CENTER FOR COMPUTER AIDED DESIGN

College of Engineering
The University of Iowa
Iowa City, Iowa 52242

N87-27220

Unclas

G3/39  C082561
Technical Report 84-17

DESIGN SENSITIVITY ANALYSIS WITH APPLICON IFAD USING THE ADJOINT VARIABLE METHOD

Marjorie C. Frederick and Kyung K. Choi

Center for Computer Aided Design
College of Engineering
The University of Iowa

December 1984

Research was partially supported by
NASA Grant NAG-1-215
and Applicon
ABSTRACT

A numerical method is presented to implement structural design sensitivity analysis using the versatility and convenience of existing finite element structural analysis program and the theoretical foundation in structural design sensitivity analysis. Conventional design variables, such as thickness and cross-sectional areas, are considered. Structural performance functionals considered include compliance, displacement, and stress. It is shown that calculations can be carried out outside existing finite element codes, using postprocessing data only. That is, design sensitivity analysis software does not have to be imbedded in an existing finite element code.

The finite element structural analysis program used in the implementation presented is IFAD. Feasibility of the method is shown through analysis of several problems, including built-up structures. Accurate design sensitivity results are obtained without the uncertainty of numerical accuracy associated with selection of a finite difference perturbation.
TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>LIST OF TABLES</th>
<th>iv</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF FIGURES</td>
<td>v</td>
</tr>
<tr>
<td>LIST OF SYMBOLS</td>
<td>vi</td>
</tr>
</tbody>
</table>

CHAPTER

I. INTRODUCTION ........................................ 1
   1.1 Purpose ........................................ 1
   1.2 Adjoint Variable Method ......................... 4
   1.3 Adjoint Variable Method Results ............... 4

II. DESIGN SENSITIVITY ANALYSIS METHOD ............... 6
   2.1 Calculation Procedure for Structural Components ........................................ 6
       2.1.1 Membranes .................................. 8
       2.1.2 Bending of Beams ......................... 15
       2.1.3 Bending of Plates ....................... 20
   2.2 Calculation Procedure for a Built-Up Structure ........................................ 26

III. NUMERICAL EXAMPLES .................................. 36
    3.1 Membranes ..................................... 36
    3.2 Bending of Beams ................................ 41
    3.3 Bending of Plates ................................ 45
    3.4 Built-up Structure ............................. 50

IV. CONCLUSIONS ........................................ 55

REFERENCES ............................................. 56

APPENDIX
   DESIGN SENSITIVITY IFAD PROGRAM ................... 57

iii
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Membrane Design Sensitivity Check for Compliance</td>
<td>38</td>
</tr>
<tr>
<td>2. Design Sensitivity Check for Displacement</td>
<td>39</td>
</tr>
<tr>
<td>3. Design Sensitivity Check for Stress</td>
<td>40</td>
</tr>
<tr>
<td>4. Beam Design Sensitivity Check for Compliance</td>
<td>42</td>
</tr>
<tr>
<td>5. Beam Design Sensitivity Check for Displacement</td>
<td>44</td>
</tr>
<tr>
<td>6. Beam Design Sensitivity Check for Stress</td>
<td>44</td>
</tr>
<tr>
<td>7. Plate Design Sensitivity Check for Compliance</td>
<td>47</td>
</tr>
<tr>
<td>8. Plate Design Sensitivity Check for Displacement</td>
<td>47</td>
</tr>
<tr>
<td>9. Plate Design Sensitivity Check for von Mises' Stress</td>
<td>49</td>
</tr>
<tr>
<td>10. Built-up Structure Design Sensitivity Check for Compliance</td>
<td>52</td>
</tr>
<tr>
<td>(δh = 0.01h, δb = 0.01b, and δt = 0.01t)</td>
<td></td>
</tr>
<tr>
<td>11. Built-up Structure Design Sensitivity Check for Displacement</td>
<td>52</td>
</tr>
<tr>
<td>(δh = 0.01h, δb = 0.01b, and δt = 0.01t)</td>
<td></td>
</tr>
<tr>
<td>12. Built-up Structure Design Sensitivity Check for von Mises' Stress</td>
<td>54</td>
</tr>
<tr>
<td>(δt = 0.001t, δb = 0.001b, and δh = 0.001h)</td>
<td></td>
</tr>
</tbody>
</table>
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Flow Chart of Feasibility Check Procedure</td>
<td>3</td>
</tr>
<tr>
<td>2. Flow Chart of Design Sensitivity Calculation Procedure</td>
<td>7</td>
</tr>
<tr>
<td>3. Clamped Elastic Solid of Variable Thickness h(x)</td>
<td>8</td>
</tr>
<tr>
<td>4. Cantilever Beam with Variable Width and Height</td>
<td>15</td>
</tr>
<tr>
<td>5. Clamped Plate of Variable Thickness t(x)</td>
<td>20</td>
</tr>
<tr>
<td>6. Roof Structure Example</td>
<td>28</td>
</tr>
<tr>
<td>7. Plane Elastic Solid Finite Element Model</td>
<td>37</td>
</tr>
<tr>
<td>8. Cantilever Beam Finite Element Model</td>
<td>42</td>
</tr>
<tr>
<td>9. Bending Plate Finite Element Model</td>
<td>46</td>
</tr>
<tr>
<td>10. Built-up Structure Finite Element Model</td>
<td>51</td>
</tr>
</tbody>
</table>
# LIST OF SYMBOLS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_u$</td>
<td>Bilinear form which is dependent on $u$</td>
</tr>
<tr>
<td>$\ell_u$</td>
<td>Load linear form which is dependent on $u$</td>
</tr>
<tr>
<td>$u$</td>
<td>Design vector</td>
</tr>
<tr>
<td>$\ell$</td>
<td>Design sensitivity vector; Gauss quadrature counter; Local coordinate system of a beam</td>
</tr>
<tr>
<td>$\psi, \psi_i$</td>
<td>Various constraint functionals</td>
</tr>
<tr>
<td>$\psi', \psi'_i$</td>
<td>Design sensitivities of various constraint functionals</td>
</tr>
<tr>
<td>$\delta u$</td>
<td>Perturbed design vector</td>
</tr>
<tr>
<td>$h$</td>
<td>Thickness of membrane; height of beam</td>
</tr>
<tr>
<td>$t$</td>
<td>Thickness of bending plate</td>
</tr>
<tr>
<td>$b$</td>
<td>Width of beam</td>
</tr>
<tr>
<td>$E$</td>
<td>Young's modulus</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Poisson's ratio</td>
</tr>
<tr>
<td>$\hat{D}(u)$</td>
<td>Flexural rigidity of plate</td>
</tr>
<tr>
<td>$J$</td>
<td>Torsional rigidity of beam; determinant of the Jacobian</td>
</tr>
<tr>
<td>$W_i, W_{\ell}$</td>
<td>Gauss quadrature weight factor</td>
</tr>
<tr>
<td>$G$</td>
<td>Shear modulus</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Kroneker Delta</td>
</tr>
<tr>
<td>$\varepsilon, \varepsilon^{ij}$</td>
<td>Strain</td>
</tr>
<tr>
<td>$\sigma, \sigma^{ij}$</td>
<td>Stress</td>
</tr>
<tr>
<td>$\lambda, \lambda^{(i)}$</td>
<td>Adjoint variable vector</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>Domain of the system considered</td>
</tr>
</tbody>
</table>
\[ \Gamma \] Boundary of the system considered
\[ \sum \] Summation
\[ z, z^i, z_x^i \] Displacement
\[ F, F^i, F_x^i \] Applied body force
\[ T, T^i \] Applied traction
\[ \tilde{z}, \tilde{z}^i \] Virtual displacement
\[ \delta h \] Perturbation of \( h \)
\[ \delta h \] Perturbation of \( b \)
\[ \delta t \] Perturbation of \( t \)
\[ m_p \] Characteristic function
\[ \partial \] Partial derivative
\[ [ ] \] Matrix
\[ \bar{d} \] Virtual displacement of an element
\[ z_{ii}, z_{xx}, z_{xx}^i \] Beam curvature due to \( z \)
\[ \bar{z}_{xx}, \bar{z}_{ii} \] Beam curvature due to \( \bar{z} \)
\[ k \] Element counter
\[ \gamma \] Material density
\[ \bar{\lambda}_{xx}, \lambda_{xx}^i \] Beam curvature due to \( \bar{\lambda} \)
\[ L \] Length of a beam element
\[ A \] Area of a triangle
\[ \phi \] General function
CHAPTER I
INTRODUCTION

1.1 Purpose

To date there exists a wide variety of finite element structural analysis programs that are used as reliable tools for structural analysis. They give the designer pertinent information such as stresses, strains and displacements of the mechanical system being modeled. However, if this information reveals that the mechanical system does not meet specified constraint requirements, the designer must make intuitive guesses as to how to improve the design. If the mechanical system is complex, it becomes very difficult to decide what step must be taken to improve the design. There is however, substantial literature on the theory of design sensitivity analysis, which predicts the effect that structural design changes have on the performance of a mechanical system. Use of this technique has been primarily confined to papers in structural optimization literature.

The purpose of this work is to develop and implement structural design sensitivity analysis using the adjoint variable method that takes advantage of the versatility and convenience of an existing finite element structural analysis program and the theoretical foundation in structural design sensitivity analysis that is found in Ref. 1. The finite element program that will be used is IFAD [3]. It is developed
by Applicon Inc. and has been provided to the Center for Computer Aided Design for the use in this study.

In order to check the feasibility of using the design sensitivity analysis technique with IFAD, an approximation of the differential $\psi'$ of a structural performance measure $\psi$ is made using the finite difference method. An appropriate design perturbation $\delta u$ must be selected in order to insure accuracy of the perturbation $\Delta \psi$ of the constraint functional. If $\delta u$ is too small, $\Delta \psi = \psi(u + \delta u) - \psi(u)$ may be inaccurate due to loss of significant digits in the difference. On the other hand, if $\delta u$ is too large, $\Delta \psi$ will be influenced by nonlinearities and the differential approximation will be inaccurate. The feasibility check procedure is outlined in the flow chart of Fig. 1. Details of the calculations of the constraint functionals, the adjoint loads, and the design sensitivity vectors for each constraint functional are described in Chapter II, for different types of finite elements. The design sensitivity $\psi'$ of the constraint functional is the scalar product of the design sensitivity vector $\ell$ and the design variable perturbation vector $\delta u$. If the design variable is constant throughout the finite element model of the mechanical system, this becomes a scalar multiplication. If the design sensitivity is an accurate prediction of the performance of the mechanical system due to a design change, it should be equivalent to the difference of the constraint functionals of the two finite element models, the original and the perturbed model.
Original FE model with adjoint load cases design variable is $u$

FE Analysis

Original structural response and adjoint structural responses

Calculate constraint functional $\psi(u)$ and design sensitivity $\psi' = T^T \delta u$

Perturbed FE model design variable is $u + \delta u$

FE Analysis

Perturbed structural response

Calculate constraint functional $\psi(u+\delta u)$

Calculate

$\Delta \psi = \psi(u+\delta u) - \psi(u)$

% Accuracy $= \psi' \times 100/\Delta \psi$

Figure 1. Flow Chart of Feasibility Check Procedure
1.2 Adjoint Variable Method

A number of methods could be used to implement structural design sensitivity analysis with an existing finite element code, but the most powerful is the adjoint variable method. This method can be implemented outside of an existing finite element code, using only postprocessing data. This is convenient, because the source code for most finite element programs is not readily available. If the code is available, less programming is involved. The same subroutines for the element shape functions used in the finite element model can be used in the design sensitivity analysis, since this method is dependent on element type. Generality is another factor that adds flexibility to the adjoint variable method. The code can be written to include basic design variables, constraints and loading conditions. This enables the designer to choose what design variables to modify to give the best design improvement.

The adjoint variable method can easily be used for complex mechanical systems that have more than one structural component. The details of this procedure are discussed in Section 2.2. Design sensitivity of a built-up structure is formed by combining the design sensitivities of each structural component. The only precaution that is necessary is in making sure that the interaction between the components is taken into account.

1.3 Adjoint Variable Method Results

The design sensitivity vector is the derivative of the constraint functional with respect to the design variables. It has the same number
of components as there are elements in the finite element model. The magnitude of each component reflects how sensitive the element is to a change in design relative to the constraint functional. If the vector component is negative, the corresponding design variable should be decreased to increase $\psi$. Likewise, if the vector component is positive, the design variable should be increased to increase $\psi$. In addition, if the magnitude of the vector component is large, then the corresponding design variable will have a more substantial effect on design improvement.

When a designer uses a finite element structural analysis in design of a mechanical system, it is most likely that a number of program runs are necessary before a substantially improved design is obtained. With the aid of a design sensitivity vector, the designer will know what direction to take to improve the design most efficiently.
CHAPTER II
DESIGN SENSITIVITY ANALYSIS METHOD

2.1 Calculation Procedure for Structural Components

To implement the adjoint variable technique of design sensitivity analysis, the adjoint load for each constraint functional must be calculated. This procedure is developed in Ref. 1 using compliance, displacement, stress and natural frequency as constraint functionals. For the compliance functional, the adjoint equation is the same as the state equation. In this special case the adjoint system does not need to be solved. For the displacement functional the adjoint load is a unit point load acting at the point where the displacement constraint is imposed. To calculate the adjoint response it is necessary only to restart the finite element analysis with unit loads applied at varying points along the structure. For the stress functional the shape function of the structural component used in the finite element analysis must be known. This shape function is used to calculate the adjoint load for a stress constraint of a specific element of the structure. From this point the procedure is similar to the displacement functional, in that a restart of the finite element analysis must be completed using the adjoint loads of elements as other load cases.

The flow chart of Fig. 2 shows the overall process. This procedure is implemented after the structural response of the finite element
Figure 2. Flow Chart of Design Sensitivity Calculation Procedure
model due to the original load has been solved. The original structural response plus the adjoint response for each constraint is then utilized to calculate the design sensitivity vectors.

The following sections give detailed explanations of the calculation procedures and equations necessary for analyzing membranes, bending beams and bending plates.

2.1.1 Membranes

Consider a variable thickness thin elastic clamped solid, as shown in Fig. 3. The design variable is taken as the variable thickness \( u = h(x) \) of the plate.

\[ \int_{\Omega} h(x) \sum_{i,j=1}^{2} \sigma_{ij}^{s}(x) e_{ij}^{f}(x) d\Omega \]

Figure 3. Clamped Elastic Solid of Variable Thickness \( h(x) \)

The energy bilinear form and the load linear form of the plane elasticity problem are given as [1]
where $z = [z^1, z^2]^T$ is the displacement, $F = [F^1, F^2]^T$ is the applied body force, $T = [T^1, T^2]^T$ is the traction, and $\sigma^{ij}(z)$ and $\varepsilon^{ij}(\bar{z})$ are the stress and strain fields associated with the displacement $z$ and the virtual displacement $\bar{z}$ respectively. The state equation is given as [1]

$$a_u(z, \bar{z}) = \mathcal{L}_u(\bar{z})$$

for all kinematically admissible virtual displacement $\bar{z}$.

First consider the functional representing the compliance of the structure as

$$\psi_1 = \int\int_{\Omega} h(x) \left[ \sum_{i=1}^{2} F^{i-1} \right] d\Omega + \int_{\Gamma} \left[ \sum_{i=1}^{2} T^{i-1} \right] d\Gamma$$

The first variation of Eq. (4) is

$$\psi_1' = \int\int_{\Omega} h(x) \left[ \sum_{i=1}^{2} F^{i-1} \right] \delta h d\Omega + \int\int_{\Omega} h(x) \left[ \sum_{i=1}^{2} F^{i-1} \right] d\Omega$$

$$+ \int_{\Gamma} \left[ \sum_{i=1}^{2} T^{i-1} \right] d\Gamma$$

In order to eliminate the dependence on the state variable in Eq. (5), it is necessary to define the adjoint equation as [1]

$$a_u(\lambda, \bar{\lambda}) = \int\int_{\Omega} h(x) \left[ \sum_{i=1}^{2} F^{i-1} \right] d\Omega + \int_{\Gamma} \left[ \sum_{i=1}^{2} T^{i-1} \right] d\Gamma$$

for all kinematically admissible virtual displacement $\bar{\lambda}$. Since Eq. (6) is identical to Eq. (4) if $\lambda = z$ and $\bar{\lambda} = \bar{z}$, the adjoint equation does
not need to be solved. Using the adjoint variable method of design sensitivity analysis gives [1]

\[
\psi_1' = \int_\Omega \left[ \sum_{i=1}^{2} F_{i}^i \lambda_{i} \right] \delta h \, d\Omega + \int_\Omega \left[ \sum_{i=1}^{2} \sigma_{i,j}^{i}(z) \right] \delta h \, d\Omega
\]

\[
= \int_\Omega \left[ \sum_{i=1}^{2} F_{i}^i \lambda_{i} - \sum_{i,j=1}^{2} \sigma_{i,j}^{i}(z) \varepsilon_{i,j}^{i}(z) \right] \delta h \, d\Omega
\]

(7)

since \( z = \lambda \) for the compliance functional.

