in the mesh-line set—the next-best unit after elements. The stress computation at a mesh point starts with the determination of the element set, and consequently, the mesh-line set associated with this mesh point. Then, if this mesh point is on the boundary, the average boundary surface area associated with this node and the direction cosines of the outer normal are computed.

Selection of Local Coordinate Systems at the Mesh Points. In a given problem, it is desirable to have one fixed, right-handed coordinate system to express the stresses. However, this is not practical for structures composed of anisotropic material, at the boundary points where the outer normal is not coincident with the coordinate lines, and for shell structures. The following method is adopted in the ELAS program for the selection of local coordinate systems at the mesh points.

At an internal node, the local axes may be taken as the material axes unless the material is isotropic, in which case they should be taken (1) parallel to the overall coordinate system in plates and three-dimensional solids, and (2) as the principal curvature directions and the normal of the middle surface, or any other suitable system that the user inputs in shells. At a boundary node, the first local axis may be coincident with the outer normal, and the directions of the remaining local axes may be determined so that (1) the local third axis becomes the middle surface normal in plates and shells, and (2) the direction defined by the cross-product of outer normal with the overall axis, which makes the largest angle with the outer normal, then becomes the second local axis in three-dimensional solid structures.

Stress Computation at an Internal Mesh Point. Let j be the label of a mesh line in the mesh-line set, with $\Delta\rho_{\gamma'}$ the position vector and $\Delta u_{\gamma'}$ the displacement vector of the far end of the mesh line relative to the mesh point where the local coordinates are defined. As the first approximation of the strain along the jth mesh line, the following may be written:

$$\epsilon = \frac{\Delta\rho_{\gamma'}\,\Delta u_{\gamma'}}{\Delta\rho_{\mu'}\,\Delta\rho_{\mu'}} \tag{31}$$

The same strain may be obtained from the strain tensor of mesh point s as

$$\epsilon = \frac{\Delta\rho_{\alpha'}\,\Delta\rho_{\beta'}\,\epsilon_{\alpha'\beta'}}{\Delta\rho_{\mu'}\,\Delta\rho_{\mu'}} \tag{32}$$

Equating Eq. (31) to Eq. (32) and cancelling the denominators results in the following expression:

$$(\Delta\rho_{\alpha'}\,\Delta\rho_{\beta'})_j\,\epsilon_{\alpha'\beta'} = (\Delta\rho_{\gamma'}\,\Delta u_{\gamma'})_j \tag{33}$$

The number of equations in Eq. (33) is equal to the range j. Usually the range of j is greater than the number of independent components of the strain tensor. (If not, the mesh may be readjusted by repeating the deflection computation.) Therefore, in Eq. (33), there are more equations than the unknown strain components. Such a set may be solved by least squares, first by multiplying both sides with the transpose of the coefficient matrix, then by inverting the new coefficient matrix. This leads to

$$\epsilon_{\alpha'\beta'} = [(\Delta\rho_{\alpha'}\,\Delta\rho_{\beta'})_i(\Delta\rho_{\delta'}\,\Delta\rho_{\nu'})_i]^{-1}\,(\Delta\rho_{\delta'}\,\Delta\rho_{\nu'})_j\,(\Delta\rho_{\gamma'}\,\Delta u_{\gamma'})_j \tag{34}$$

where the range of i is equal to that of j. The stresses at the mesh point may be obtained by substituting $\epsilon_{\alpha'\beta'}$ from Eq. (34) into Eq. (5). If the problem is a plane-strain problem, one should first impose

$$\epsilon_{3'3'} = \epsilon_{1'3'} = \epsilon_{3'1'} = \epsilon_{2'3'} = \epsilon_{3'2'} = 0 \tag{35}$$

on Eq. (33). For plane-stress problems, $D_{\alpha'\beta'\gamma'\delta'}$ in Eq. (5) should be modified to guarantee

$$\epsilon_{1'3'} = \epsilon_{3'1'} = \epsilon_{2'3'} = \epsilon_{3'2'} = \sigma_{3'3'} = 0 \tag{36}$$

For the bending of plates and shells, $\epsilon_{\alpha'\beta'}$ should be interpreted as curvature changes and, in Eq. (33), $(\Delta\rho_{\gamma'}\,\Delta u_{\gamma'})_j$ should be taken as the projection of the rotations vector of the far-end mesh point relative to the current mesh point in the jth mesh line on $\vec{i}_\xi \times \Delta\vec{\rho}$ direction, where $\vec{i}_\xi$ is the unit vector of the third local axis. Also, the conditions in Eq. (36) should be imposed on $D_{\alpha'\beta'\gamma'\delta'}$ and Eq. (5) should be replaced by

$$M_{\gamma'\delta'} = -\frac{t^3}{12} D_{\gamma'\delta'\alpha'\beta'}\,\epsilon_{\alpha'\beta'} \tag{37}$$

where $M_{\gamma'\delta'}$ denotes the bending moments and t is the thickness. The membrane case of shells is identical with the plane-stress case, provided that Eq. (5) is replaced with

$$N_{\gamma'\delta'} = t\,D_{\gamma'\delta'\alpha'\beta'}\,\epsilon_{\alpha'\beta'} \tag{38}$$

where $N_{\gamma'\delta'}$ denotes the membrane forces.