To numerically integrate Eqs. (4) and (7), a two-point Gauss quadrature formula is used. The equations become

\[
\psi_1 = \sum_{k=1}^{N} \left[ h_k \sum_{\ell=1}^{2} \sum_{i=1}^{2} F_{i}^i \varepsilon_{i,j}^{i} \right] W_{\ell} J + \sum_{i=1}^{2} T_{i}^i \varepsilon_{i,j}^{i} \]

(8)

and

\[
\psi_1' = \sum_{k=1}^{N} \left[ \sum_{\ell=1}^{2} \sum_{i=1}^{2} 2F_{i}^i \varepsilon_{i,j}^{i} \delta_{i,j}^{i} \right] \delta h_k
\]

(9)

respectively, where \( J \) is the Jacobian, \( N \) is the total number of elements, subscript \( \ell \) is the counter for the number of Gauss points, subscript \( k \) is the counter for the element number, \( W \) is the weighting constant for the \( \ell \)th Gauss point, and superscript \( i \) is the direction of the force and the displacement.

Next consider the functional representing the displacement \( z \) at a discrete point \( \hat{x} \) as

\[
\psi_2 \equiv z(\hat{x}) = \int_\Omega \delta(\hat{x} - \hat{x}) z(x) \, d\Omega
\]

(10)
where \( \delta(x) \) is the Dirac measure in the plane, acting at the origin. The first variation of Eq. (10) is

\[
\psi'_2 = \iint_\Omega \delta(x - \hat{x}) z'(x) d\Omega
\]  
(11)

The adjoint equation in this case is [1]

\[
a_u(\lambda, \bar{\lambda}) = \iint_\Omega \delta(x - \hat{x}) \bar{\lambda}(x) d\Omega
\]
(12)

for all kinematically admissible virtual displacement \( \bar{\lambda} \). This equation has a unique solution \( \lambda^{(2)} \), where \( \lambda^{(2)} \) is the plate displacement due to a unit point load acting at a point \( \hat{x} \). Using the adjoint variable method of design sensitivity analysis gives

\[
\psi'_2 = \iint_\Omega [ \sum_{i=1}^{2} F_i^i \lambda^{(2)}_i \sum_{i,j=1}^{2} \sigma_{ij}^i(z) \varepsilon_{ij}^{(2)}(\lambda^{(2)}) ] \delta h d\Omega
\]
(13)

where \( \lambda^{(2)} \) is the solution of Eq. (12).

For this constraint two equations must be solved. The adjoint load of Eq. (12) is just a unit load applied at a discrete point in the finite element model. All that is necessary is a restart of the model so that load cases of applied unit loads at various nodal points can be analyzed. The resulting strains due to the adjoint load are then used in calculating Eq. (13). Note that for each displacement constraint there is a different adjoint load.

With numerical techniques applied as in the compliance constraint case, Eq. (13) becomes
Finally consider the general functional representing a locally averaged stress on an element as

$$\psi_2 = \sum_{k=1}^{N} \left[ \sum_{l=1}^{2} \sum_{i=1}^{2} \int_{\Omega} \left( \frac{d}{d\lambda} \sigma_{ij}^{(l)}(\lambda) \right) \frac{d\Omega}{\partial\lambda} \right] \delta h_k \delta h_k$$

(14)

where $m_p$ is a characteristic function defined on a finite element $\Omega_p$ as

$$m_p = \begin{cases} \frac{1}{\Omega_p} \frac{d\Omega}{\partial\lambda}, & x \in \Omega_p \\ 0, & x \notin \Omega_p \end{cases}$$

(16)

The first variation of Eq. (15) is

$$\psi_3' = \int_{\Omega} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma_{ij}^{(l)}} \sigma_{ij}^{(l)} \right] m_p \, d\Omega$$

(17)

Replacing the variation in state $z'$ by a virtual displacement $\lambda$, the adjoint equation is obtained as [1]

$$a_u(\lambda, \lambda) = \int_{\Omega} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma_{ij}^{(l)}} \sigma_{ij}^{(l)} \right] m_p \, d\Omega$$

(18)

for all kinematically admissible virtual displacements $\lambda$. Eq. (18) has a unique solution for a displacement field $\lambda^{(3)}$. Using the adjoint variable method of design sensitivity analysis gives

$$\psi_3' = \int_{\Omega} \left[ \sum_{i=1}^{2} F^i \lambda^{(3)} - \sum_{i,j=1}^{2} \sigma_{ij}^{(l)}(z) \varepsilon_{ij}^{(l)}(\lambda^{(3)}) \right] \delta h \, d\Omega$$

(19)

where $\lambda^{(3)}$ is the solution of Eq. (18).
With numerical techniques applied as in the displacement constraint case, Eqs. (15), (18), and (19) become

\begin{equation}
\psi_3 = \sum_{k=1}^{2} [g(\sigma_i(z))_m \ W_k] J
\end{equation}

\begin{equation}
\int_{\Omega} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma_{ij}} \sigma_{ij}(\lambda) \right]_m \ W_k \ d\Omega
= \sum_{k=1}^{2} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma_{ij}} \sigma_{ij}(\lambda) \right]_m \ W_k \ J \ \partial \Omega
\end{equation}

and

\begin{equation}
\psi'_3 = \sum_{k=1}^{N} \left[ \sum_{i=1}^{2} \sum_{j=1}^{2} F_{i,k} \lambda_{ij} - \sum_{i,j=1}^{2} \sigma_{ij}(z) \epsilon_{i,j}(\lambda(3)) W_k \right] J \ \delta h_k
\end{equation}

respectively. The numerical calculation of the adjoint load is considerably more difficult in the stress functional case. The shape function of the element must be known so that \( \sigma_{ij}(\lambda) \) can be calculated using the finite element technique [2]

\begin{equation}
\sigma = [E][B]d = [E]\epsilon
\end{equation}

where [E] is the elasticity matrix, [B] is the strain-displacement matrix, which relates to the element shape function, d is the displacement vector, and \( \epsilon \) is the strain vector.

For a plane stress problem,

\begin{equation}
[E] = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}
\end{equation}
The adjoint load becomes

\[
\int \int_\Omega \left\{ \sum_{i,j=1}^{2} \frac{\partial g_{ij}}{\partial \sigma_{ij}} \sigma_{ij}(\lambda) \right\} m_p \, d\Omega
\]  \hspace{1cm} (25)

where \( F \) is the adjoint equivalent nodal force. Since \( m_p \) is a characteristic function on the finite element \( \Omega_p \), \( F \) acts only on the nodal points of element \( \Omega_p \).

After the adjoint load is calculated for various elements, a restart of the finite element model for each load case corresponding to each element adjoint load is made. The strains resulting from these adjoint loads are then used in calculating Eq. (22) for the sensitivity of each functional.

When principal stress is selected as the functional,

\[
g_1 = \frac{\sigma_{11} + \sigma_{22}}{2} + \tau_{\text{max}}
\]  \hspace{1cm} (26)

\[
\tau_{\text{max}} = \left\{ \left[ (\sigma_{11} - \sigma_{22}/2)^2 + (\sigma_{12})^2 \right]^{1/2}\right\}
\]  \hspace{1cm} (27)

and

\[
\frac{\partial g_1}{\partial \sigma_{11}} = \frac{1}{2} + \frac{1}{4} \frac{\sigma_{11} - \sigma_{22}}{\tau_{\text{max}}}
\]

\[
\frac{\partial g_1}{\partial \sigma_{22}} = \frac{1}{2} - \frac{1}{4} \frac{\sigma_{11} - \sigma_{22}}{\tau_{\text{max}}}
\]  \hspace{1cm} (28)

\[
\frac{\partial g_1}{\partial \sigma_{12}} = \sigma_{12}/\tau_{\text{max}}
\]

and when von Mises' stress is selected as the functional,

\[
g_2 = \left[ (\sigma_{11})^2 - \sigma_{11} \sigma_{22} + (\sigma_{22})^2 + 3(\sigma_{12})^2 \right]^{1/2}
\]  \hspace{1cm} (29)
and

\[ \frac{\partial g_2}{\partial \sigma_{11}} = \frac{1}{2} (2\sigma_{11} - \sigma_{22})/g_2 \]

\[ \frac{\partial g_2}{\partial \sigma_{22}} = \frac{1}{2} (2\sigma_{22} - \sigma_{11})/g_2 \]

\[ \frac{\partial g_2}{\partial \sigma_{12}} = 3\sigma_{12}/g_2 \]  

(30)

2.1.2 Bending of Beams

Consider a cantilever beam with variable width and height and self weight, as shown in Fig. 4. The width and height are the design variables, \( u = [b(x), h(x)]^T \).

The energy bilinear form and the load linear form of the beam are

\[ a_u(z, \bar{z}) = \int_0^L E \frac{bh^3}{12} z_{xx} \bar{z}_{xx} \, dx \]  

(31)

and

\[ l_u(\bar{z}) = - \int_0^L (F + \gamma bh)\bar{z} \, dx \]  

(32)

Figure 4. Cantilever Beam with Variable Width and Height
where $\gamma$ is the weight density of the beam material, $F$ is the distributed load, $E$ is the modulus of elasticity of the beam material, $bh^3/12$ is the moment of inertia, $\bar{z}$ is the virtual displacement, $z_{xx}$ is the beam curvature, and $\bar{z}_{xx}$ is the beam curvature due to the virtual displacement $\bar{z}$.

The negative sign in the load linear equation is due to the fact that the load is applied in the $-z$ direction.

The state equation is [1]

$$a_u(z, \bar{z}) = \ell_u(\bar{z})$$

(33)

for all kinematically admissible virtual displacements $\bar{z}$.

First consider the functional representing the compliance of the structure as

$$\psi_4 = -\int_0^L (F + \gamma bh)z \, dx$$

(34)

The first variation of Eq. (34) is

$$\psi'_4 = -\int_0^L (F + \gamma bh)z' \, dx - \int_0^L \gamma yz \, dx \delta b - \int_0^L b yz \, dx \delta h$$

(35)

To replace the variation in state $z'$ by a virtual displacement $\lambda$, the adjoint equation is defined as [1]

$$a_u(\lambda, \lambda) = -\int_0^L (F + \gamma bh)\lambda \, dx$$

(36)

for all kinematically admissible virtual displacements $\lambda$. Since Eq. (36) is identical to Eq. (34), if $\lambda = z$ the adjoint equation does not need to be solved. Using the adjoint variable method of design sensitivity analysis gives
To numerically integrate Eqs. (34) and (37), a three-point Gauss quadrature formula is used. These equations become

$$\psi = \sum_{k=1}^{N} \left\{ \sum_{l=1}^{3} \left[ F_k + \gamma b_k \frac{h_k}{12} \right] z_{x_k} w_{x_k} \right\} \delta b_k$$

and

$$\psi' = \sum_{k=1}^{N} \left\{ \sum_{l=1}^{3} \left[ -2\gamma b_k z_{x_k} - \left( \frac{E b_k h_k^2}{12} \right) \frac{h_k}{12} \right] w_{x_k} J \right\} \delta b_k$$

where $N$ is the total number of elements, $k$ is the Gauss point counter, $w$ is the weighting constant for the $k$th Gauss point and $J$ is the Jacobian. The beam curvature $z_{x_k}$ is calculated using a cubic polynomial for the standard beam shape functions. Because the load is in the $-z$ direction, it is necessary to change the sign of the local element $y$-rotation $\theta_y$.

Next consider the functional representing the displacement $z$ at a discrete point $\hat{x}$ as

$$\psi_5 = z(\hat{x}) = \int_0^L \delta(x - \hat{x}) z(x) dx$$

where $\delta(x)$ is the Dirac measure at zero. The first variation of Eq. (40) is
The adjoint equation is defined as [1]

\[ a_u(\lambda, \bar{\lambda}) = \int_0^L \delta(x - \hat{x})\bar{\lambda}(x)\,dx \]  

(42)

for all kinematically admissible virtual displacements \( \bar{\lambda} \). Equation (42) has a unique solution \( \lambda^{(5)} \), where \( \lambda^{(5)} \) is the beam displacement due to a unit point load acting at a point \( \hat{x} \). Using the adjoint variable method of design sensitivity analysis gives

\[
\psi_5' = \int_0^L \left[ -h\gamma\lambda^{(5)} - (Eh^3/12)z_{xx} \lambda^{(5)} \right] \,dx \, \delta b
\]

\[ + \int_0^L \left[ -b\gamma\lambda^{(5)} - (3Ebh^2/12)z_{xx} \lambda^{(5)} \right] \,dx \, \delta h \]  

(43)

As in the membrane displacement constraint case, only Eq. (43) needs to be solved numerically. Using the three-point Gauss quadrature technique, Eq. (43) becomes

\[
\psi_5' = \sum_{k=1}^N \left\{ \sum_{\ell=1}^3 \left[ -h_{k,\ell} \gamma\lambda^{(5)} - (Eh^3_{k,\ell}/12)z_{xx} \lambda^{(5)} \right] W_\ell \right\} J \, \delta b_k
\]

(44)

\[ + \sum_{k=1}^N \left\{ \sum_{\ell=1}^3 \left[ -b_{k,\ell} \gamma\lambda^{(5)} - (3Ebh^2_{k,\ell}/12)z_{xx} \lambda^{(5)} \right] W_\ell \right\} J \, \delta h_k \]

Finally consider the functional representing the allowable stresses in the beam as

\[
\psi_6 = \int_0^L -\frac{i}{2} hEz_{xx} m_p \,dx \]

(45)
where \( h/2 \) is the half-depth of the beam, and \( m_p \) is a characteristic function defined on a finite element \( dx_p \) as

\[
m_p = \begin{cases} 
\frac{1}{\int_{dx_p}} &, & x \in dx_p \\
0 &, & x \notin dx_p 
\end{cases}
\]  

(46)

The first variation of Eq. (45) is

\[
\psi_6' = \int_0^L \left[ -\frac{1}{2} hE z'_m \left( z' \right) m - \frac{1}{2} E z'' \m \delta h \right] dx
\]

(47)

Replacing the variation in state \( z' \) by a virtual displacement \( \bar{\lambda} \), the adjoint equation is defined as

\[
a_u(\lambda, \bar{\lambda}) = -\int_0^L \frac{1}{2} hE \bar{\lambda}' z_{xx} m_p dx
\]

(48)

for all kinematically admissible virtual displacements \( \bar{\lambda} \). Equation (48) has a unique solution for a displacement field \( \lambda^{(6)} \). Using the adjoint variable method of design sensitivity analysis gives

\[
\psi_6' = \int_0^L \left[ -h \gamma \lambda^{(6)} - (Eh^3/12) z_{xx} \lambda^{(6)} \right] dx \delta b
\]

\[+ \int_0^L \left[ -\frac{1}{2} Ez_{xx} m_p - b \gamma \lambda^{(6)} - (3Ebh^2/12) z_{xx} \lambda^{(6)} \right] dx \delta h\]

(49)

where \( \lambda^{(6)} \) is the solution of Eq. (48).

With the three-point Gauss quadrature numerical integration technique, the integrals in Eqs. (45), (48), and (49) become

\[
\psi_6 = \sum_{k=1}^{N} \left\{ \sum_{l=1}^{3} \left( -\frac{1}{2} h_k E z_{xx}^m m_p \right) N_l \right\} J
\]

(50)
\[ \int_0^L \frac{1}{2} E \bar{\lambda}_{xx} m_p \, dx \]

\[ = \sum_{k=1}^N \left\{ \sum_{\ell=1}^3 \left( -\frac{1}{2} E [B]_{k,\ell} m_p \right) W_{k,\ell} J \delta b_k \right\} \]

and

\[ \psi'_6 = \sum_{k=1}^N \left\{ \sum_{\ell=1}^3 \left[ -h_k \gamma \lambda^{(6)}_{\ell} - \left( \frac{E h_k}{12} \right) z_{xx,\ell} \lambda^{(6)}_{xx,\ell} \right] W_{k,\ell} J \delta b_k \right\} \]

\[ + \sum_{k=1}^N \left\{ \sum_{\ell=1}^3 \left[ -\frac{1}{2} E z_{xx,\ell} m_p - h_k \gamma \lambda^{(6)}_{\ell} \right] W_{k,\ell} J \delta b_k \right\} \]

\[ - \left( 3E b_k h_k^2 / 12 \right) z_{xx,\ell} \lambda^{(6)}_{xx,\ell} W_{k,\ell} J \delta b_k \]  

where \([B]\) is the strain-displacement matrix, which is the second derivative of the shape functions.

### 2.1.3 Bending of Plates

Consider the clamped plate in Fig. 5 of variable thickness \( u = t(x) \), with a distributed load \( f(x) \) that consists of an externally applied pressure \( F(x) \) and self weight, given by [1]

\[ f(x) = F(x) + \gamma t(x) \]

where \( \gamma \) is the weight density of the material.

![Figure 5. Clamped plate of variable thickness \( t(x) \)](image-url)
For this design dependent loading, the energy bilinear form and the load linear form for the plate are given as [1]

\[
a_u(z, \bar{z}) = \iint_\Omega \hat{D}(u)[z_{11}\bar{z}_{11} + z_{22}\bar{z}_{22} + \nu(z_{22}\bar{z}_{11} + z_{11}\bar{z}_{22}) + 2(1 - \nu)z_{12}\bar{z}_{12}] d\Omega
\]

and

\[
\ell_u(\bar{z}) = \iint_\Omega [F + \gamma t]\bar{z} d\Omega
\]  

where \(\hat{D}(u) = Et^3/12(1-\nu^2)\) is the flexural rigidity, \(E\) is Young's modulus, \(\nu\) is Poisson's ratio, \(F\) is the externally applied pressure, and \(\gamma\) is the material density. The state equation is [1]

\[
a_u(z, \bar{z}) = \ell_u(\bar{z})
\]

for all kinematically admissible virtual displacements \(\bar{z}\).

First consider the functional representing the compliance of the structure as

\[
\psi_f = \iint_\Omega (F + \gamma t)z d\Omega
\]

The first variation of Eq. (57) is

\[
\psi_f' = \iint_\Omega [(F + \gamma t)z' + \gamma z \delta t] d\Omega
\]

The adjoint equation is defined as

\[
a_u(\lambda, \bar{\lambda}) = \iint_\Omega (F + \gamma t)\bar{\lambda} d\Omega
\]

for all kinematically admissible displacements \(\bar{\lambda}\). As in the previous cases (membranes and bending beams), Eq. (59) is identical to Eq. (57)
if $\tilde{\lambda} = z$. Using the adjoint variable method of design sensitivity analysis gives

$$\psi_j = \iint_{\Omega} [2yz - Et^2(z_1^2 + 2\nu z_{12} + 2(1 - \nu)z_{12}^2)]/4 \ 1 - \nu^2] \delta t \ d\Omega$$

$$= \iint_{\Omega} [2yz - \sum_{i,j=1}^{2} \sigma_{ij}^{1j}(z)\varepsilon_{ij}^{1j}(z)] \delta t \ d\Omega \quad (60)$$

where $\sigma_{ij}^{1j}(z)$ and $\varepsilon_{ij}^{1j}(z)$ are the stress and strain of the extreme fiber, given as

$$\varepsilon_{ij}^{1j} = -\frac{tz_{ij}}{2}, \quad i,j=1,2 \quad (61)$$

and

$$\sigma_{11}^{11} = -\frac{Et}{2(1 - \nu^2)} (z_{11} + \nu z_{22})$$

$$\sigma_{22}^{12} = -\frac{Et}{2(1 - \nu^2)} (z_{22} + \nu z_{11}) \quad (62)$$

$$\sigma_{12}^{12} = -\frac{Et}{2(1 + \nu)} z_{12}$$

To numerically integrate Eqs. (57) and (60), a one-point Gauss quadrature formula on a triangular element is used to correspond to the IFAD integration technique for this element. Equations (57) and (60) become

$$\psi_j = \sum_{k=1}^{N} [(F + \gamma t_k) z \ W] J \quad (63)$$

and

$$\psi_j = \sum_{k=1}^{N} [2yz - \sum_{i,j=1}^{2} \sigma_{ij}^{1j}(z)\varepsilon_{ij}^{1j}(z)] W J \delta t_k \quad (64)$$
Next consider the functional representing the displacement $z$ at a
discrete point $\hat{x}$ as

$$\psi_8 \equiv z(\hat{x}) = \int_\Omega \hat{\delta}(x - \hat{x})z(x)\,d\Omega \quad (65)$$

The first variation of Eq. (65) is

$$\psi'_8 = \int_\Omega \hat{\delta}(x - \hat{x})z'(x)\,dx \quad (66)$$

The adjoint equation is defined as

$$a_u(\lambda, \bar{\lambda}) = \int_\Omega \hat{\delta}(x - \hat{x})\bar{\lambda}(x)\,d\Omega \quad (67)$$

for all kinematically admissible virtual displacements $\bar{\lambda}$. This equation
has a unique solution $\lambda^{(8)}$, where $\lambda^{(8)}$ is the plate displacement due to
a unit vertical load acting at a point $\hat{x}$. Using the adjoint variable
method of design sensitivity analysis gives

$$\psi'_8 = \int_\Omega \left\{ \gamma \lambda^{(8)} - E\varepsilon^2 \left[ z_{11} \lambda_{11}^{(8)} + z_{22} \lambda_{22}^{(8)} + \nu(z_{11} \lambda_{22}^{(8)} + z_{22} \lambda_{11}^{(8)}) \right] + 2(1 - \nu)z_{12} \lambda_{12}^{(8)} \right\}/4(1 - \nu^2) \, d\Omega$$

$$= \int_\Omega \left[ \gamma \lambda^{(8)} - \sum_{i,j=1}^2 \sigma_{ij}(z) \varepsilon_{ij}(\lambda^{(8)}) \right] \delta t \, d\Omega \quad (68)$$

The same numerical integration procedure is used as in the
compliance case. Equation (68) becomes

$$\psi'_8 = \sum_{k=1}^N \left[ \gamma \lambda^{(8)} - \sum_{i,j=1}^2 \sigma_{ij}(z) \varepsilon_{ij}(\lambda^{(8)}) \right] w J \delta t_k \quad (69)$$

Finally, consider the functional representing a locally averaged
stress in the plate as
\[ \psi_g = \iint_{\Omega} g(\sigma(z)) m_p \, d\Omega \quad (70) \]

where \( g(\sigma(z)) \) may be principal stress, von Mises' stress, or some other material failure criteria and \( m_p \) is a characteristic function defined on a finite element \( \Omega_p \) as

\[
m_p = \begin{cases} 
\frac{1}{\int_{\Omega_p} d\Omega}, & x \in \Omega_p \\
0, & x \notin \Omega_p 
\end{cases} \quad (71)
\]

The first variation of Eq. (70) is

\[
\psi'_g = \iint_{\Omega} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma^{ij}} \sigma^{ij}(z') \right] m_p \, d\Omega \quad (72)
\]

The adjoint equation is defined as [1]

\[
a_u(\lambda, \lambda) = \iint_{\Omega} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma^{ij}} \sigma^{ij}(\lambda) \right] m_p \, d\Omega \quad (73)
\]

for all kinematically admissible virtual displacement \( \lambda \). Using the adjoint variable method of design sensitivity analysis gives

\[
\psi'_g = \iint_{\Omega} \left[ y^{(9)} - \sum_{i,j=1}^{2} \sigma^{ij}(z)e^{ij}(\lambda^{(9)}) \right] \delta t \, d\Omega + \iint_{\Omega} \frac{\partial g}{\partial t} \delta t \, d\Omega \quad (74)
\]

where \( \lambda^{(9)} \) is the solution of Eq. (73). For principal stress the last term on the right of Eq. (74) becomes
where $\tau_{\text{max}}$ is defined in Eq. (27). For von Mises' stress, this term becomes

$$
\iint_{\Omega} \left[ \left( \sigma_{11}^2 + \sigma_{22}^2 + 2\tau_{\text{max}} \right) / (2t) \right] m_p \, d\Omega
$$

(75)

The IFAD thin shell element is the element that is used for plate bending. The membrane effects can be eliminated so that only the bending term exists. This element is a hybrid element which uses a derivative smoothing technique [4]. The IFAD code evaluates the normal and shear stresses at the centroid of the triangle, but Ref. 4 stipulates that the stresses at the midside nodes of the triangle give the most accurate results. The integration of a function $\phi$ from its mid-stresses is [4]

$$
\int \phi \, dA = \frac{A}{3} \left[ \phi(0, \frac{1}{2}, \frac{1}{2}) + \phi(\frac{1}{2}, \frac{1}{2}, 0) + \phi(\frac{1}{2}, \frac{1}{2}, 0) \right]
$$

(77)

where $A$ is the area of the triangle and $(0, \frac{1}{2}, \frac{1}{2})$, $(\frac{1}{2}, 0, \frac{1}{2})$ and $(\frac{1}{2}, \frac{1}{2}, 0)$ are the area coordinates of the mid-side nodes. To achieve the most accurate stresses possible, this numerical integration technique is applied on Eqs. (70), (73), and (74). These equations become

$$
\Psi_g = \sum_{n=1}^{3} g(\sigma_n(z)) m_p \frac{A}{3}
$$

(78)

$$
\iint_{\Omega} \left[ \sum_{j=1}^{3} \frac{\partial \Psi_g}{\partial \sigma_{ij}} \sigma_{ij} \right] m_p \, d\Omega = \sum_{n=1}^{3} \left[ \frac{\partial \Psi_g}{\partial \sigma_{11}^n}, \frac{\partial \Psi_g}{\partial \sigma_{22}^n}, \frac{\partial \Psi_g}{\partial \sigma_{12}^n} \right]
$$
\[ \begin{aligned}
\{\mathbf{E}\} \mathbf{B} \frac{A}{3} \mathbf{p} \mathbf{d} &= \mathbf{F}^T \mathbf{d} \\
\end{aligned} \quad (79) \]

and

\[ \psi_k = \sum_{k=1}^N \left\{ \sum_{n=1}^3 \left[ \gamma \lambda_n^{(9)} - \frac{2}{3} \sum_{i,j=1}^{ij\eta_n} \overline{\lambda}^{ij}\eta_n \epsilon^{ij\eta_n^{(9)}} + \left( \frac{\partial g}{\partial t} \right)_n \right] \frac{A}{3} \right\} \delta t_k \quad (80) \]

where subscript \( n \) is the counter for the midside value. The term \( \left( ag/\partial t \right)_n \) for principal and von Mises' stress is

\[ \left[ (\sigma_{11} + \sigma_{22} + 2\tau_{\max})/(2\tau_{\max}) \right]_n \quad (81) \]

and

\[ \frac{1}{\tau_{\max}} \left[ (\sigma_{11}^2 - \sigma_{11}^2 \sigma_{22}^2 + (\sigma_{22}^2)^2 + 3(\sigma_{12}^2)^2)^{1/2} \right]_n \quad (82) \]

respectively.

To numerically calculate the adjoint load, the shape function of the element must be known so that \( \sigma_{ij}(\lambda) \) can be calculated using the finite element technique described in Section 2.1.1, Eqs. (23) and (25).

2.2 Calculation Procedure for a Built-Up Structure

The foundation of the built-up structure design sensitivity analysis method is the structural component analysis developed in Section 2.1. A built-up structure consists of various structural components that interact with each other. This interaction is taken into account by generalizing the individual components such that twisting, bending, transverse shear terms, etc. are included in the formulation of the sensitivity vector. Coordinate system precautions need to be taken to insure that the constraints and the sensitivity vectors are calculated correctly. In general, if the calculations are performed at
the local element coordinate system level, there will be no problem when components are oriented differently in the global coordinate system.

An example of a built-up structure using the structural components developed in the previous section would be a bending beam and plate problem, where a framework of beams could act as the supporting structure for the plate; i.e., a roof structure.

Figure 6 shows a built-up structure that has design variables $u = [b(x), h(x), t(x)]^T$, where $b(x)$ is the width of the beams, $h(x)$ is the height of the beams, and $t(x)$ is the thickness of the plates.

It is assumed that the plates are welded along the length of the beams. This would infer that the appropriate components in the finite element analysis must be chosen to insure kinematic compatibility along the component boundaries.

The energy bilinear form of the system equation is just the sum of the plate and beam energy bilinear equations, with an additional beam torsion term given as

$$
\begin{align*}
a_u(z, \tilde{z}) &= \sum \int \int \hat{D}(u) \left[ z_{xx} z_{xx} + z_{yy} z_{yy} \right] dx \Omega + \sum \int_0^L \frac{bh^3}{12} z_{xx} \tilde{z}_{xx} dx \\
&\quad + 2(1 - v)z_{xy} \tilde{z}_{xy} d\Omega + \sum \int_0^L E_b \frac{bh^3}{12} z_{xx} \tilde{z}_{xx} dx \\
&\quad + \sum \int_0^L GJ z_{xy} \tilde{z}_{xy} dx \\
&= \sum a_u(z, \tilde{z})_{\text{plate}} + \sum a_u(z, \tilde{z})_{\text{beam}} + \sum \int_0^L GJ z_{xy} \tilde{z}_{xy} dx
\end{align*}
$$

(83)
where \( a_u(z,z)_{\text{plate}} \) and \( a_u(z,z)_{\text{beam}} \) are the energy bilinear forms of the plate, Eq. (54) and beam, Eq. (31), respectively, \( G \) is the modulus of rigidity, \( J \) is the torsional moment of inertia of the beam, and \( R \) represents the local beam coordinate system, where \( x_R \) runs along the length of the beam.

In Eq. (83) \( z_{xy} \) represents the beam torsion term. Since each structural component needs to be solved individually to make the analysis feasible with an existing finite element code, a relationship between \( z_{xy} \) and the beam rotation has to be used. This kinematic compatibility states that for the beam and plate system the term has to be equivalent to the relative angle of twist over an element length.

Figure 6. Roof Structure
That is to say,
\[ z_{xy} = \left( \theta_2^x - \theta_1^x \right)/L \]  

where \( \theta_2^x \) is the local element rotation at node 2 of the beam, \( \theta_1^x \) is the local element rotation at node 1 of the beam, and \( L \) is the beam element length.

The load linear form of the system is
\[ \ell_{u}(\vec{z}) = \int_{\Omega} [F_p + \gamma_p t + \gamma_b bh] \vec{z} \, d\Omega \]  

where \( F_p \) is the externally applied plate pressure, \( \gamma_p \) and \( \gamma_b \) are the material densities for the plates and beams respectively, and \( \vec{z} \) is the virtual displacement. The state equation is [1]
\[ a_u(z,\vec{z}) = \ell_{u}(\vec{z}) \]  

for all kinematically admissible virtual displacements \( \vec{z} \).

Since the energy bilinear form of the system equation is just the addition of each structural component's energy bilinear forms, the design sensitivity equation of the system turns out also to be an additive process. The generalized design sensitivity of the built-up structure is
\[ \psi' = \psi'_b \delta b + \psi'_h \delta h + \psi'_t \delta t \]  

The only necessary step to calculating this value is the reformulation of the beam to include the torsional term. The energy bilinear form of the beam component becomes
\[ a_u(z,\vec{z}) = \int_0^L E(bh^3/12)z \, \vec{z}_{xx} \, dx + \int_0^L GJ \, z_{xy}^2 \, dx \]
where Eq. (84) defines $z_{xy}$.

If the compliance, displacement, and stress functionals of the beam - Eqs. (34), (40), and (45), respectively - remain the same, the only additional term in the design sensitivity analysis [1] is due to the differentiation of the torsional terms in the energy bilinear equation with respect to the design variables $b$ and $h$. This term is defined as

$$\int_0^L \left[ \frac{\partial J}{\partial h} \delta h + \frac{\partial J}{\partial b} \delta b \right] G z_{xy} \lambda_{xy} \, dx$$

(89)

For a beam with a rectangular cross section [5]

$$J = bh^3 \left[ \frac{1}{3} - 0.21(b/h)(1 - \frac{b^4}{12h^4}) \right]$$

(90)

and the derivatives with respect to the design variables are

$$\frac{\partial J}{\partial b} = \frac{h^3}{3} - 0.042b \left( h^2 + \frac{b^4}{4h^2} \right)$$

(91)

and

$$\frac{\partial J}{\partial b} = bh^2 - 0.42 b^2 \left( h - \frac{b^4}{12h^3} \right)$$

(92)

The compliance sensitivity of Eq. (37) becomes

$$\psi'_4 = \int_0^L \left[ -2\gamma h z - \left( Eh^3 / 12 \right) (z_{xx})^2 - \frac{\partial J}{\partial b} G(z_{xy})^2 \right] dx \, \delta b$$

$$+ \int_0^L \left[ -2\gamma b z - \left( 3Eb^2 / 12 \right) (z_{xx})^2 - \frac{\partial J}{\partial h} G(z_{xy})^2 \right] dx \, \delta h$$

(93)
the displacement sensitivity of Eq. (43) becomes

\[ \psi_5^t = \int_0^L \left[ -h \gamma \lambda(5) - \frac{E h^3}{12} z_{xx} \lambda(5) \frac{\partial}{\partial \delta b} G z_{xy} \lambda_x(5) \right] dx \delta b \]

\[ + \int_0^L \left[ -2 \gamma b \lambda(5) - \frac{3E bh^2}{12} z_{xx} \lambda_x(5) \frac{\partial}{\partial \delta h} G z_{xy} \lambda_y(5) \right] dx \delta h \]

(94)

and the stress sensitivity of Eq. (49) becomes

\[ \psi_6^t = \int_0^L \left[ -h \gamma \lambda(6) \frac{E h^3}{12} z_{xx} \lambda(6) \frac{\partial}{\partial \delta b} G z_{xy} \lambda_x(6) \right] dx \delta b \]

\[ + \int_0^L \left[ -\frac{1}{2} E z_{xx} m^p - b \gamma \lambda(6) - \frac{3E bh^2}{12} z_{xx} \lambda_x(6) \frac{\partial}{\partial \delta h} G z_{xy} \lambda_y(6) \right] dx \delta h \]

(95)

when the torsional term is added, where \( \lambda(5) \) and \( \lambda(6) \) are the solutions to the adjoint Eqs. (42) and (48), respectively.