Stress Computation at a Boundary Mesh Point. The procedure for stress computation at a boundary mesh point is basically the same as the computation at an internal mesh point. Here, the stress boundary conditions, expressed in terms of the strains, are included in Eq. (33) before the application of the least-squares scheme for their solution. The stress boundary conditions may be written as

$$\frac{n_{\beta'}}{D_{1'1'1'1'}} D_{\alpha'\beta'\gamma'\delta'}\,\epsilon_{\gamma'\delta'} = \frac{-p_{\alpha'}}{D_{1'1'1'1'}} \tag{39}$$

where $p_{\alpha'}$ represents the prescribed boundary stresses. If at the boundary mesh point the deflections, in place of the stresses, are prescribed, $R_{\alpha'}$ reaction forces may be found from the equilibrium equations of the boundary node, and the following may be written:

$$p_{\alpha'} = -\frac{R_{\alpha'}}{A} \tag{40}$$

where A is the average boundary surface area associated with the mesh point. In Eq. (39), the reason for division by $D_{1'1'1'1'}$ is to reduce the coefficients of strains in the stress boundary equations to the same order of magnitude as those of Eq. (33). The procedure described here for a three-dimensional solid may be readily extended to other types of structures with the help of previous paragraphs.

III. COMMON Variables and COMMON Blocks of the Program

The memory organization in each of the four links of ELAS is illustrated in Fig. III-1. Table III-1 lists the blocks of COMMON sequentially, and gives a short description of each block. In the table, the variable-address blocks are listed with increasing COMMON addresses. It should be noted that the variable-address blocks in COMMON are packed in a string, one after the other, without any waste of core locations. Such blocks may be properly located by means of *pointers*, which are also in COMMON. A pointer is a word whose content is one less than the COMMON address of the first word of the associated COMMON block. The constituents of Block Group 1 are listed in Table VI-3 (Vol. I), in the order in which they appear in COMMON. These constituents are alphabetically ordered with their symbolic names in Table III-2. In Table III-3, the meanings of entries of important vectors, especially those defined by the pointers, are given. The additional COMMON variables of Link 4 are listed alphabetically in Table III-4, and with increasing COMMON addresses in Table III-5. Table III-5 also contains a short description of these variables.