Caution has to be taken when the constraint functionals on the system are defined. When a von Mises' stress functional is specified for a particular plate element, the allowable beam bending stress term

\[ - \int_0^L \frac{1}{2} E z_{xx} m \ d x \delta h \]

must be removed from Eq. (95), so that the design sensitivity of Eq. (87) is calculated only for the von Mises' stress functional. In the same way, if an allowable beam bending stress functional is specified for a particular beam element, the von Mises' stress term \( \partial g/\partial t \) of Eq. (76) must be removed from Eq. (74), so that the design sensitivity of Eq. (87) is calculated only for the allowable beam bending stress functional. The application of this procedure is shown below for a von Mises' stress functional, a compliance functional and a displacement functional on a plate element and an allowable
bending stress functional on a beam element of the built-up structure of Fig. 6.

The functional representing a von Mises' stress constraint on a plate element is

$$
\psi_{10} = \iint_{\Omega} \left[ \sigma_{xx}^2 + \sigma_{xx} \sigma_{yy} + \sigma_{yy}^2 + 3 \sigma_{xy}^2 \right]^{1/2} m_p \, d\Omega
$$

(96)

The adjoint load linear form is defined as

$$
\iint_{\Omega} \left[ \sum_{i,j=1}^{2} \frac{\partial g}{\partial \sigma_{ij}} \sigma_{ij}(\lambda) \right] m_p \, d\Omega

= \left[ \frac{\partial g}{\partial \sigma_{xx}} , \frac{\partial g}{\partial \sigma_{yy}} , \frac{\partial g}{\partial \sigma_{xy}} \right] [E][B] m_p \, d\Omega \, \bar{d}
$$

(97)

where $\partial g/\partial \sigma_{ij}$ is defined in Eq. (30) and $F$ is the adjoint equivalent nodal force. Using Eq. (87) gives the design sensitivity of the roof structure as

$$
\psi'_{10} = \sum \iint_{\Omega} [g_{\lambda}^{(10)} - \sum_{i,j=1}^{2} \sigma_{ij}(z) \varepsilon_{ij}(\lambda^{(10)}) + \frac{\partial g}{\partial t}] \delta t \, d\Omega

+ \sum \int_{0}^{L} [-h g_{\lambda}^{(10)} - \frac{Eh^3}{12} \varepsilon_{xx} \lambda_{xx}^{(10)} - \frac{3J}{2h} G z_{xy} \lambda_{xy}^{(10)}] dt \, \delta b

+ \sum \int_{0}^{L} [-b g_{\lambda}^{(10)} - \frac{3Ebh^2}{12} \varepsilon_{xx} \lambda_{xx}^{(10)} - \frac{3J}{2h} G z_{xy} \lambda_{xy}^{(10)}] dt \, \delta h
$$

(98)

where subscript $L$ refers to the beam local coordinate system and $\partial g/\partial t$ is defined in Eq. (76).

The functional representing a compliance constraint on a plate element is
where $\gamma t$ and $\gamma bh$ are the self weight of the plate and beam, respectively. The adjoint load linear form is defined as

$$a_u(\lambda, \bar{\lambda}) = \iint_\Omega (F + \gamma t + \gamma bh) \bar{\lambda} \, d\Omega$$

(100)

Since Eq. (100) is identical to Eq. (99) if $\bar{\lambda} = z$, the adjoint equation does not need to be solved. Using Eq. (87) gives the design sensitivity of the built-up structure as

$$\psi_{11} = \sum \iint_\Omega [2\gamma z - \sum_{i,j=1}^{2} \sigma_{ij}^{ij}(z) e_{ij}(z)] \delta t \, d\Omega$$

$$+ \sum \int_0^L [-2\gamma bz - \frac{Eh^3}{12}(z_{xx})^2 - \frac{3Ebh^2}{12}(z_{xx}) \delta x \delta b] \, dx \delta b$$

$$+ \sum \int_0^L [-2\gamma bz - (3Ebh^2/12)(z_{xx})^2 - \frac{3Ebh^2}{12}(z_{xx}) \delta x \delta b] \, dx \delta b$$

(101)

where subscript $b$ refers to the beam local coordinate system.

The functional representing the displacement $z$ at a discrete point $\hat{x}$ is

$$\psi_{12} \equiv z(\hat{x}) = \iint_\Omega \delta(x - \hat{x})z(x) \, d\Omega$$

(102)

The adjoint load linear form is defined as

$$a_u(\lambda, \bar{\lambda}) = \iint_\Omega \delta(x - \hat{x}) \bar{\lambda}(x) \, d\Omega$$

(103)

for all kinematically admissible virtual displacements $\bar{\lambda}$. This equation has a unique solution $\lambda^{(12)}$, where $\lambda^{(12)}$ is the plate displacement due
to a unit vertical load acting at a point $x$. Using Eq. (87) gives the design sensitivity of the built-up structure as

$$
\psi_{12}' = \sum \int_D \left[ \gamma(12) \dot{\lambda}^{(12)} - \frac{2}{i,j=1} \sigma_{ij}(z) \dot{\epsilon}_{ij}(\lambda^{(12)}) \right] d\Omega
$$

$$
+ \sum \int_0^L \left[ -h\gamma(12) - \frac{Eh^3}{12} z_{xx} \dot{\lambda}^{(12)} + \frac{\partial}{\partial b} G z_{xy} \dot{\lambda}_{xy}^{(12)} \right] dx \frac{\delta b}{\delta b}
$$

$$
+ \sum \int_0^L \left[ \frac{3Ebh^2}{12} z_{xx} \dot{\lambda}^{(12)} + \frac{\partial}{\partial n} G z_{xy} \dot{\lambda}_{xy}^{(12)} \right] dx \frac{\delta h}{\delta h}
$$

where subscript 1 refers to the beam local coordinate system.

The functional representing an allowable bending stress on a beam element is

$$
\psi_{13} = - \int_0^L \frac{1}{2} hEz_{xx} m_p \, dx
$$

The adjoint load linear form is defined as

$$
\int_0^L \frac{1}{2} hEz_{xx} m_p \, dx = \int_0^L \frac{1}{2} hE[B]m_p \, dx = F^T d
$$

where $F$ is the adjoint equivalent nodal force. Using Eq. (87) gives the design sensitivity of the built-up structures as

$$
\psi_{13}' = \sum \int_D \left[ \gamma(13) \dot{\lambda}^{(13)} - \frac{2}{i,j=1} \sigma_{ij}(z) \dot{\epsilon}_{ij}(\lambda^{(13)}) \right] d\Omega
$$

$$
+ \sum \int_0^L \left[ -h\gamma(13) - \frac{Eh^3}{12} z_{xx} \dot{\lambda}^{(13)} + \frac{\partial}{\partial b} G z_{xy} \dot{\lambda}_{xy}^{(13)} \right] dx \frac{\delta b}{\delta b}
$$
\[ + \sum \int_0^L \left[ -b_\gamma \lambda^{(13)} - \frac{1}{2} E_{zz} m - \frac{3Ebh^2}{12} \right] \left( \frac{\partial^2}{\partial x^2} \right) \lambda^{(13)} \]

\[ - \frac{\partial J}{\partial h} G_{xy} \lambda^{(13)} \right] dx \delta h \]

where subscript \( \ell \) refers to the beam local coordinate system.
CHAPTER III
NUMERICAL EXAMPLES

The design sensitivity of a constraint functional $\psi$ is the differential $\psi'$ of the constraint functional. In order to make sure the design sensitivity is accurate, an approximation $\Delta \psi$ is made using the finite difference method. It is very important that an appropriate perturbed design variable $\delta u$ is selected. If $\delta u$ is too small, the change in the constraint functional $\Delta \psi$ may be inaccurate due to losses of significant digits. If $\delta u$ is too large, $\Delta \psi$ will be influenced by nonlinearities in the constraint functional, which in turn will cause an inaccurate design sensitivity prediction.

In all the following examples, perturbations in design of 1% and/or 5% are used, with the exception of the plate bending and built up structure examples. Because of the nonlinearity characteristics of built-up structural response, the perturbation of 0.1% was used.

3.1 Membranes

The finite element membrane model in Fig. 7 is a simple plane elastic solid that is restrained at one end and loaded with a distributed tensile load at the other end. It contains 80 isoparametric elements (IFAD plane stress element type 1104), 289 nodal points, and 560 degrees-of-freedom, with the design variable being the variable thickness $u = h(x)$. The material property constraints, Young's modulus,
and Poission's ratio are given as $E = 3 \times 10^7$ psi and $\nu = 0.3$, respectively. Each finite element is discretized so that a constant thickness of $h = 0.5$ in. is used in order to simplify the sensitivity calculation.

The compliance sensitivity results are shown in Table 1, where $\Delta \psi_1 = \psi_1(h + \delta h) - \psi(h)$ and $\psi'$ and is the predicted value calculated from Eq. (9), with design perturbations of $\delta h = 0.01h$ and $\delta h = 0.05h$. The percent accuracy of the sensitivity prediction is calculated using $\psi' \times 100/\Delta \psi_1$.

<table>
<thead>
<tr>
<th>$\delta h$</th>
<th>$\psi_1(h)$</th>
<th>$\psi_1(h+\delta h)$</th>
<th>$\Delta \psi_1$</th>
<th>$\psi'$</th>
<th>$\psi' \times 100/\Delta \psi_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01h</td>
<td>265.302</td>
<td>262.676</td>
<td>-2.627</td>
<td>-2.653</td>
<td>101.0</td>
</tr>
<tr>
<td>0.05h</td>
<td>265.302</td>
<td>252.668</td>
<td>-12.632</td>
<td>-13.265</td>
<td>105.0</td>
</tr>
</tbody>
</table>

Several discrete points shown in Fig. 6 are selected to check accuracy of the design sensitivity of the displacement functional of Eq. (14). In order to calculate this equation, the strain $\varepsilon_{ij}$ due to the adjoint load is needed. Since the adjoint load is just a unit point load at point $\hat{x}$, acting in the direction of the displacement, a restart of the finite element analysis is all that is needed. For every node direction, there is a separate load case that produces a strain $\varepsilon_{ij}$, which in turn is used to calculate sensitivity of displacement. Design sensitivity predictions and differences, with $\delta h = 0.05h$ are given in Table 2.
Table 2. Design Sensitivity Check for Displacement

<table>
<thead>
<tr>
<th>Node No.</th>
<th>Dir</th>
<th>( \psi_2(h) )</th>
<th>( \psi_2(h+\delta h) )</th>
<th>( \Delta \psi_2 )</th>
<th>( \psi'_2 )</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>x1</td>
<td>2.974E-04</td>
<td>2.832E-04</td>
<td>-1.416E-04</td>
<td>-1.487E-05</td>
<td>105.0</td>
</tr>
<tr>
<td>27</td>
<td>x1</td>
<td>4.058E-04</td>
<td>3.865E-04</td>
<td>-1.932E-05</td>
<td>-2.029E-05</td>
<td>105.0</td>
</tr>
<tr>
<td>27</td>
<td>x2</td>
<td>-2.248E-04</td>
<td>-2.141E-04</td>
<td>1.071E-05</td>
<td>1.124E-05</td>
<td>105.0</td>
</tr>
<tr>
<td>185</td>
<td>x1</td>
<td>3.298E-03</td>
<td>3.141E-03</td>
<td>-1.571E-04</td>
<td>-1.649E-04</td>
<td>105.0</td>
</tr>
<tr>
<td>187</td>
<td>x1</td>
<td>3.299E-03</td>
<td>3.142E-03</td>
<td>-1.571E-04</td>
<td>-1.650E-04</td>
<td>105.0</td>
</tr>
<tr>
<td>187</td>
<td>x2</td>
<td>-2.014E-04</td>
<td>-1.918E-04</td>
<td>0.959E-05</td>
<td>1.007E-05</td>
<td>105.0</td>
</tr>
<tr>
<td>365</td>
<td>x1</td>
<td>6.633E-03</td>
<td>6.317E-03</td>
<td>-3.158E-04</td>
<td>-3.316E-04</td>
<td>105.0</td>
</tr>
<tr>
<td>369</td>
<td>x1</td>
<td>6.633E-03</td>
<td>6.317E-03</td>
<td>-3.158E-04</td>
<td>-3.316E-04</td>
<td>105.0</td>
</tr>
<tr>
<td>369</td>
<td>x2</td>
<td>-4.000E-04</td>
<td>-3.809E-04</td>
<td>1.905E-05</td>
<td>2.000E-05</td>
<td>105.0</td>
</tr>
</tbody>
</table>

To check the stress constraint sensitivity of Eq. (22), the equivalent nodal force of the adjoint load on the right of Eq. (25) has to be calculated so that \( \psi_2^{ij}(\lambda^{(3)}) \) is known for each constrained element. This is accomplished by a restart of the original IFAD model, with each adjoint load for an element being a separate loading case. Design sensitivity results for principal and von Mises' stress functionals are given in Table 3 for several finite elements. The perturbations are \( \delta h = 0.01h \) and \( \delta h = 0.05h \) for von Mises' stress and \( \delta h = 0.05h \) for principal stress.

The design sensitivity calculation is performed using double precision accuracy. Since the finite difference approximation in Tables 1, 2, and 3 are no smaller than two significant digits of the actual constraint functional, loss of significant digits is minimal. Therefore, the design perturbation is not to small. With all three
### Table 3. Design Sensitivity Check for Stress

(a) von Mises' Stress with $\delta h = 0.01h$

<table>
<thead>
<tr>
<th>El. No.</th>
<th>$\psi_3(h)$</th>
<th>$\psi_3(h + \delta h)$</th>
<th>$\Delta \psi_3$</th>
<th>$\psi'_3 (\psi'_3/\psi_3 \times 100%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9888.882</td>
<td>9790.973</td>
<td>-97.910</td>
<td>-98.889</td>
</tr>
<tr>
<td>10</td>
<td>9989.932</td>
<td>9891.022</td>
<td>-98.910</td>
<td>-99.899</td>
</tr>
<tr>
<td>20</td>
<td>9999.982</td>
<td>9900.972</td>
<td>-99.010</td>
<td>-100.000</td>
</tr>
<tr>
<td>21</td>
<td>8752.850</td>
<td>8666.188</td>
<td>-86.662</td>
<td>-87.528</td>
</tr>
<tr>
<td>30</td>
<td>10024.586</td>
<td>9925.333</td>
<td>-99.253</td>
<td>-100.246</td>
</tr>
<tr>
<td>40</td>
<td>9999.853</td>
<td>9900.845</td>
<td>-99.008</td>
<td>-100.000</td>
</tr>
</tbody>
</table>

(b) von Mises' Stress with $\delta h = 0.05h$

<table>
<thead>
<tr>
<th>El. No.</th>
<th>$\psi_3(h)$</th>
<th>$\psi_3(h + \delta h)$</th>
<th>$\Delta \psi_3$</th>
<th>$\psi'_3 (\psi'_3/\psi_3 \times 100%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9888.882</td>
<td>9417.984</td>
<td>-470.899</td>
<td>-494.444</td>
</tr>
<tr>
<td>10</td>
<td>9989.932</td>
<td>9514.222</td>
<td>-475.711</td>
<td>-499.497</td>
</tr>
<tr>
<td>20</td>
<td>9999.982</td>
<td>9523.793</td>
<td>-476.189</td>
<td>-499.998</td>
</tr>
<tr>
<td>21</td>
<td>8752.850</td>
<td>8336.048</td>
<td>-416.802</td>
<td>-437.642</td>
</tr>
<tr>
<td>40</td>
<td>9999.853</td>
<td>9523.670</td>
<td>-476.813</td>
<td>-499.993</td>
</tr>
</tbody>
</table>

(c) Principal Stress with $\delta h = 0.05h$

<table>
<thead>
<tr>
<th>El. No.</th>
<th>$\psi_3(h)$</th>
<th>$\psi_3(h + \delta h)$</th>
<th>$\Delta \psi_3$</th>
<th>$\psi'_3 (\psi'_3/\psi_3 \times 100%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10582.770</td>
<td>10078.829</td>
<td>-503.941</td>
<td>-529.138</td>
</tr>
<tr>
<td>10</td>
<td>9987.061</td>
<td>9511.487</td>
<td>-475.574</td>
<td>-499.353</td>
</tr>
<tr>
<td>20</td>
<td>10000.013</td>
<td>9523.823</td>
<td>-476.191</td>
<td>-500.001</td>
</tr>
<tr>
<td>21</td>
<td>9660.279</td>
<td>9200.266</td>
<td>-460.013</td>
<td>-483.014</td>
</tr>
<tr>
<td>30</td>
<td>10012.976</td>
<td>9536.617</td>
<td>-476.808</td>
<td>-500.649</td>
</tr>
<tr>
<td>40</td>
<td>9999.987</td>
<td>9523.797</td>
<td>-476.189</td>
<td>-500.000</td>
</tr>
</tbody>
</table>
constraint functionals, the design sensitivity results compared to the finite difference approximation are excellent. This infers that the design perturbation is not too large as to cause significant nonlinearity effects in the calculation of design sensitivity.