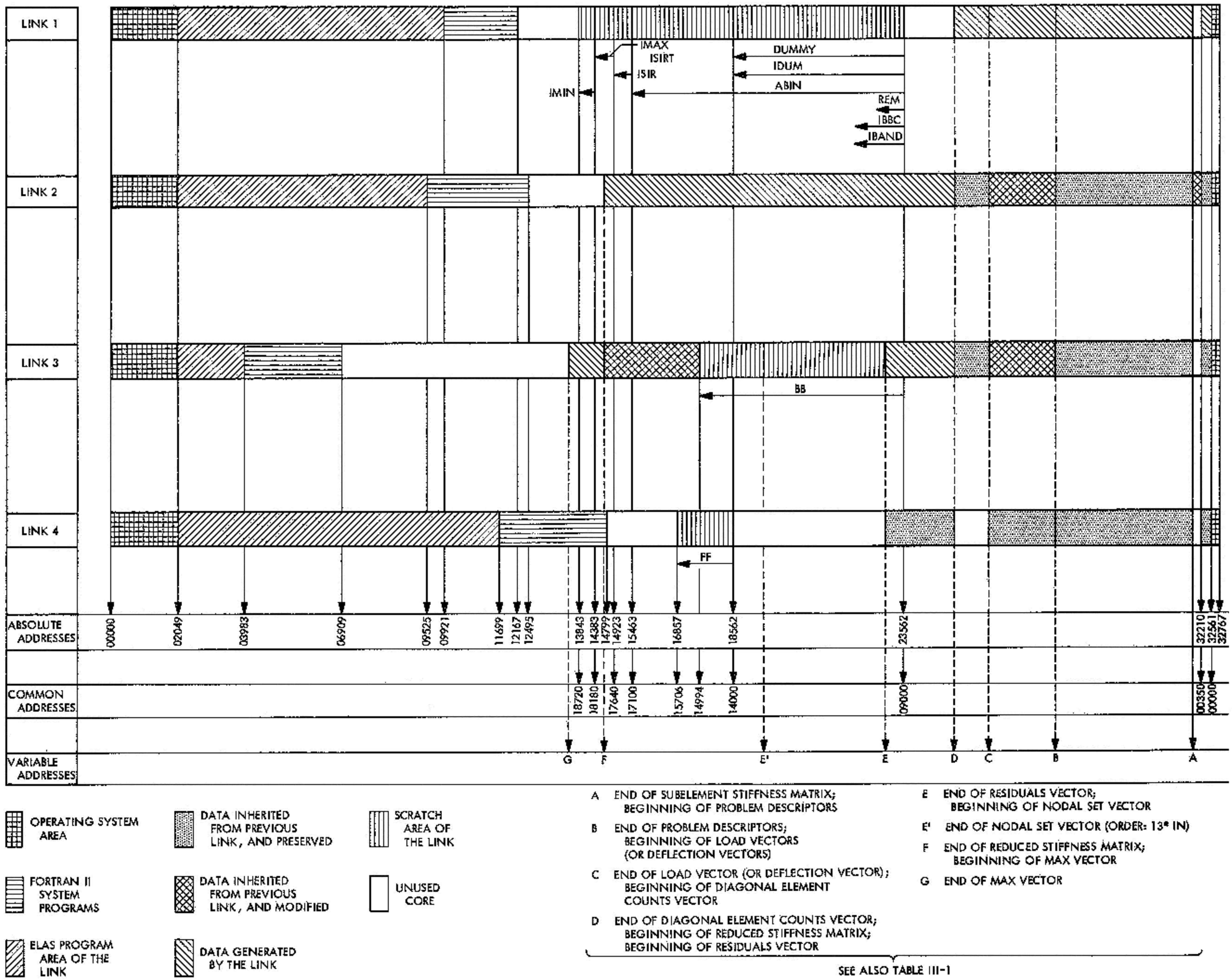

Fig. III-1. Memory organization for the four links of ELAS

Table III-1. Sequence and descriptions of COMMON blocks**

Block groups	Block name	Pointer of the block†	Common address of first word of block	Block length‡	Link 1 operation§	Link 2 operation§	Link 3 operation§	Link 4 operation§
1	Important constants△		1	17	G	I	I	I
	General descriptors I△		17	10		S		S
	Important constants and pointers I△		27	52	G	I	I	I
	General descriptors II△		79	28		S		S
	Subelement load vector△		107	24		S		S
	General descriptors III△		131	69		S		S
	General descriptors IV▲		200	144		S	G	I, S
	Important constants and pointers II△		344	6	G	I	I	I
2	Subelement stiffness matrix	IIS	351	IDS*IDS		S		
	J1W vector□	J1	Variable	IT	G	I	I	I
	J2W vector□	J2	Variable	IT	G	I	I	I
	J3W vector□	J3	Variable	IT	G	I	I	I
	J4W vector□	J4	Variable	IT	G	I	I	I
	J5W vector□	J5	Variable	IT	G	I	I	I
	J6W vector□	J6	Variable	IT or 0	G	I	I	I
	J7W vector□	J7	Variable	IT or 0	G	I	I	I
	J8W vector□	J8	Variable	IT or 0	G	I	I	I
	J9W vector□	J9	Variable	IT or 0	G	I	I	I
	J10W vector□	J10	Variable	IT or 0	G	I	I	I
	IBB vector■	IBB	Variable	IND	G	I	I	I
	IBO vector■	IBO	Variable	IND	G	I	I	I
	Material constants vector	IID	Variable	k_1° *IMAT	G	I	I	I
	Thermal expansion constants vector	IIA	Variable	$k_2^{\bullet}$ *IMAT	G	I	I	I
	Pressures vector	IPR	Variable	IPRS	G	I	I	I
	Thicknesses vector	ITE	Variable	NTIC	G	I	I	I
	Uniform temperature increases vector	IDT	Variable	ISDT	G	I	I	I
	Temperature gradients (*y*-direction) vector	IDY	Variable	ISDY	G	I	I	I
	Temperature gradients (*z*-direction) vector	IDZ	Variable	ISDZ	G	I	I	I
	Cross-sectional areas vector	ICAR	Variable	IARE	G	I	I	I
	Torsion constants vector	ICIX	Variable	IMMX	G	I	I	I
	Moments of inertia (about *y*-axis) vector	ICIY	Variable	IMMY	G	I	I	I
	Moments of inertia (about *z*-axis) vector	ICIZ	Variable	IMMZ	G	I	I	I
	Angle types vector	ICFI	Variable	IMFI	G	I	I	I
	X-coordinates of nodes vector	IXX	Variable	IN	G	I	I	I
	Y-coordinates of nodes vector	IYY	Variable	IN	G	I	I	I
	Z-coordinates of nodes vector	IZZ	Variable	IN or 0	G	I	I	I
	C-constants vector■	IIC	Variable	IND	G	I	I	I
3	Deflections (or loads) vector	IDEF	Variable	IND1	G	M	M	I
4	Diagonal-element counts vector	IU	Variable	ISUM + 1	G	I	I	
5	Reduced stiffness matrix vector	IST	Variable	IORD1		G	M	
6	MAX vector	MAX	Variable	ISUM			S	

Legend

** variable address blocks with unique pointers are listed with increasing COMMON addresses

† pointer is a word whose content is the COMMON address of the word immediately before the block (see Table III-2 for COMMON addresses of pointers)

‡ see Table III-2 for the meaning of symbols and their COMMON addresses

§ G generate for next link
S generate for immediate use
I inherit from previous link and preserve
M inherit from previous link and modify

△ see Table VI-3 (Vol. I) and Table III-2 for details

▲ see Tables III-4 and III-5 for details for Link 4

□ see Table IV-3 (Vol. I)

■ see Table VI-2 (Vol. I)