It is interesting to note that in Tables 1, 2, and 3 the finite difference approximation is nearly 1% of the constraint functional when $\delta h = 0.01h$ and nearly 5% of the constraint functional when $h = 0.05h$. The results also show that as $\delta h$ approaches zero, $\psi'/\Delta \psi$ approaches one.

3.2 Bending of Beams

A cantilevered beam finite element model shown in Fig. 8 is loaded with a constant distributed force $f(x) = 0.03$ lb/in. along the entire length of the beam. It contains 20 IFAD beam elements of type 0501, each 3" in length, 121 nodal points, and 40 degrees-of-freedom, with design variables $u = [b(x), h(x)]^T$, the width and height of the beam. In Fig. 8, the element numbers are along the top of the beam and the node numbers are along the bottom of the beam. The beam has a rectangular cross-section with constant width and height $b = 0.5$ in. and $h = 0.75$ in., respectively. This gives the moment of inertia $I_y = 0.01758$ in.$^4$ and the cross-sectional area $A_c = 0.375$ in.$^2$. The material property constants, Young's modulus and Poisson's ratio are $E = 3\times10^7$ psi and $\nu = 0.3$, respectively. Self weight is included in the analysis.
Figure 8. Cantilever Beam Finite Element Model

The compliance sensitivity results are shown in Table 4, where
\[
\Delta \psi_4 = \psi_4(u + \delta u) - \psi_4(u)
\]
and \(\psi'_4\) is the predicted value calculated from Eq. (39), with design perturbations of \(\delta b = 0.05b\), \(\delta h = 0.05h\) and \(\delta b = 0.01b\), \(\delta h = 0.01h\).

Table 4. Beam Design Sensitivity Check for Compliance

<table>
<thead>
<tr>
<th>(\delta h)</th>
<th>(\delta b)</th>
<th>(\psi_4(u))</th>
<th>(\psi_4(u+\delta u))</th>
<th>(\Delta \psi_4)</th>
<th>(\psi'_4)</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05h</td>
<td>0.05b</td>
<td>1.3684</td>
<td>1.3153</td>
<td>-0.0531</td>
<td>-0.0601</td>
<td>113.1</td>
</tr>
<tr>
<td>0.01h</td>
<td>0.01b</td>
<td>1.3684</td>
<td>1.3566</td>
<td>-0.0118</td>
<td>-0.0120</td>
<td>102.0</td>
</tr>
</tbody>
</table>

These results show that nonlinearities in the compliance of the cantilevered beam are highly evident. If too large a perturbation in design is chosen, the sensitivity will be inaccurate.
Several discrete points along the beam are selected to check the accuracy of design sensitivity of the displacement functional of Eq. (44). In order to calculate this equation, the beam curvature due to the adjoint load is needed. Since the adjoint load is just a unit point load at the point \( x \) acting in the \(-z\) direction, a restart of the finite element analysis is all that is needed. Displacement results are shown in Table 5 for design perturbations of \( \delta b = 0.05b, \delta h = 0.05h \) and \( \delta b = 0.01b, \delta h = 0.01h \).

In both Tables 5(a) and 5(b), results show that the design sensitivity compared to the finite difference approximation is good, with the exception of node 3. Since the finite difference approximation is not too small in comparison to the constraint functional, loss of significant digits is not a valid reason for this inconsistency. The accuracy decreases as the node location approaches the restrained end of the beam. This is most likely due to the restraining effect, which causes rigidity in the beam and in turn gives smaller deflections.

To check the stress constraint sensitivity of Eq. (52), the equivalent nodal force of the adjoint load on the right side of Eq. (51) has to be calculated so that the curvature \( \lambda_{xx}^{(6)} \) is known for each constraint element. This is accomplished by a restart of the original IFAD model, with each adjoint load for each element being a separate loading case. Allowable bending stress results for several finite elements are shown in Table 6 for a design perturbation of \( \delta h = 0.05h \) and \( \delta b = 0.05b \).
Table 5. Beam Design Sensitivity Check for Displacement

(a) $\delta b = 0.01b$ and $\delta h = 0.01h$

<table>
<thead>
<tr>
<th>Node</th>
<th>$\psi_5(u)$</th>
<th>$\psi_5(u+\delta u)$</th>
<th>$\Delta\psi_5$</th>
<th>$\psi'_5$</th>
<th>$\psi'_5 \times 100/\Delta\psi_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>7.8353E-03</td>
<td>7.6473E-03</td>
<td>-1.8804E-04</td>
<td>-1.6082E-04</td>
<td>85.5</td>
</tr>
<tr>
<td>6</td>
<td>4.4162E-02</td>
<td>4.3102E-00</td>
<td>-1.0604E-03</td>
<td>-9.9992E-04</td>
<td>94.2</td>
</tr>
<tr>
<td>9</td>
<td>1.0181E-01</td>
<td>9.9365E-02</td>
<td>-2.4451E-03</td>
<td>-2.3554E-03</td>
<td>96.3</td>
</tr>
<tr>
<td>12</td>
<td>1.7317E-01</td>
<td>1.6901E-01</td>
<td>-4.1590E-03</td>
<td>-4.0441E-03</td>
<td>97.2</td>
</tr>
<tr>
<td>15</td>
<td>2.5229E-01</td>
<td>2.4623E-01</td>
<td>-6.0594E-03</td>
<td>-5.9211E-03</td>
<td>97.7</td>
</tr>
<tr>
<td>18</td>
<td>3.3493E-01</td>
<td>3.2686E-01</td>
<td>-8.0447E-03</td>
<td>-7.8836E-03</td>
<td>97.9</td>
</tr>
<tr>
<td>21</td>
<td>4.1857E-01</td>
<td>4.0852E-01</td>
<td>-1.0054E-02</td>
<td>-9.8701E-03</td>
<td>98.2</td>
</tr>
</tbody>
</table>

(b) $\delta b = 0.05b$ and $\delta h = 0.05h$

<table>
<thead>
<tr>
<th>Node</th>
<th>$\psi_5(u)$</th>
<th>$\psi_5(u+\delta u)$</th>
<th>$\Delta\psi_5$</th>
<th>$\psi'_5$</th>
<th>$\psi'_5 \times 100/\Delta\psi_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>7.8353E-03</td>
<td>6.9743E-03</td>
<td>-8.6100E-04</td>
<td>-8.0412E-04</td>
<td>93.4</td>
</tr>
<tr>
<td>6</td>
<td>4.4162E-02</td>
<td>3.9306E-02</td>
<td>-4.8555E-03</td>
<td>-4.9959E-04</td>
<td>102.9</td>
</tr>
<tr>
<td>9</td>
<td>1.0181E-01</td>
<td>9.0619E-02</td>
<td>-1.120E-02</td>
<td>-1.1777E-02</td>
<td>105.2</td>
</tr>
<tr>
<td>12</td>
<td>1.7317E-01</td>
<td>1.5412E-01</td>
<td>-1.9043E-02</td>
<td>-2.0220E-02</td>
<td>106.2</td>
</tr>
<tr>
<td>15</td>
<td>2.5229E-01</td>
<td>2.2454E-01</td>
<td>-2.7745E-02</td>
<td>-2.9605E-02</td>
<td>106.7</td>
</tr>
<tr>
<td>18</td>
<td>3.3493E-01</td>
<td>2.9810E-01</td>
<td>-3.6835E-02</td>
<td>-3.9418E-02</td>
<td>107.0</td>
</tr>
<tr>
<td>21</td>
<td>4.1857E-01</td>
<td>3.7253E-01</td>
<td>-4.6034E-02</td>
<td>-4.9350E-02</td>
<td>107.2</td>
</tr>
</tbody>
</table>

Table 6. Beam Design Sensitivity Check for Stress

<table>
<thead>
<tr>
<th>El.</th>
<th>$\psi_6(u)$</th>
<th>$\psi_6(u+\delta u)$</th>
<th>$\Delta\psi_6$</th>
<th>$\psi'_6$</th>
<th>$\psi'_6 \times 100/\Delta\psi_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4932.767</td>
<td>4609.263</td>
<td>-323.504</td>
<td>-349.234</td>
<td>108.0</td>
</tr>
<tr>
<td>5</td>
<td>3110.235</td>
<td>2906.201</td>
<td>-204.055</td>
<td>-219.226</td>
<td>107.4</td>
</tr>
<tr>
<td>10</td>
<td>1420.622</td>
<td>1327.375</td>
<td>-93.287</td>
<td>-99.103</td>
<td>106.2</td>
</tr>
<tr>
<td>15</td>
<td>385.008</td>
<td>359.664</td>
<td>-25.344</td>
<td>-26.076</td>
<td>102.9</td>
</tr>
<tr>
<td>20</td>
<td>3.295</td>
<td>3.067</td>
<td>-0.228</td>
<td>-0.147</td>
<td>64.7</td>
</tr>
</tbody>
</table>
Near the end of the beam, where the allowable bending stress is near zero, the design sensitivity decreases significantly. Since the design sensitivity is the derivative of the constraint functional, and the derivative is physically interpreted as the slope of the beam, this decrease can be attributed to the large increase in slope at the free end of the beam.

3.3 Bending of Plates

The clamped plate finite element model shown in Fig. 9 is uniformly loaded with a pressure $f(x) = -1.5$ lb/in. in the $z$ direction. Since the model is symmetric along two planes, only one quarter of it needs to be analyzed and symmetric boundary conditions need to be applied. The quarter model contains 100 IFAD triangular thin shell elements of type 1601, with only the bending terms active. It has 61 nodal points and 140 degrees-of-freedom.

The design variable is the plate thickness $u = t(x)$, and the material property constants, Young's Modulus and Poisson's ratio, are $E = 30.5 \times 10^6$ psi and $\nu = 0.3$, respectively. The constant plate thickness is $t = 0.4$ in. and the self-weight of the plate is neglected.

Compliance sensitivity results are shown in Table 7, where $\Delta \psi_7 = \psi_7(t + \delta t) - \psi_7(t)$ and $\psi_7^*$ is the predicted value calculated from Eq. (60), with design perturbations of $\delta t = 0.01t$ and $\delta t = 0.05t$.

Both perturbations for the compliance constraint functional give good correlation between the design sensitivity and the finite difference approximation. This implies that a five percent change in thickness is acceptable when making design improvements.
Figure 9. Bending Plate Finite Element Model
Table 7. Plate Design Sensitivity Check for Compliance

<table>
<thead>
<tr>
<th>δt</th>
<th>$\psi_7(0)$</th>
<th>$\psi_7(t+\delta t)$</th>
<th>$\Delta \psi_7$</th>
<th>$\psi'_7$</th>
<th>$\psi'_7 \times 100/\Delta \psi_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>.05t</td>
<td>4.9625</td>
<td>4.2868</td>
<td>-0.6757</td>
<td>-0.7204</td>
<td>106.6</td>
</tr>
<tr>
<td>.01t</td>
<td>4.9625</td>
<td>4.8165</td>
<td>-0.1459</td>
<td>-0.1441</td>
<td>98.7</td>
</tr>
</tbody>
</table>

Several discrete points in Fig. 8 are selected to check accuracy of the design sensitivity of the displacement functional in Eq. (69). In order to calculate this equation, just as in the membrane case, the strain $\varepsilon^{ij}(\lambda^{(8)})$ due to the adjoint load is needed. A restart of the finite element analysis, using a unit point load in the $-z$ direction for each selected point as a separate load case, accomplishes this task. Some displacement results are shown in Table 8 for a design perturbation of $\delta t = 0.01t$.

Table 8. Plate Design Sensitivity Check for Displacement

<table>
<thead>
<tr>
<th>Node</th>
<th>$\psi_8(t)$</th>
<th>$\psi_8(t+\delta t)$</th>
<th>$\Delta \psi_8$</th>
<th>$\psi'_8$</th>
<th>$\psi'_8 \times 100/\Delta \psi_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>1.9033E-03</td>
<td>1.8473E-03</td>
<td>-5.5976E-05</td>
<td>-5.2596E-05</td>
<td>94.0</td>
</tr>
<tr>
<td>15</td>
<td>3.0497E-03</td>
<td>2.9600E-03</td>
<td>-8.9692E-05</td>
<td>-8.5144E-05</td>
<td>94.9</td>
</tr>
<tr>
<td>24</td>
<td>1.9033E-03</td>
<td>1.8473E-03</td>
<td>-5.5976E-05</td>
<td>-5.2596E-05</td>
<td>94.0</td>
</tr>
<tr>
<td>27</td>
<td>1.1258E-02</td>
<td>1.0927E-02</td>
<td>-3.3109E-04</td>
<td>-3.2620E-04</td>
<td>98.5</td>
</tr>
<tr>
<td>31</td>
<td>9.7772E-03</td>
<td>9.4897E-03</td>
<td>-2.8754E-04</td>
<td>-2.8309E-04</td>
<td>98.4</td>
</tr>
<tr>
<td>35</td>
<td>3.0497E-03</td>
<td>2.9600E-03</td>
<td>-8.9692E-05</td>
<td>-8.5144E-05</td>
<td>94.9</td>
</tr>
<tr>
<td>38</td>
<td>1.8450E-02</td>
<td>1.7908E-02</td>
<td>-5.4262E-04</td>
<td>-5.3957E-04</td>
<td>99.4</td>
</tr>
<tr>
<td>47</td>
<td>1.1258E-02</td>
<td>1.0927E-02</td>
<td>-3.3109E-04</td>
<td>-3.2620E-04</td>
<td>98.5</td>
</tr>
<tr>
<td>48</td>
<td>1.8450E-02</td>
<td>1.7908E-02</td>
<td>-5.4262E-04</td>
<td>-5.3957E-04</td>
<td>99.4</td>
</tr>
<tr>
<td>57</td>
<td>4.2639E-03</td>
<td>3.9443E-03</td>
<td>-1.1952E-04</td>
<td>-1.1410E-04</td>
<td>95.5</td>
</tr>
<tr>
<td>60</td>
<td>2.5084E-02</td>
<td>2.4366E-02</td>
<td>-7.3773E-04</td>
<td>-7.3688E-04</td>
<td>99.9</td>
</tr>
<tr>
<td>61</td>
<td>2.6942E-02</td>
<td>2.6150E-02</td>
<td>-7.9237E-04</td>
<td>-7.9223E-04</td>
<td>100.0</td>
</tr>
</tbody>
</table>
The design sensitivity results of Table 8 seem to substantiate the fact that nodes near a clamped edge give a somewhat less accurate prediction of the design sensitivity. However, these results are still considered good, since there is less than 15\% deviation from the approximate finite difference result. Since the finite difference result is only an approximation, it is reasonable to say that the design sensitivity prediction is good everywhere along the plate.

A number of elements are selected to check the design sensitivity for von Mises' stress in Eq. (80). Before this can be evaluated, the nodal force of the adjoint load on the right side of Eq. (79) has to be calculated so that the strain $\varepsilon_{ij}^{(4)}(\lambda^{(9)})$ is known for each constraint element. This is accomplished by a restart of the original IFAD model with each adjoint load for each element being a separate loading case. Von Mises' stress results are shown in Table 9 for a design perturbation of $\delta t = 0.001t$.