○ $k_1 = 2, 9,$ or 21, depending upon whether ITYPE is 0, 1, or 2, respectively

● $k_2 = 1, 2,$ or 3, depending upon whether ITYPE is 0, 1, or 2, respectively

Table III-1 (contd)**

Block groups	Block name	Pointer of the block†	Common address of first word of block	Block length‡	Link 1 operation§	Link 2 operation§	Link 3 operation§	Link 4 operation§	Legend
7	Residual forces vector	IST	Variable	IND			G	I	** variable address blocks with unique pointers are listed with increasing COMMON addresses
8	DUMMY vector		09000	5000	S				
	IDUM vector		09000	5000	S				† pointer is a word whose content is the COMMON address of the word immediately before the block (see Table III-2 for COMMON addresses of pointers)
	ABIN matrix		09000	8100	S				
	ISIR vector		17100	540	S				
	IMAX vector		17640	540	S				
	ISIRT vector		17640	540	S				‡ see Table III-2 for the meaning of symbols and their COMMON addresses
	IMIN vector		18180	540	S				
9	BB matrix		09000	5994			S		§ G generate for next link
10	FF vector♦							S	S generate for immediate use I inherit from previous link and preserve M inherit from previous link and modify ♦ see Tables III-4 and III-5

Table III-2. Alphabetical listing of the constituents of COMMON block group 1[a]

Symbol	Location in COMMON	Brief description
AA	1–···	Name of whole COMMON block for floating-point references
ACEL	39	Body force per unit volume
AL1	83	Thermal expansion coefficient of an element in first material axis direction
AL2	84	Thermal expansion coefficient of an element in second material axis direction
AL3	85	Thermal expansion coefficient of an element in third material axis direction
CONS	45	Constant for element load vector
DG	82	Temperature gradient for an element in direction y (or z)
DGY	332	Temperature gradient along local y-axis for an element
DGZ	331	Temperature gradient along local z-axis for an element
DT	81	Value of temperature change for an element
D21	86–106	Material constants for an element
G1	47	First direction cosine of acceleration vector
G2	48	Second direction cosine of acceleration vector
G3	49	Third direction cosine of acceleration vector
IA	1–···	Name of whole COMMON block for fixed-point references
IARE	16	Number of cross-sectional area types
IBB	59	Pointer for IBB array
IBN	2	Total number of dbc input units
IBO	60	Pointer for IBO array
IBUN	327	Indicator for boundary conditions input
ICAR	66	Pointer for cross-sectional areas array
ICFI	70	Pointer for angles array
ICIX	67	Pointer for torsional constants array
ICIY	68	Pointer for y-moments of inertia array
ICIZ	69	Pointer for z-moments of inertia array
ICOR	328	Indicator for coordinates input
IDEF	75	Pointer for unknown deflections array (initially loads array)
IDEG	8	Degrees of freedom at a node
IDS	36	Order of the subelement stiffness matrix
IDT	63	Pointer for temperature changes array
IDY	64	Pointer for temperature gradients array (y-direction)
IDZ	334	Pointer for temperature gradients array (z-direction)
IELT	28	Element type number
IERR	79	Error indicator
IGEM	78	Indicator for structures inscribed in ($Z = 0$)-plane
IH	10	Maximum number of vertices
IIA	62	Pointer for thermal expansion coefficient array
IIC	74	Pointer for dbc unit constants array
IID	61	Pointer for material constants array
IIS	77	Pointer for subelement stiffness matrix (IIS = 350)
IMAT	7	Number of material types
IMES	326	Indicator for mesh topology input
IMET	31	Material type number
IMFI	15	Number of angle types
IMMX	12	Number of torsion constants types
IMMY	13	Number of y-moment of inertia types
IMMZ	14	Number of z-moment of inertia types
IMS	34	Number of vertices of current element
IN	1	Total number of nodal points
IND	33	IND = IDEG * IN
INP	42	Indicator for output level
INX	9	Number of last link to be executed
IORD	37	Number of words allocated for the reduced stiffness matrix
IOD1	38	IORD1 = IORD + 1
IP	4	Total number of nonzero concentrated load components
IPBG	43	Integer constant for element load vector
IPEN	44	Integer constant for element load vector
IPIR	329	Indicator for local coordinate axes selection
IPR	333	Pointer for pressure array
IPRS	5	Number of pressure types
ISDT	348	Number of temperature change types
ISDY	347	Number of temperature gradients along local y-axis
ISDZ	346	Number of temperature gradients along local z-axis
ISHUF	35	Relabeling indicator
IST	76	Pointer for reduced stiffness matrix of the whole structure
ISTR	27	Indicator for plane-strain case
ISUM	32	Number of equations in the reduced system
IT	3	Total number of elements

[a]See Table III-1 for sequence and descriptions of COMMON blocks. Table III-2 is a reordering of Table VI-3 (Vol. I), in both of which the "General Descriptors IV" section of the block is partly excluded.

Table III-2 (contd)

Symbol	Location in COMMON	Brief description
ITAP	41	Chain program tape number
ITAS	335	Scratch tape number
ITE	65	Pointer for thicknesses array
ITEM	29	Temperature change type number
ITIC	30	Thickness type number
ITYPE	6	Indicator for material type
IU	46	Pointer for diagonal-element count vector
IXX	71	Pointer for X-coordinates array
IYY	72	Pointer for Y-coordinates array
IZZ	73	Pointer for Z-coordinates array
I8	11	Maximum number of words to describe an element
JARE	340	Type number of cross-sectional area
JMFI	15	Number of angle types
JMMX	339	Type number of the torsional constant about local x-axis
JMMY	338	Type number of the sectional moment of inertia about local y-axis
JMMZ	337	Type number of the sectional moment of inertia about local z-axis
JPRS	343	Type number of pressure
JSDY	342	Type number of temperature gradient along local y-axis
JSDZ	341	Type number of temperature gradient along local z-axis
J1	50	Pointer for J1W array
J2	51	Pointer for J2W array
J3	52	Pointer for J3W array

Symbol	Location in COMMON	Brief description
J4	53	Pointer for J4W array
J5	54	Pointer for J5W array
J6	55	Pointer for J6W array
J7	56	Pointer for J7W array
J8	57	Pointer for J8W array
J9	345	Pointer for J9W array
J10	344	Pointer for J10W array
M	25	Label of current element
N_i	17–24	Labels of vertices of an element
NTIC	349	Number of thickness types
P	107–130	Load vector of a subelement
PRES	330	Pressure value for an element
S	351–···	Subelement stiffness matrix
TE	80	Value of thickness for an element
UV	131–154	Deflections due to temperature changes for an element
X	155–162	Overall X-coordinates for vertices of an element
XD	179–185	X-coordinates of vertices, other than the first vertex, of an element relative to the first vertex
Y	163–170	Overall Y-coordinates of vertices of an element
YD	186–192	Y-coordinates of vertices, other than the first vertex, of an element relative to the first vertex
Z	171–178	Overall Z-coordinates of vertices of an element
ZD	193–199	Z-coordinates of vertices, other than the first vertex, of an element relative to the first vertex
ZGEM	40	Floating-point equivalent of IGEM

Table III-3. Meanings of the entries of important vectors

Vector in COMMON	Meaning of *r*th entry of the vector (all divisions are in integer arithmetic sense)
AA vector	The *r*th component of the total COMMON vector in floating-point mode
D21 vector	The *r*th component of a row-listed upper 6×6 material matrix (see Fig. III-2b, Vol. I), if it exists
IA vector	The *r*th component of the total COMMON vector in fixed-point mode
IBB-pointer-related vector	IBB value of *J*th degree of freedom direction at *i*th node (user's label); $i = 1 + (r - 1)$/IDEG, $J = r - (i - 1)$* IDEG (see Table VI-2, Vol. I)
IBO-pointer-related vector	IBO value of *J*th degree of freedom direction at *i*th node (user's label); $i = 1 + (r - 1)$/IDEG, $J = r - (i - 1)$* IDEG (see Table VI-2, Vol. I)
ICAR-pointer-related vector	Value of *r*th-type cross-sectional area, if it exists
ICFI-pointer-related vector	Value of *r*th-type angle defining principal axes of cross section, if it exists
ICIX-pointer-related vector	Value of *r*th-type torsional constant, if it exists
ICIY-pointer-related vector	Value of *r*th-type *y*-moment of inertia, if it exists
ICIZ-pointer-related vector	Value of *r*th-type *z*-moment of inertia, if it exists
IDEF-pointer-related vectors	(1) Value of prescribed concentrated load in *J*th degree of freedom direction at node *i* (user's label); $i = 1 + (r - 1)$/IDEG, $J = r - (i - 1)$* IDEG (2) Value of *r*th component of reduced load vector (the right-hand-side vector in Fig. II-1, Vol. I) (3) Value of *r*th component of reduced deflection vector ($\{d\}$ vector in Fig. II-1, Vol. I) (4) Value of deflection at the *J*th degree of freedom direction at node *i* (user's label); $i = 1 + (r - 1)$/IDEG, $J = r - (i - 1)$* IDEG For (2) and (3) the node *i* (user's label) and direction *J* associated with the *r*th entry may be obtained as follows: Let r'' be the entry number of the word, in IBB-pointer-related vector, where the absolute value is *r* and r''th entry of IBO-pointer-related vector is -1. Then $i = 1 + (r'' - 1)$/IDEG, and $J = r'' - (i - 1)$* IDEG
IDT-pointer-related vector	Value of *r*th-type temperature increase, if it exists
IDY-pointer-related vector	Value of *r*th-type temperature gradient in *y*-direction, if it exists
IDZ-pointer-related vector	Value of *r*th-type temperature gradient in *z*-direction, if it exists
IIA-pointer-related vector	Value of thermal expansion coefficient in the *J*th material axes direction in element *i*; $i = 1 + (r - 1)/k_2$, $J = r - k_2{}^* (i - 1)$, where k_2 is 1, 2, or 3, depending upon whether ITYPE is 0, 1, or 2, respectively
IIC-pointer-related vector	Value C of *J*th degree of freedom direction at *i*th node (user's label); $i = 1 + (r - 1)$/IDEG, $J = r - (i - 1)$* IDEG (see Table VI-2, Vol. I)
IID-pointer-related vector	Value of *J*th material constant of material type *i*; $i = 1 + (r - 1)/k_1$, $J = r - k_1{}^* (i - 1)$, where k_1 is 2, 9, or 21, depending upon whether ITYPE is 0, 1, or 2, respectively
IIS-pointer-related vector	Element k_{mn} of the free-free subelement stiffness matrix, $m = 1 + (r - 1)$/IDS′, $n = r -$ IDS′* $(m - 1)$; *m* corresponds to m'th degree of freedom direction ($m' = 1 + (m - 1)$/IMS′) at vertex m'' ($m'' = m -$ IMS′* $(m - 1)$; *n* corresponds to n'th degree of freedom direction ($n' = 1 + (n - 1)$/IMS′) at vertex n'' ($n'' = n -$ IDS′* $(n' - 1)$; IMS′ is the number of vertices of subelement, and IDS′ = IMS′* IDEG
IPR-pointer-related vector	Value of *r*th-type pressure, if it exists
IST-pointer-related vector	(1) Element K_{mn} of the stiffness matrix of the supported structure. To find *m*th direction, enter IBB-pointer-related vector with the entry number r' of the word, in IU-pointer-related vector, which is closest to, but not greater than *r*. Let r'' be the entry number of the word, in IBB-pointer-related vector, whose absolute value is r' and the r''th entry in IBO-pointer-related vector is -1; *m*th direction corresponds to m''th degree of freedom direction at node m' (user's label); $m' = 1 + (r'' - 1)$/IDEG, $m'' = r'' - (m' - 1)$* IDEG. To find *n*th direction, determine s' by adding to r' the difference between *r* and the r'th entry of IU-pointer-related vector. Let s'' be the entry number of the word in IBB-pointer-related vector, whose absolute value is s' and the s''th entry in IBO-pointer-related vector is -1; *n*th direction corresponds to n''th degree of freedom direction at node n' (user's label); $n' = 1 + (s'' - 1)$/IDEG, $n'' = s'' - (n' - 1)$* IDEG (2) Value of residual force acting at node *i* in direction *J*, where $i = 1 + (r - 1)$/IDEG, $J = r - (i - 1)$* IDEG