Recall that the design sensitivity and the constraint functional was calculated using an integration technique where the stresses were calculated at the midside nodes of the triangular element. This integration technique was used so as to get the best design sensitivity results as possible for the finite elements. This is the technique used because the element is a hybrid element [4]. The results in Table 9 are excellent when this technique is used. These results further substantiate the fact that as design perturbation approaches zero, $\psi/\Delta\psi$ approaches one.
Table 9. Plate Design Sensitivity Check for von Mises' Stress

<table>
<thead>
<tr>
<th>El. No.</th>
<th>$\psi_g(t)$</th>
<th>$\psi_g(t+\delta t)$</th>
<th>$\Delta \psi_g$</th>
<th>$\psi'_g$</th>
<th>$\psi'_g \times 100/\Delta \psi_g$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>172.1862</td>
<td>171.8423</td>
<td>-0.3439</td>
<td>-0.3445</td>
<td>100.17</td>
</tr>
<tr>
<td>2</td>
<td>752.7383</td>
<td>751.2351</td>
<td>-1.5032</td>
<td>-1.5049</td>
<td>100.11</td>
</tr>
<tr>
<td>3</td>
<td>172.1861</td>
<td>171.8423</td>
<td>-3.4386</td>
<td>-3.4446</td>
<td>100.17</td>
</tr>
<tr>
<td>4</td>
<td>752.7383</td>
<td>751.2351</td>
<td>-1.5032</td>
<td>-5.0495</td>
<td>100.11</td>
</tr>
<tr>
<td>10</td>
<td>1442.8840</td>
<td>1140.0025</td>
<td>-2.8815</td>
<td>-2.8841</td>
<td>100.09</td>
</tr>
<tr>
<td>17</td>
<td>2457.3522</td>
<td>2452.4447</td>
<td>-4.9074</td>
<td>-4.9125</td>
<td>100.10</td>
</tr>
<tr>
<td>25</td>
<td>1149.0097</td>
<td>1146.7151</td>
<td>-2.2947</td>
<td>-2.2970</td>
<td>100.11</td>
</tr>
<tr>
<td>26</td>
<td>1331.9565</td>
<td>1329.2966</td>
<td>-2.6600</td>
<td>-2.6624</td>
<td>100.10</td>
</tr>
<tr>
<td>27</td>
<td>1149.0097</td>
<td>1146.7151</td>
<td>-2.2946</td>
<td>-2.2970</td>
<td>100.11</td>
</tr>
<tr>
<td>28</td>
<td>1331.9565</td>
<td>1329.2966</td>
<td>-2.6600</td>
<td>-2.6624</td>
<td>100.09</td>
</tr>
<tr>
<td>34</td>
<td>973.8975</td>
<td>971.9526</td>
<td>-1.9450</td>
<td>-1.9465</td>
<td>100.08</td>
</tr>
<tr>
<td>44</td>
<td>1442.8840</td>
<td>1440.0025</td>
<td>-2.8815</td>
<td>-2.8841</td>
<td>100.09</td>
</tr>
<tr>
<td>49</td>
<td>1284.1558</td>
<td>1281.5913</td>
<td>-2.5645</td>
<td>-2.5667</td>
<td>100.08</td>
</tr>
<tr>
<td>50</td>
<td>1278.5066</td>
<td>1275.9534</td>
<td>-2.5532</td>
<td>-2.5554</td>
<td>100.09</td>
</tr>
<tr>
<td>51</td>
<td>1274.1558</td>
<td>1281.5913</td>
<td>-2.5645</td>
<td>-2.5667</td>
<td>100.08</td>
</tr>
<tr>
<td>52</td>
<td>1278.5066</td>
<td>1275.9531</td>
<td>-2.5532</td>
<td>-2.5554</td>
<td>100.09</td>
</tr>
<tr>
<td>57</td>
<td>1110.0286</td>
<td>1107.8118</td>
<td>-2.2168</td>
<td>-2.2188</td>
<td>100.09</td>
</tr>
<tr>
<td>68</td>
<td>973.8975</td>
<td>971.9526</td>
<td>-1.9449</td>
<td>-1.9465</td>
<td>100.08</td>
</tr>
<tr>
<td>73</td>
<td>1470.1247</td>
<td>1417.2887</td>
<td>-2.8360</td>
<td>-2.8384</td>
<td>100.08</td>
</tr>
<tr>
<td>74</td>
<td>1629.1926</td>
<td>1625.9391</td>
<td>-3.2536</td>
<td>-3.2563</td>
<td>100.08</td>
</tr>
<tr>
<td>75</td>
<td>1420.1247</td>
<td>1417.2887</td>
<td>-2.8360</td>
<td>-2.8384</td>
<td>100.08</td>
</tr>
<tr>
<td>76</td>
<td>1629.1926</td>
<td>1625.9391</td>
<td>-3.2536</td>
<td>-3.2561</td>
<td>100.08</td>
</tr>
<tr>
<td>83</td>
<td>2457.3522</td>
<td>2452.4447</td>
<td>-4.9074</td>
<td>-4.9125</td>
<td>100.10</td>
</tr>
<tr>
<td>91</td>
<td>1110.0286</td>
<td>1107.8118</td>
<td>-2.2168</td>
<td>-2.2188</td>
<td>100.09</td>
</tr>
<tr>
<td>97</td>
<td>1884.4503</td>
<td>1880.6870</td>
<td>-3.7633</td>
<td>-3.7660</td>
<td>100.07</td>
</tr>
<tr>
<td>98</td>
<td>2010.1163</td>
<td>2006.1021</td>
<td>-4.0143</td>
<td>-4.0170</td>
<td>100.07</td>
</tr>
<tr>
<td>99</td>
<td>1884.4503</td>
<td>1880.6870</td>
<td>-3.7633</td>
<td>-3.7660</td>
<td>100.07</td>
</tr>
<tr>
<td>100</td>
<td>2010.1663</td>
<td>2006.1021</td>
<td>-4.0143</td>
<td>-4.0170</td>
<td>100.07</td>
</tr>
</tbody>
</table>
3.4 Built-Up Structure

A built-up structure that uses both beams and plates is shown in Fig. 10. Since it is symmetric along two planes, only a quarter is modeled. The built-up structure is clamped on two edges, with symmetric boundary conditions applied along the other two edges. The plates and the beams are considered to be welded. A uniform pressure $f(x) = -1.5 \text{ lb/in.}^2$ is applied on the top surface of the plates. The model contains 100 IFAD triangular thin shell elements of type 1601, with only the bending terms active, and 20 IFAD beam elements of type 0501. There are 61 nodal points and 140 degrees-of-freedom. There are three design variables, beam width, beam height, and plate thickness. The material constants, Young's modulus and Poisson's ratio, for both the beams and the plates are $E = 30.5 \times 10^6$ psi and $\nu = 0.3$, respectively. Self weight is neglected and the initial design variables are $b = 0.5$ in., $h = 0.75$ in., and $t = 0.4$ in.

Compliance sensitivity results are shown in Table 10, where $\Delta \psi_{11} = \psi_{10}(u + \delta u) - \psi_{10}(u)$ and $\psi_{11}^1$ is the predicted design sensitivity, using Eq. (101) with a 1% design perturbation for all the design variables.

Several discrete points in Fig. 10 are selected to check the accuracy of design sensitivity of the displacement functional in Eq. (104). To calculate this equation, the strain $\varepsilon_{ij}^1(\lambda)^{(12)}$ due to the adjoint load is obtained by doing a restart of the finite element analysis. A unit point load in the $-z$ direction for each selected point
Figure 10. Built-up Structure Finite Element Model
Table 10. Built-up Structure Design Sensitivity
Check for Compliance with $\delta h = 0.01h$
$\delta b = 0.01b$, and $\delta t = 0.01t$

<table>
<thead>
<tr>
<th>$\psi_{11}(u)$</th>
<th>$\psi_{11}(u+\delta u)$</th>
<th>$\Delta\psi_{11}$</th>
<th>$\psi_{11}'$</th>
<th>$\psi_{11}' \times 100/\Delta\psi_{11}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.7200</td>
<td>3.6017</td>
<td>-0.1183</td>
<td>-0.1189</td>
<td>100.5</td>
</tr>
</tbody>
</table>

is specified as a separate load case. The design sensitivity results
for these selected points are shown in Table 11, where a 1% perturbation
for each design variable is used.

Table 11. Built-up Structure Design Sensitivity
Check for Displacement with $\delta h = 0.01h$,
$\delta b = 0.01b$, and $\delta t = 0.01t$

<table>
<thead>
<tr>
<th>Node</th>
<th>$\psi_{12}(u)$</th>
<th>$\psi_{12}(u+\delta u)$</th>
<th>$\Delta\psi_{12}$</th>
<th>$\psi_{12}'$</th>
<th>$\psi_{12}' \times 100/\Delta\psi_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>4.768E-04</td>
<td>4.616E-04</td>
<td>-1.520E-05</td>
<td>-1.453E-05</td>
<td>95.6</td>
</tr>
<tr>
<td>15</td>
<td>2.294E-06</td>
<td>2.222E-03</td>
<td>-7.283E-05</td>
<td>-6.750E-05</td>
<td>92.7</td>
</tr>
<tr>
<td>17</td>
<td>3.064E-03</td>
<td>2.966E-03</td>
<td>-9.723E-05</td>
<td>-9.152E-05</td>
<td>94.1</td>
</tr>
<tr>
<td>25</td>
<td>4.125E-03</td>
<td>3.994E-03</td>
<td>-1.313E-04</td>
<td>-1.255E-04</td>
<td>95.6</td>
</tr>
<tr>
<td>27</td>
<td>5.421E-03</td>
<td>8.153E-03</td>
<td>-2.680E-04</td>
<td>-2.621E-04</td>
<td>97.8</td>
</tr>
<tr>
<td>35</td>
<td>2.294E-03</td>
<td>2.221E-03</td>
<td>-7.283E-05</td>
<td>-6.750E-05</td>
<td>92.7</td>
</tr>
<tr>
<td>37</td>
<td>1.095E-02</td>
<td>1.060E-02</td>
<td>-3.484E-04</td>
<td>-3.448E-04</td>
<td>99.0</td>
</tr>
<tr>
<td>39</td>
<td>1.485E-02</td>
<td>1.438E-02</td>
<td>-4.724E-04</td>
<td>-4.737E-04</td>
<td>100.3</td>
</tr>
<tr>
<td>47</td>
<td>8.421E-03</td>
<td>8.153E-03</td>
<td>-2.680E-04</td>
<td>-2.621E-04</td>
<td>97.8</td>
</tr>
<tr>
<td>49</td>
<td>1.755E-02</td>
<td>1.699E-02</td>
<td>-5.580E-04</td>
<td>-5.621E-04</td>
<td>100.7</td>
</tr>
<tr>
<td>55</td>
<td>1.955E-02</td>
<td>1.892E-02</td>
<td>-6.215E-04</td>
<td>-6.296E-04</td>
<td>101.3</td>
</tr>
<tr>
<td>57</td>
<td>3.064E-03</td>
<td>2.966E-03</td>
<td>-9.723E-05</td>
<td>-9.152E-05</td>
<td>94.1</td>
</tr>
<tr>
<td>59</td>
<td>1.485E-02</td>
<td>1.438E-02</td>
<td>-4.724E-04</td>
<td>-4.737E-04</td>
<td>101.3</td>
</tr>
<tr>
<td>61</td>
<td>2.025E-02</td>
<td>1.961E-02</td>
<td>-6.439E-04</td>
<td>-6.535E-04</td>
<td>101.5</td>
</tr>
</tbody>
</table>

Both the compliance and displacement design sensitivity results
show good correlation with the approximated finite difference. This
indicates that the method of design sensitivity is a good method for
predicting the response of a built-up structure for these constraint functionals. This is substantiated by the fact that correlation between the design sensitivities and the finite difference approximations for the built-up structure in Tables 10 and 11 are not that different than those in Tables 7 and 8 where plate bending results are shown.

To check the stress constraint sensitivity of Eq. (98), the equivalent nodal force of the adjoint load on the right of Eq. (97) has to be calculated so that $e^{ij}(\lambda)(10)$ is known for each constrained element. A restart of the original IFAD built-up structure model is made, with each adjoint load for a finite element being a separate loading case. The design sensitivity results for the von Mises' stress functional are given in Table 12, with design perturbation $\delta t = 0.001t$.

The design sensitivity results compared to the finite difference approximations fluctuate more than would be expected, considering the excellent correlation of the von Mises' stress design sensitivities for the bending plate in Table 9. It doesn't appear that the problem is caused by the design perturbation being too small, because the finite difference approximation is no smaller than three significant digits of the actual constraint functional. It is possible that the nonlinear response of the built-up structure is partially the cause of the inconsistencies in the design sensitivities, but not the whole problem, since the design perturbation is quite small.

It is most likely that the beam contribution to the built-up structure is the major influence. The beam seems to be more influenced
by nonlinearities, as shown in Tables 4, 5, and 6. It is most probable that if a more complex beam element, such as a cubic beam, was used, design sensitivity results would improve. Unfortunately the IFAD code does not currently support this type of beam, so this cannot be verified in this study.

Table 12. Built-up Structure Design Sensitivity Check for von Mises' Stress with $\delta t = 0.001t$, $\delta b = 0.001b$, and $\delta h = 0.001h$

<table>
<thead>
<tr>
<th>Element</th>
<th>$\psi_{10}(t)$</th>
<th>$\psi_{10}(t+\delta t)$</th>
<th>$\Delta \psi_{10}$</th>
<th>$\psi'_{10}$</th>
<th>$\psi'<em>{10} \times 100/\Delta \psi</em>{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>127.8318</td>
<td>127.5490</td>
<td>-0.2828</td>
<td>-0.3339</td>
<td>118.1</td>
</tr>
<tr>
<td>2</td>
<td>549.7750</td>
<td>548.5339</td>
<td>-1.2411</td>
<td>-1.2230</td>
<td>98.5</td>
</tr>
<tr>
<td>3</td>
<td>127.8318</td>
<td>127.5490</td>
<td>-0.2828</td>
<td>-0.3339</td>
<td>118.1</td>
</tr>
<tr>
<td>4</td>
<td>549.7750</td>
<td>548.5339</td>
<td>-1.2411</td>
<td>-1.2230</td>
<td>98.5</td>
</tr>
<tr>
<td>10</td>
<td>1074.7414</td>
<td>1072.3202</td>
<td>-2.4212</td>
<td>-2.7653</td>
<td>114.2</td>
</tr>
<tr>
<td>17</td>
<td>1828.5601</td>
<td>1824.4411</td>
<td>-4.1190</td>
<td>-4.2965</td>
<td>104.3</td>
</tr>
<tr>
<td>25</td>
<td>875.4135</td>
<td>873.4610</td>
<td>-1.9525</td>
<td>-2.3201</td>
<td>118.1</td>
</tr>
<tr>
<td>26</td>
<td>985.6012</td>
<td>983.3797</td>
<td>-2.2215</td>
<td>-2.6663</td>
<td>120.0</td>
</tr>
<tr>
<td>27</td>
<td>875.4135</td>
<td>873.4610</td>
<td>-1.9525</td>
<td>-2.3201</td>
<td>118.1</td>
</tr>
<tr>
<td>28</td>
<td>985.6012</td>
<td>983.3747</td>
<td>-2.2215</td>
<td>-2.6663</td>
<td>120.0</td>
</tr>
<tr>
<td>34</td>
<td>735.6485</td>
<td>734.0066</td>
<td>-1.6419</td>
<td>-2.2807</td>
<td>138.9</td>
</tr>
<tr>
<td>44</td>
<td>1074.7414</td>
<td>1072.3202</td>
<td>-2.4212</td>
<td>-2.7653</td>
<td>114.2</td>
</tr>
<tr>
<td>50</td>
<td>962.8944</td>
<td>960.7381</td>
<td>-2.1563</td>
<td>-2.4538</td>
<td>113.8</td>
</tr>
<tr>
<td>51</td>
<td>944.5642</td>
<td>942.4288</td>
<td>-2.1354</td>
<td>-2.7966</td>
<td>131.0</td>
</tr>
<tr>
<td>57</td>
<td>804.5913</td>
<td>802.7656</td>
<td>-1.8257</td>
<td>-2.4206</td>
<td>122.7</td>
</tr>
<tr>
<td>68</td>
<td>735.6485</td>
<td>734.0066</td>
<td>-1.6420</td>
<td>-2.2807</td>
<td>138.9</td>
</tr>
<tr>
<td>73</td>
<td>1074.1010</td>
<td>1071.6982</td>
<td>-2.4028</td>
<td>-2.7000</td>
<td>112.6</td>
</tr>
<tr>
<td>74</td>
<td>1243.6633</td>
<td>1240.8853</td>
<td>-2.7780</td>
<td>-3.2605</td>
<td>117.4</td>
</tr>
<tr>
<td>75</td>
<td>1074.1010</td>
<td>1071.6982</td>
<td>-2.4028</td>
<td>-2.7000</td>
<td>112.6</td>
</tr>
<tr>
<td>76</td>
<td>1243.6633</td>
<td>1240.8853</td>
<td>-2.7780</td>
<td>-3.2605</td>
<td>117.4</td>
</tr>
<tr>
<td>83</td>
<td>1828.5601</td>
<td>1824.4411</td>
<td>-4.1190</td>
<td>-4.2965</td>
<td>104.3</td>
</tr>
<tr>
<td>91</td>
<td>804.5913</td>
<td>802.7656</td>
<td>-1.8257</td>
<td>-2.2406</td>
<td>122.7</td>
</tr>
<tr>
<td>97</td>
<td>1401.1703</td>
<td>1398.0144</td>
<td>-3.1559</td>
<td>-3.2171</td>
<td>101.9</td>
</tr>
<tr>
<td>98</td>
<td>1517.4522</td>
<td>1514.0511</td>
<td>-3.4011</td>
<td>-3.6964</td>
<td>108.7</td>
</tr>
<tr>
<td>99</td>
<td>1401.1703</td>
<td>1398.0144</td>
<td>-3.1559</td>
<td>-3.2171</td>
<td>101.9</td>
</tr>
<tr>
<td>100</td>
<td>1517.4522</td>
<td>1514.0511</td>
<td>-3.4011</td>
<td>-3.6964</td>
<td>108.7</td>
</tr>
</tbody>
</table>
CHAPTER IV
CONCLUSIONS

The results of this study indicate that it is feasible to implement the theoretical design sensitivity analysis of Ref. 1 with an existing finite element code. Calculations of the design sensitivities can be accomplished outside of the finite element code using only the postprocessing data. In addition, the results show that accurate design sensitivity can be predicted without the uncertainty of numerical accuracy associated with the selection of a finite difference perturbation. However, results also indicate that the integration technique used in the calculation of the design sensitivity and the knowledge of the exact finite element shape functions used for each finite element in the finite element code is important in getting the most accurate design sensitivities possible.