Table III-3 (contd)

Vector in COMMON	Meaning of *r*th entry of the vector (all divisions are in integer arithmetic sense)
ITE-pointer-related vector	Value of *r*th-type thickness, if it exists
IU-pointer-related vector	Entry number in IST-pointer-related vector of *r*th diagonal element of the reduced stiffness matrix
IXX-pointer-related vector	*X*-coordinate of node *r* (user's label)
IYY-pointer-related vector	*Y*-coordinate of node *r* (user's label)
IZZ-pointer-related vector	*Z*-coordinate of node *r* (user's label), if it exists
J1-pointer-related vector	J1W value of *r*th element (see Table IV-3, Vol. I)
J2-pointer-related vector	J2W value of *r*th element (see Table IV-3, Vol. I)
J3-pointer-related vector	J3W value of *r*th element (see Table IV-3, Vol. I)
J4-pointer-related vector	J4W value of *r*th element (see Table IV-3, Vol. I)
J5-pointer-related vector	J5W value of *r*th element (see Table IV-3, Vol. I)
J6-pointer-related vector	J6W value of *r*th element (see Table IV-3, Vol. I), if it exists
J7-pointer-related vector	J7W value of *r*th element (see Table IV-3, Vol. I), if it exists
J8-pointer-related vector	J8W value of *r*th element (see Table IV-3, Vol. I), if it exists
J9-pointer-related vector	J9W value of *r*th element (see Table IV-3, Vol. I), if it exists
J10-pointer-related vector	J10W value of *r*th element (see Table IV-3, Vol. I), if it exists
N vector	The label (user's) of the *r*th vertex of an element
MAX-pointer-related vector	Number of nonzero entries above *r*th diagonal element of the decomposed reduced stiffness matrix (see Fig. II-1, Vol. I)
P vector	Element load acting in direction J of ith vertex of a subelement; $J = 1 + (r - 1)/\mathrm{IMS}'$, $i = r - (J - 1)^* \mathrm{IMS}'$ (IMS′ = number of vertices of the subelement)
S matrix	See IIS-pointer-related vector
UV vector	Deflection in direction J of ith vertex of a subelement subjected to temperature change in local coordinates; $J = 1 + (r - 1)/\mathrm{IMS}'$, $i = r - (J - 1)^* \mathrm{IMS}'$ (IMS′ = number of vertices of subelement)
X vector	*X*-coordinate of *r*th vertex of an element
XD vector	*X*-coordinate, relative to the first vertex, of $(r + 1)$st vertex of an element
Y vector	*Y*-coordinate of *r*th vertex of an element
YD vector	*Y*-coordinate, relative to first vertex, of $(r + 1)$st vertex of an element
Z vector	*Z*-coordinate of the *r*th vertex of an element
ZD vector	*Z*-coordinate, relative to the first vertex, of $(r + 1)$st vertex of an element

Table III-4. Alphabetical list of additional COMMON variables for Link 4[a]

Symbol	COMMON location	Symbol	COMMON location	Symbol	COMMON location	Symbol	COMMON location
A	14786–15415	ICAS	212	IWG	14660–14749	NU	292–294
ANGLE	211	ICLA	206	JM1	293	QF	253–258
ARE	205	ICLAS	274–277	JP1	292	QN	247–252
AST	203	ICN	201	JS1	294	RED	265–270
B	15416–15479	ICOL	295	LM	202	RES	259–264
BAS	271–273	ICON	210	MAC	14400–14659	SIR	223–225
BIR	220–222	IDR	297	MSET	15596–15695	SR	235–240
BST	217	IE	213	MB	215	W	15696–15704
C	15480–15495	IM	208	NB	214	XF	244–246
DD	14750–14785	IMEL	207	NBAN	278–287	XN	241–243
DIN	226–234	INBON	204	NEL	14000–14399	XII	226–228
ETA	229–231	IONE	200	NES	295–297	ZTA	232–234
FF	14000–15704	IRIG	296	NSET	15496–15595		
IC	209	IROT	216				

[a]See Table III-5 for meanings of variables.

Table III-5. List of additional COMMON variables for Link 4[a]

Location in COMMON	Symbol	Brief description
1–199		This portion of COMMON is as in Table VI-3 of Vol. I
200	IONE	Total number of one-dimensional elements in the structure
201	ICN	Label of current mesh point (ICN varies from 1 to IN)
202	LM	Total number of non-one-dimensional elements meeting at mesh point ICN
203	AST	Indicator containing * or BCD blank, depending upon whether mesh point ICN is on boundary or not, respectively
204	INBON	Indicator containing 1 or 0, depending upon whether mesh point ICN is on boundary or not, respectively
205	ARE	Average boundary surface area for mesh point ICN, if it is on boundary
206	ICLA	Total number of class[b] types for elements of material type group IM at mesh point ICN
207	IMEL	Total number of material types at mesh point ICN
208	IM	Current material type group number (IM varies from 1 to IMEL)
209	IC	Current class[b] type group number (IC varies from 1 to ICLA)
210	ICON	Sequence number of current strain-deflection equation at mesh point ICN for material group IM and for class[b] group IC
211	ANGLE	Angle between XII local axis and the 1–2 line of the lowest labeled shell element attached to mesh point ICN
212	ICAS	Class[b] type number of ICth class[b] group of IMth material group at mesh point ICN
213	IE	Number of mesh elements (of class[b] group IC of material group IM) plus 1 at mesh point ICN
214	NB	Total number of mesh points in node set at mesh point ICN
215	MB	Number of boundary points attached to mesh point ICN
216	IROT	Indicator containing 0 or 1, depending upon whether local axes at mesh point ICN are parallel to overall axes or not, respectively
217	BST	Indicator containing BCD blank or **, depending upon whether local axes at mesh point ICN are parallel to overall or not, respectively
220–222	BIR	Direction cosines of outer unit normal vector at mesh point ICN, if it is on boundary

[a]This table is not applicable to subroutine DIMI of Link 4.
[b]Class types are those of Table VI-6, Vol. I.

Table III-5 (contd)

Location in COMMON	Symbol	Brief description
223–225	SIR	Vector heading towards structure at mesh point ICN, if it is on boundary
226–234	DIN	Direction cosines of local axes in overall coordinate system at mesh point ICN (the columns of DIN are named as XII, ETA, and ZTA)
235–240	SR	Independent components of stress tensor for ICth class[b] group of IMth material group at mesh point ICN
241–243	XN	Overall coordinates of mesh point ICN
244–246	XF	Overall coordinates of IIth vertex of ILth element of ICth class[b] group of IMth material group at mesh point ICN
247–252	QN	Deflection components in overall coordinates of mesh point ICN
253–258	QF	Deflection components in overall coordinates of mesh point whose overall coordinates are in XF
259–264	RES	Residual forces[c] in overall coordinates at mesh point ICN, if on boundary
265–270	RED	Relative deflections (in overall coordinates) of mesh point related with XF vector with respect to mesh point ICN
271–273	BAS	Direction cosines of 1–2 line of the lowest labeled element of class[b] group IC of material group IM at mesh point ICN
274–277	ICLAS	Number of class[b] groups in each material group (maximum 4) of mesh point ICN
278–287	NBAN	List of labels of boundary mesh points attached to mesh point ICN
292–294	NU	Vector containing the sequence numbers of the vertices after (JP1), before (JM1), and above (JS1) mesh point ICN in the Ith mesh element of the node set (with Table III-5, vol. I)
292	JP1	See NU (1)
293	JM1	See NU (2)
294	JS1	See NU (3)
295–297	NES	Vector containing number of independent strain components (ICOL), number of right-hand sides (IRIG) and indicator of right-hand-side arrangement (IDR) (IDR = 0 means lineal strains first, IDR = 1 means rotational strains first) for current ICN/IM/IC
295	ICOL	See NES (1)
296	IRIG	See NES (2)
297	IDR	See NES (3)
329–349		This portion of COMMON is as shown in Table VI-3, Vol. I. See also Fig. III-3, Link 4
349–13999		See Fig. III-1, Link 4
14000–15704	FF	Vector containing information for stress conputation at mesh point ICN
14000–14399	NEL	Element set information of mesh point ICN (see Table VI-7, Vol. I)
14400–14659	MAC	Table for classes and material of element set at mesh point ICN (see Table VI-7, Vol. I)
14660–14749	IWG	Vector of weights of strain-deflection equations for current ICN/IM/IC
14750–14785	DD	Material matrix for current ICN/IM/IC
14786–15415	A	Augmented matrix of strain-deflection equations for current ICN/IM/IC
15416–15479	B	Coefficient matrix (or its inverse) of the least-squares equations for strain for current ICN/IM/IC
15480–15495	C	Right-hand-side vector(s) of the least-squares equations for strains for current ICN/IM/IC
15496–15595	NSET	List of labels of mesh points on the boundary and attached to mesh point ICN
15596–15695	MSET	Auxiliary array for NSET
15696–15704	W	Direction cosines of new material axes in the old for current ICN/IM/IC

[c]Residual forces are those listed by Output Item 20 (see Sect. VI-D and VI-E, Vol. I).