Results of the built-up structure indicate that design sensitivities of the built-up structure cannot be any more accurate than the accuracy of the individual components. Care must be taken to use the best integration techniques and the same shape functions as the finite element analysis for the individual components, if at all possible.
REFERENCES


APPENDIX

DESIGN SENSITIVITY IFAD PROGRAM
PROGRAM SENSIT
CP***********************************************************************
CP* SENSIT: THE MAIN PROGRAM FOR CALCULATING DESIGN SENSITIVITY
CP* CP***********************************************************************
CP* DESCRIPTION:
CP* CP***********************************************************************
CP* 'SENSIT' IS THE MAIN PROGRAM FOR THE DESIGN SENSITIVITY
CP* VECTOR CALCULATION. IT CHECKS THE ACCURACY OF THE
CP* SENSITIVITY TO THE FINITE DIFFERENCE METHOD IF THE
CP* TWO FINITE ELEMENT ANALYSES ( THE ORIGINAL ANALYSIS
CP* AND THE FERIURBED ANALYSIS ) ARE ALREADY CREATED.
CP* MAX. OF 500 ELEMENTS, IF MORE ELEMENTS ARE NEEDED THE
CP* SVECTR.MON COMMON BLOCK FILE NEEDS TO CHANGED.
CP* CP***********************************************************************
CP***********************************************************************
INCLUDE 'CAEGSDR.INCJ IMPLIC.SPC'
INCLUDE 'CAEGSDR.INCJ INTL.MON'
INCLUDE 'CAEGSDR.INCJ SVECTR.MON'
C EQUIVALENCE (NUAT(46),NLMH),(NUAT(1H1),IPNAME)
C DIMENSION PS18(2),IPNAME(2),IPNAME1(2),IPNAME2(2)
CHARACTER YESNO*1
C
NT = 0
ISAC = 0
LCS = 1
NLC = 1
C ASK FOR FINITE ELEMENT MODEL FILE NAME
C PRINT *,
C PRINT *,'ENTER THE ORIGINAL PROBLEM NAME (1-8 CHARs)',
READ(5,1009) IPNAME
C C ASK FOR CONSTRAINT TYPE
C PRINT *,
C PRINT *,'ENTER CONSTRAINT TYPE: 1 = COMPLIANCE',
PRINT *, 2 = DISPLACEMENT',
PRINT *, 3 = STRESS',
READ(5,1006) ICT
IF(ICT.NE.3) GO TO 20
PRINT *,
PRINT *,'ENTER STRESS TYPE: 1 = PRINCIPAL',
PRINT *, 2 = VON MISS',
PRINT *, 3 = BEAM ALLOWABLE',
READ(5,1006) IST
C 20 PRINT *,'CALCULATING ADJOINT LOADS (Y/N) ?'
READ(5,1000) YESNO
IF(YESNO.EQ.'Y') ISAC = 1
IF(ISAC.EQ.1) GO TO 40
C C ASK FOR THE FERIURBED FINITE ELEMENT NAME
C PRINT *,
PRINT *,'ENTER THE PERTURBED PROBLEM NAME (1-8 CHARS)' READ(5,1009) IFNAME2
C
C OPEN AN OUTPUT FILE
C OPEN(UNIT=10,NAME='RESULT.DAT',TYPE='NLW')
WRITE(10,1007) ICT
IF(ICT.NE.3) GO TO 40
WRITE(10,1008) IST
C
C GET LOAD CASE START AND ENN NOS.
C 40 PRINT *,'ENTER NO. OF LOAD CASES TO BE PROCESSED'
READ(5,1006) NLC
PRINT *,'ENTER FIRST LOAD CASE NO.,'
READ(5,1006) LCS
LCS = LCS - 1
C
C GET CONSTRAINED ELEMENTS FOR STRESS CALCULATION
C IF(ICT.NE.3) GO TO 90
PRINT *,'',
IF(ISAC.EQ.1) GO TO 60
PRINT 2000, NLC
GO TO 70
C
C GET ELEMENTS FOR ADJOIN LOAD CALCULATION
C 50 PRINT 2001, NLC
70 DO 80 NE=1,NLC
80 READ(5,1006) ICE(NE)
90 DO 800 NE=1,NLC
IF(ISAC.EQ.1) GO TO 100
WRITE(10,1010) ICE(NE)
C
C GET SENSITIVITY VECTOR
C 100 IF(NT.EQ.1) GO TO 110
IFNAME(1) = IFNAME1(1)
IFNAME(2) = IFNAME1(2)
GO TO 120
110 IFNAME(1) = IFNAME2(1)
IFNAME(2) = IFNAME2(2)
120 CALL GETSEN(PSID,NT,NELM,IPNAME)
IF(ISAC.EQ.1) GO TO 410
IF(NT.GT.1) GO TO 200
GO TO 100
C
C CALC. CHANGE IN PLATE THICKNESS, CHANGE IN BEAM WIDTH AND DEPTH
C 200 N1 = 0
DM = DABS(TM(2)-TM(1))
DG = DABS(BW(2)-BW(1))
DH = DABS(BH(2)-BH(1))
TB = DABS(FB(2)-FB(1))
C
C CALCULATE THE PERCENT ACCURACY
C DPSIBDP = 0.0
DO 300 I=1,NELM
DFSIBBB = DFSIBBB+DFSIT(I)*HT+DFSISH(I)*DH
+DFSISH(I)*DM+DFSITM(I)*TM

CONTINUE
PNUM = DFSIBBB*100
PDEN = PSIS(2)-PSIS(1)
IF(PDEN.EQ.0) GO TO 801
PACCUR = PNUM/PDEN
C
WRITE(10,*),' ','
WRITE(10,1004) DT,DM,DI
WRITE(10,1005) PDEN
WRITE(10,1002) DFSISH
WRITE(10,1003) PACCUR
CONTINUE
C
GO TO 900
C
PRINT *, 'DFSIS(W)*DI = 0'
C
CLOSE(10)
CONTINUE
C
1000 FORMAT(A)
1001 FORMAT(F8.5)
1002 FORMAT(1X,'DFSIS(W)*DELTAB=',E16.8)
1003 FORMAT(1X,'PERCENT ACCURACY=',F16.8)
1004 FORMAT(1X,'CHANGE IN MEMBRANE THICKNESS =',F8.5,
* IN BEAM WIDTH =',F8.5,
* IN MEM. DEPTH =',F8.5,
* IN INERT PLATE THICKNESS =',F8.5)
1005 FORMAT(1X,'PSI(RDB) - PSI(W) = ',F16.8)
1006 FORMAT(I4)
1007 FORMAT(1X,'**CONTRAINT TYPE =',I4)
1008 FORMAT(1X,'**STRESS TYPE =',I4)
1009 FORMAT(2A4)
1010 FORMAT(1X,'CONTRAINT ELEMENT IS',I4)
2000 FORMAT(1X,'ENTER',I4,' CONTRAINT ELEMENTS, FOLLOW EACH BY
* A RETURN'
)
2001 FORMAT(1X,'ENTER',I4,' ELEMENTS THAT ARE TO HAVE AN AJOINT
* LOAD CALCULATED, FOLLOW EACH BY A RETURN.
')
C
END
SUBROUNINE AL16(X,Y,C,AL,THK,IT)

CP*****************************************************
CP
CP* AL16: ADJOINT LOAD CALCULATION FOR TRIANGULAR ELEMENT 1601
CP*
CP*****************************************************
CP*
CP* DESCRIPTION:
CP*
CP* 'AL16' CALCULATES THE ADJOINT LOADS FOR THE TRIANGULAR
CP*
CP* PLATE BENDING ELEMENT. AL = [C]@[L]@MP WHERE
CP*
CP* [C] IS THE DERIVATIVE OF THE STRESS FUNCTION VECTOR
CP*
CP* TIMES THE ELASTICITY MATRIX. SINCE IFAL CALUCILATES
CP*
CP* STRESS RESULTS FIRST, [AL] MUST BE MULTIPLIED BY
CP*
CP* PLATE THICKNESS DIVIDE BY 2.
CP*
CP* [BJ IS A 3x9 MATRIX IN COLUMNS 4,5 & 6 OF [AW]. MP
CP*
CP* IS THE CHARACTERISTIC FUNCTION THAT IS 1/AREA OF THE
CP*
CP* ELEMENT THAT IS CONSTRAINED AND ZERO FOR ALL THE OTHER
CP*
CP* ELEMENTS.
CP*
CP*****************************************************
CP*
CP* X THE LOCAL ELEMENT X COORDINATE
CP*
CP* Y THE LOCAL ELEMENT Y COORDINATE
CP*
CP* C MATRIX [C] = [DG][NJ][E][Y]/2  3x3 MATRIX
CP*
CP* AL ADJOINT LOAD VECTOR
CP*
CP* TB TRANSFORMATION MATRIX
CP*
CP* THK MATERIAL THICKNESS
CP*
CP* IT MIDSIDE NODE COLUMN LOCATOR
CP*
CP*****************************************************
C
C INCLUDE 'LAEGSDR.INCJ IMPL.IUC.SPC'
C INCLUDE 'EAEGSINC.INCJ (NIN,HUM)'
C
C EQUIVALENCE (NMA(14),IFR)
C
C DIMENSION X(3),Y(3),GPTS(3,3),XL(6),YL(6),W(18,7),LG(3),
C    * F(9),AL(18),K(6,3),TH(6,6),RP(6,3)
C
C DATA GPTS/0.000,0.500,0.500, 0.000,0.000,0.000, 0.500,0.500,0.000/
C
C*** INITIALIZE VARIABLES
C
C DO 10 I=1,Y
C 10 F(I) = 0.000
C DO 12 I=1,THK
C 12 AL(18) = 0.000
C DO 14 I=1,6
C 14 J=1,3
C     R(I,J) = 0.000
C CONTINUE
C
C AREA = EUTRIN(X,Y)
C XMP = 1.000/AREA
C
C*** GET LOCAL X AND Y COORDINATES
C
C CALL MOVESP(XL,X,3*IFR)
C CALL MOVESP(YL,Y,3*IFR)
C CALL SF1501(XL,YL,GPTS(1,1),W,THK)
DO 50 M=1,4
GASH = 0.0DO
DO 40 J=1,3
   GASH = (GASH+L(J))#W(M,J+3)
40   CONTINUE
   F(M) = F(M)+GASH*XMP
50   CONTINUE
C
C** ROTATE ADJOINLOAD VECTOR TO GLOBAL COORDINATE SYSTEM
C
N = 1
DO 60 MM=1,3
   R(3,MM) = F(N)
   R(4,MM) = F(N+1)
   R(5,MM) = F(N+2)
   N = N+3
60   CONTINUE
CALL UMXAB(1H+R+RP,6,3,6)
N1 = 0
DO 80 N=1,3
   DO 70 M1=1,6
      AL(M1+N1) = RP(M1+N)
70   CONTINUE
   N1 = N1+6
80   CONTINUE
DO 90 N2 = 1,18
   AL(N2) = AL(N2)#THK/2.0DO
90   CONTINUE
C
C
RETURN
END
FUNCTION AREA1(X,Y)
CP******************************************************************************
CP*
CP* AREA1: CALCULATES THE AREA OF A STRAIGHT SIDED FOUR OR EIGHT NODE ELEMENT.
CP*
CP******************************************************************************
CP*
CP* X  GLOBAL X COORDINATES
CP* Y  GLOBAL Y COORDINATES
CP*
CP******************************************************************************
C
C INCLUDE 'CAGDISR.INC' IMPLIC.SPC'
C INCLUDE 'CAGDISR.INC' ACIPN.MON'
C INCLUDE 'CAGDISR.INC' FIELDS.MON'
C
C DIMENSION X(4),X1(3),X2(3),Y(4),Y1(3),Y2(3),Z(4),BUF(100)
C
C DATA IREF/1/
C
C***** CALCULATES THE AREA OF A QUADRILATERAL
C
C IF(NUNFE.EQ.0)  GO TO 20
C
C FOUR NODDED ELEMENT
C
C X1(1) = X(1)
X1(2) = X(2)
X1(3) = X(4)
Y1(1) = Y(1)
Y1(2) = Y(2)
Y1(3) = Y(4)
A1 = EU(RIA(X1,Y1))
X2(1) = X(2)
X2(2) = X(3)
X2(3) = X(4)
Y2(1) = Y(2)
Y2(2) = Y(3)
Y2(3) = Y(4)
A2 = EU(RIA(X2,Y2))
GO TO 30
C
C EIGHT NODDED ELEMENT WITH STRAIGHT SIDES
C
C 20
C X1(1) = X(1)
X1(2) = X(3)
X1(3) = X(5)
Y1(1) = Y(1)
Y1(2) = Y(3)
Y1(3) = Y(5)
A1 = EU(RIA(X1,Y1))
X2(1) = X(1)
X2(2) = X(3)
X2(3) = X(5)
Y2(1) = Y(1)
Y2(2) = Y(3)
Y2(3) = Y(5)
A2 = EU(RIA(X2,Y2))
C
C 30 AREA1 = A1+A2
C     GO TO 900
C 907 PRINT 877, IEK
C 877 FORMAT(1X,'ACCEL RETURNED WITH ERROR', 14)
 900 CONTINUE
C     RETURN
END
SUBROUTINE COMB(PSIB, NT, NELM)