Appendix

Corrigenda for Volume I

Corrigenda for Volume I

Page	Location	Error	Correction
12	Table III-5, Element 9	Counterclockwise sequence for the first three . . .	Clockwise sequence for the first three . . .
13	Fig. III-2(c)	$b = \dfrac{G(2G)}{E(3G - E)}$	$b = \dfrac{G(E - 2G)}{E(3G - E)}$
14	Col. 2, line 19	. . . , the other may not appear more than twice.	. . . , the other may not appear in such units.
18	Col. 2, line 8	For one-dimensional structural elements, pressure . . .	For one-dimensional structural elements (excluding element type 3), pressure . . .
19	Col. 2, line 27	and the overall Y axis; ϕ is always less than 90 deg.	and the overall Y axis; ϕ is never greater than 90 deg.
19	Col. 2, line 29	. . . and the overall X axis is less than 90 deg.	. . . and the overall X axis is not greater than 90 deg.
23	Table IV-2, column heads	. . . . . . Range Format . . .	. . . . . . Format Range . . .
28	Table V-2		
29	Table V-3	Numbers in second column (Length in 36-bit words) may be slightly different because of maintenance corrections	
30	Table V-4		
31	Table V-5		
44	Table VI-2, third entry	i p k b Deflection component . . .	i n k b Deflection component . . .
45	Table VI-3	32 ISUM Number of equations	32 ISUM Number of equations in the reduced system
45	Table VI-3	35 (not used)	35 ISHUF Relabeling indicator
45	Table VI-3	36 IDS Order of the element stiffness matrix	36 IDS Order of the subelement stiffness matrix
45	Table VI-3	77 IIS Pointer for element stiffness matrix	77 IIS Pointer for subelement stiffness matrix
46	Table VI-3	107 P Loading vector for an element	107 P Loading vector for a subelement
53	Col. 2, line 1	sweeps performed, . . .	interchanges performed, . . .
56	Table VII-1, No. 3	. . . IS OUTSIDE THE RANGE. THE TYPE NUMBER(S) IS ASSUMED LARGEST POSSIBLE	. . . IS OUTSIDE THE RANGE. THE TYPE NUMBER IS ASSUMED LARGEST POSSIBLE.
62	Fig. VIII-1, vertical dimensions	0.17 0.33 0.50	0.13 0.37 0.50
63	Fig. VIII-2, legend	$G = 11. \times 10^6$ psi	$G = 4. \times 10^6$ psi
65	Table VIII-2, item 5, line 1	AT 0. SEC. OF RELABELLING —0 SWEEPS PERFORMED.	AT 0. SEC. OF RELABELING —0 INTERCHANGES DONE.
65	Table VIII-2, item 5, line 2	AT 0.28 SEC. OF RELABELLING 44 SWEEPS PERFORMED.	AT 0.28 SEC. OF RELABELING 44 INTERCHANGES DONE.
72	Table VIII-5, item 22	The numbers in this table are slightly modified because of correction on card E4ABQ142 on Jan. 22, 1969	
73	Col. 2, line 26	ELAS1424, ELAS1426, ELAS1491 in subroutine DARN . . .	ELAS1423, ELAS1426, ELAS1491, E1SR61A9, E1SR61B0 in subroutine DARN. . . .

References

1. Collatz, L., *Numerische Behandlung von Differential gleichungen*. Springer-Verlag, OHG, Berlin, 1951.

2. Ritz, W., "Uber eine neue Methode zur Losung gewisser Variations—probleme der mathematischen Physik," *J. reine u. angew. Math.*, Vol. 135, pp. 1–61, 1909.

3. Synge, J. L., *The Hypercircle in Mathematical Physics*. Cambridge University Press, Cambridge, England, 1957.

4. Utku, S.,"Best Fit Stress Computation in Displacement Methods," paper presented at the ASCE Engineering Mechanics Division Specialty Conference, North Carolina University, Raleigh, Nov. 1967.

5. Akyuz, F. A., and Utku, S., "An Automatic Node Relabelling Scheme for Bandwidth Minimization of Stiffness Matrices," *AIAA Journal*, Vol. 6, No. 4, pp. 728–730, Apr. 1968.

6. Utku, S., *Computation of Stresses in Triangular Finite Elements*, Technical Report 32-948. Jet Propulsion Laboratory, Pasadena, Calif., June 1966.

Bibliography

Utku, S., "Stiffness Matrices for Thin Triangular Elements of Nonzero Gaussian Curvature," *AIAA Journal*, Vol. 5, No. 9, pp. 1659–1667, Sept. 1967.

Utku, S., "Explicit Expressions for Triangular Torus Element Stiffness Matrix," *AIAA Journal*, Vol. 6, No. 6, pp. 1174–1176, June 1968.

Utku, S., "A Conical Element Stiffness Matrix Associated With Polyhedral Deflection Fields," in *M. Inan Memorial Volume*. ITU Publications, Istanbul, 1969 (in press).

Utku, S., and Melosh, R. J., "Behavior of Triangular Shell Element Stiffness Matrices Associated With Polyhedral Deflection Distributions," *AIAA Journal*, Vol. 6, No. 2, pp. 374–376, Feb. 1968. Also available as Technical Report 32-1217, Jet Propulsion Laboratory, Pasadena, Calif., Jan. 1968.