CP$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
CP*
CP* COMB: BRANCHES TO THE APPROPRIATE ELEMENT TYPE TO CALCULATE
CP*
CP$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
CP*
CP* DESCRIPTION:
CP*
CP* 'COMP' BRANCHES TO THE APPROPRIATE ELEMENT TYPE TO
CP* CALCULATE THE COMPLIANCE AND THE COMPLIANCE SENSIT-
CP* IVITY VECTOR.
CP*
CP$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
CP
C INCLUDE 'AEQSDR,INCJ IMPLC.SPC'
C INCLUDE 'AEQSDR,INCJ ACCIPN,MON'
C INCLUDE 'AEQSDR,INCJ CNTL,MON'
C INCLUDE 'AEQSDR,INCJ ELEDES,MON'
C INCLUDE 'AEQSDR,INCJ SVECTR,MON'
C COMMON/LCSDES/DLCS(90)
C
C EQUIVALENCE (NUAT(97), IDBS), (NUAT(98), IHL)
C
C DIMENSION MAIN(50), PSIBN(500), PSIB(2), LFI PR(6), PSIBT(500)
C
C DATA IREF/1/
C
C FSIB14 = 0.0
C FSIBCS = 0.0
C SDFSIT = 0.0
C SDFSIB = 0.0
C SDFSIB = 0.0
C SDFSTB = 0.0
C
C SET LOAD CASE NUMBER FOR COMPLIANCE
C
C L1 = LCS + NC
C
C
C SETUP POINTERS
C
C CALL ACCELEM(1, IPNELM, IDBS, 1, 0, IERR)
C IF(IERR.NE.0) GO TO 800
C CALL ACCFES(1, IFNYES, IDBL, 1, L1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 801
C CALL ACCND(1, IFNCND, IHL, 1, L1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 802
C CALL ACCCLS(1, IFNCLS, IDBL, 1, 0, IERR)
C IF(IERR.NE.0) GO TO 805
C CALL ACCMOD(1, IFNMOD, IHL, 1, 0, IERR)
C IF(IERR.NE.0) GO TO 806
C CALL ACCELC(1, IFNELC, IDBS, 1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 807
C CALL ACCMAT(1, IFNMAT, IDBS, 1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 808
C CALL ACCPR(1, IFNPRI, IDBS, 1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 809
C CALL ACCEEN(1, IPNHEN, IDPL, 1, L1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 810
C
C LOOP THROUGH THE ELEMENTS
DO 100 I=1,NELM
   IF(I.GT.1) GO TO 50
C GET INTERNAL LOAD CASE NUMBER
   CALL ACCLCS(2,IPNLCS,L1,2,DLCS,IERR)
   ILCN = DLCS(2)
C GET ELEMENT DESCRIPTIONS
   CALL ACCELm(2,IPNELL1,1,2,IED,IERR)
   IF(IERR.NE.0) GO TO 800
C BRANCH TO THE APPROPRIATE ELEMENT TYPE
   IF(ITYP.EQ.11) CALL COMP11(M,I,ILCN)
   IF(ITYP.EQ.5) CALL COMPOS(M,I,ILCN,PSIBN)
   IF(ITYP.EQ.16) CALL CP16(M,I,ILCN,PSIBTB)
C
   PSIB16 = $IB16+PSIBTB(I)
   PSIBC5 = $IBC5+$IBB(I)
   SDPSIT = $DPSIT+PSIB(I)
   SDPSIB = $DPSIB+$IBB(I)
   SDPSIH = $DPSIH+$IBI(I)
   SDPSIB = SDPSIB+$IBB(I)
   CONTINUE
C
   IF(N.T.GT.1) GO TO 720
   WRITE(10,859)
   DO 710 I=1,NELM
      WRITE(10,857) LIPS1(I),DFSP(I),LIPS(I),LIPS(1)
   710 WRITE(10,857) LIPS1(I),DFSP(I),LIPS(I),LIPS(1)
C
   IF(ITYP.EQ.11) CALL LPSI11(PSIB,ILCN,IFEF)
C   PSIB(N) = PSIB + PSIBC5 + PSIB16
C
   WRITE(10,858) PSIB(N)
   PRINT 859, PSIB(N)
C
   CALL ACCL(4,IPNELL1,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 300
   CALL ACCFES(4,IPNEFE,0,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 301
   CALL ACCCNU(4,IPCHNU,0,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 302
   CALL ACCLCS(4,IPNLCS,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 303
   CALL ACCMOD(4,IPNMOD,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 304
   CALL ACCEL(4,IPNELEC,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 305
   CALL ACCMAT(4,IPMAT,0,0,0,1ERR)
   IF(IERR.NE.0) GO TO 306
   CALL ACCEFPR(4,IPNEFP,0,0,0,1ERR)
IF (IERR .NE. 0) GO TO 809
CALL ACCEEN (4, IPHNL, 0, 0, 0, 0, IERR)
IF (IERR .NE. 0) GO TO 810
C
GO TO 820
C
WHITE ERROR MESSAGES TO THE SCREEN
C
800 PRINT 870, IERR
GO TO 820
801 PRINT 871, IERR
GO TO 820
802 PRINT 872, IERR
GO TO 820
805 PRINT 875, IERR
GO TO 820
806 PRINT 876, IERR
GO TO 820
807 PRINT 878, IERR
GO TO 820
808 PRINT 879, IERR
GO TO 820
809 PRINT 876, IERR
GO TO 820
810 PRINT 880, IERR
GO TO 820
C
820 CONTINUE
C
C
957 FORMAT (13, 4X, 4(E16.8, 4X))
958 FORMAT (1X, 'PSIB=', E16.8)
959 FORMAT (1X, '/1X', 'EN', 6X, 'SENSITIVITY 1', 7X, 'SENSITIVITY H', 7X, 'SENSITIVITY 18', 7X, 'SENSITIVITY H', 6X, 'SENSITIVITY 18')
961 FORMAT (1X, '/1X', 'TOTAL=', 4(E16.8, 4X))
962 FORMAT (1X, 'ELEMENT ', 14)
970 FORMAT (1X, 'ACCELM RETURNED WITH ERROR ', 14)
971 FORMAT (1X, 'ACCFES RETURNED WITH ERROR ', 14)
972 FORMAT (1X, 'ACCCND RETURNED WITH ERROR ', 14)
975 FORMAT (1X, 'ACCLCS RETURNED WITH ERROR ', 14)
976 FORMAT (1X, 'ACCEFR RETURNED WITH ERROR ', 14)
977 FORMAT (1X, 'ACCNOD RETURNED WITH ERROR ', 14)
978 FORMAT (1X, 'ACCELCC RETURNED WITH ERROR ', 14)
979 FORMAT (1X, 'ACCMAT RETURNED WITH ERROR ', 14)
980 FORMAT (1X, 'ACCEEN RETURNED WITH ERROR ', 14)
1004 FORMAT (14)
C
C
RETURN
END
SUBROUTINE COMPOS(NIT,ILCN,FSIBK)

C********************************************************************C
C* COMPOS: CALCULATES COMPLIANCE AND SENSITIVITY FOR A BEAM    C
C* DESCRIPTION:                                              C
C* 'COMPOS' CALCULATES THE COMPLIANCE AND THE DESIGN       C
C* SENSITIVITY FOR A 1-D BEAM IN BENDING, WITH AN             C
C* APPLIED ELEMENT FORCE IN #/IN. SELF WEIGHT IS              C
C* NEGLECTED. BEAM TORSION HAS BEEN ADDED.                   C
C* C********************************************************************C
C* NT      COUNTER FOR FINITE DIFFERENCE                     C
C* I       EXTERNAL ELEMENT NO. BEING PROCESSED             C
C* ILCN    INTERNAL LOAD CASE NO.                           C
C* NELM    TOTAL NO. OF ELEMENTS                            C
C* C********************************************************************C
C
C INCLUDE 'AEGSDR.INC' IMPLC.SFC'                                    C
C INCLUDE 'AEGSDR.INC' ACCIFN.MON'                                C
C INCLUDE 'AEGSDR.INC' CNIL.MON'                                 C
C INCLUDE 'AEGSDR.INC' ELEDES.MON'                               C
C INCLUDE 'AEGSDR.INC' SVECTR.MON'                               C
C
C EQUIVALENCE (NDAT(14),IF)
C
C DIMENSION GPLW(3),FSIBE(500),DSHPF(12),X(3),Y(3),Z(3),
C   * SHPF(12),SBUF(200),NAIN(50),CFMUK(A),EBUF(200),
C   * BUF(100),CD(2,6),WLN(3),C(6,2),CDL(2,6),C(3,3),
C   * TB(6,6),COOR(3,3)
C
C DATA GPLW/,7745966700, .7745966700/
C DATA WLN/,55555555600, .55555555600/
C DATA KT/3.141592653589793/ IF(TI,1) GO TO 50
C
C FSIBK(I) = 0.0
C DFSIBG = 0.0
C DFSIHG = 0.0
C IF(I.GT.1) GO TO 50
C
C GET APPLIED FORCE IN #/IN.
C
C PRINT *, '.'
C PRINT *, 'ENTER APPLIED FORCE IN #/IN. UNITS'
C READ(5,1001) AF
C
C GET APPLIED DISTRIBUTED MOMENT
C
C PRINT *, '.'
C PRINT *, 'ENTER APPLIED DISTRIBUTED MOMENT'
C READ(5,1001) AM
C
C GET AREA MOMENT OF INERTIA ABOUT Y-AXIS
C
C CALL ACCEFR(2,IPNEFR,IPTAB,0,SBUF,LFN,IERK)
IF(IERR.NE.0) go to 809
YI = BUF(5)
H = 2.0*BUF(Y)
B = 2.0*BUF(10)
BW(MF) = B
BH(MF) = H

GET WEIGHT DENSITY AND MODULUS OF ELASTICITY

GAMMA = 0.0
E = BUF(5)
V = BUF(7)
G = E/(2.0*(1.0+V))

GET DISPLACEMENTS AT ELEMENT NODES

DO 150 J=1,NUME
CALL ACCCEL(2,1PHEL,C,KINI,IREF,BUH,LEN,IREF)
IF(IERR.NE.0) go to 802
DO 150 K=1,NUMF
CD(J,K) = CBUF(K)
150 CONTINUE

EVALUATE DISPLS. AND CURVATURE AT THE GAUSS POINT USING
SHAPE FUNCTIONS - ONE PT. FOR CURV., THREE PT. FOR DISPL

GET X, Y, AND Z OF ELEMENT NODES

CALL ACCEL(2,1PHEL,C,KINI,IREF,BUH,LEN,IREF)
IF(IERR.NE.0) go to 807
M = 1
DO 200 J=1,9,3
K = J+1
L = J+2
X(M) = BUF(J)
Y(M) = BUF(K)
Z(M) = BUF(L)
M = M+1
200 CONTINUE
DO 210 J=1,3
COOR(1,J) = X(J)
COOR(2,J) = Y(J)
COOR(3,J) = Z(J)
210 CONTINUE

FORM THE ELEMENT LOCAL COORDINATE SYSTEM

IN3 = IN3NN(3)
CALL EUROM(IN3,H,1A,COOR,T,IERR)
CALL ZERO3P(IB,3&IF)
DO 220 J=1,3
DO 220 K=1,3
TB(J,K) = T(J,K)
TB(J+3,K+3) = T(J,K)
220 CONTINUE
CALL UMXAB(TB,3,C,6,2,6)
DO 230 J=1,2
DO 230 K=1,6
CDL(J,K) = C(K,J)

CALCULATE ELEMENT LENGTH

DX = X(2)-X(1)
DY = Y(2)-Y(1)
DZ = Z(2)-Z(1)
EL = DSGRT(DX*DX+DY*DY+DZ*DZ)
IF(DY.EQ.0,AND,DZ.EQ.0) GO TO 236
DO 235 J=1,NUNPE
  CDL(J,4) = -CDL(J,4)
  CONTINUE

CHANGE LOCAL Y-ROTATION FROM POSITIVE TO NEGATIVE
IF BEAM LIES ALONG X (GLOBAL AXIS)

IF(DX.LT.0.001,AND,DY.LT.-0.001) GO TO 246
DO 240 J=1,NUNPE
  CDL(J,5) = -CDL(J,5)
  CONTINUE

F = -AF - GAMMAM**H

CALCULATE THE TWISTING ANGLE

WX = DABS((CDL(2,4)-CDL(1,4))/EL)

EVALUATE SHAPE FUNCTIONS FOR N1Spl. - THREE POINT QUADRATURE

B2 = B**B
B3 = B2**B
B4 = B3**B
H2 = H**H
H3 = H2**H
WRITE(10,1002) I
WRITE(10,1004) E6,V
WRITE(10,1005) H,B

DO 300 K=1,3
  PSI = GFLW(K)
  CALL EU3DSP(PSI,SHF,F,DDSHF,F,2,EL)
  W = (SHFF(3)*CDL(1,3)+SHFF(5)*CDL(1,5))
  *(SHFF(7)*CDL(2,3)+SHFF(11)*CDL(2,5))

EVALUATE SHAPE FUNCTIONS FOR CURV. - THREE POINT QUADRATURE

WXX = DDSHF(F(3)*CDL(1,3)+DDSHF(5)*CDL(1,5))
  *DDSHF(7)*CDL(2,3)+DDSHF(11)*CDL(2,5))
WRITE(10,860) K,W,WXX,WXY

IF(NT.GT.1) GO TO 250

CALCULATE SENSITIVITY VECTORS

PJB = H3/3,DO=-.42D0#H2*(H2#H4/(4,D0#H2))
PJH = BH#H2-.2D0*B2*(H-H4/(12,B0#H3))
DPS1HG = 1P3IS8G+(-2*GAMMA#H#U-(HE#H2/2#))#WXX#WXX-
  PJB#G#WY#WXY#WIF(K)##(EL/-2,J0)
DPS1HG = 1P3IS8G+(-2*GAMMA#B#W-(HE#B2/2#))#WXX#WXX-
  PJH#G#WY#WXY#WIF(K)##(EL/-2,J0)
C CALCULATE PSI(B) - INTEGRAL OF FORCE*DISPLACEMENT
C
250   PSIBB(I) = PSIBB(I) + (F*W+AM*WXY)*WTW(K)*(EL/2.10)
300   CONTINUE
       WRITE(10,1003) DPSIBG,DFPSIHG
C
C IF(NT.GT.1) GO TO 820
    DPSIB(I) = DPSIBG
    DPSIH(I) = DPSIHG
C
C GO TO 820
C
C WRITE ERROR MESSAGES TO THE SCREEN
C
802   PRINT 872, IERR
       GO TO 820
807   PRINT 878, IERR
       GO TO 820
808   PRINT 879, IERR
       GO TO 820
809   PRINT 876, IERR
       GO TO 820
C
C 820   CONTINUE
C
C 851   FORMAT(/,1X,'BEAM WIDTH B='+F8.5,2X,'M:AM L:/TH=',F8.5,2X
       *,E='+E9.3,2X,'1Y=',E9.3,2X,'GAMMA=',F6.5,'APPLIED FORCE
       *=+F8.5)
855   FORMAT(1X,'NODE=',I2,2X,'X=',E12.5,2X,'Y=',E12.5,2X,'Z=',E12.5,2X,
       'RX=',E12.5,2X,'RY=',E12.5,2X,'RZ=',E12.5)
860   FORMAT(1X,'GP=',I2,4X,'W=',E11.5,4X,'WWX=',E11.5,4X,
       'WXY=',E11.5)
870   FORMAT(1X,'ACCELM REUKNED WITH ERROR ',14)
872   FORMAT(1X,'ACCMU REUKNED WITH ERROR ',14)
876   FORMAT(1X,'ACCEPR REUKNED WITH ERROR ',14)
878   FORMAT(1X,'ACCELCC REUKNED WITH ERROR ',14)
879   FORMAT(1X,'ACCMATT REUKNED WITH ERROR ',14)
1001  FORMAT(E12.5)
1002  FORMAT(1X,'ELEMEN T = ',I4)
1003  FORMAT(1X,'PSIBG=',E12.5,4X,'PSIHG=',E12.5)
1004  FORMAT(1X,'E=',E12.5,4X,'G=',E12.5,4X,'V=',E12.5)
1005  FORMAT(1X,'HEIGHT=',E12.5,4X,'WIDTH=',E12.5)
C
C RETURN
C
END
SUBROUTINE COMPS11(NF, IF1, ILCN)
C******************************************************************************
C COMPS11: CALCULATES COMPLIANCE AND SENSIT. FOR PLANt STRESS
C******************************************************************************
C DESCRIPTION:
C 'COMPS11' CALCULATES THE COMPLIANCE AND THE DESIGN
C SENSITIVITY OF THE FOUR AND EIGHT NODE PLANE STRESS
C ELEMENT IN TRACTION WITHOUT SELFWEIGHT.
C******************************************************************************
C NF = COUNTBK FOR FINITE DIFFERENCE
C IF1 = EXTERNAL ELEMENT NO. BEING PROCESSED
C ILCN = INTERNAL LOAD CASE NO.
C******************************************************************************
C INCLUDE 'CAEGSDR.INC) IMPLIC.SFC'
C INCLUDE 'CAEGSDR.INC) ACCIPN.MON'
C INCLUDE 'CAEGSDR.INC) UNEN.MON'
C INCLUDE 'CAEGSDR.INC) ELFLES.MON'
C INCLUDE 'CAEGSDR.INC) SVLEC.MON'
C
DIMENSION X(8), Y(8), Z(8), SHPF(B), GPL(2,4), DATN(50),
* BUF(100), PSIB(2), SBUF(50), CBFU(6), E0BUF(50),
* DSHFGX(R), DSHFGE(R), BF(4,48), S(E(50), DSHF+2, R),
* SIGMA(6,4), EPSLN(6,4)
C
DATA GPL/2#, .57735027, .57735027, .57735027,
* 2#, .57735027, .57735027, .57735027/
DATA KT/3/ + IREF/1/
C
SE = 0.
IF(NF.GT.1) GO TO 350
C
GET ELEMENT STRESSES AND STRAINS
C
CALL ACCFS (2, IPHEES, KINT, IREF, ILCN, SBUF, LEN, IERR)
IF(IERR NE 0) GOTO 401
LOC = LEN - 1
M = 1
DO 50 K = 1, NSVAL
SIGMA(1, K) = SBUF(M)
SIGMA(2, K) = SBUF(M+1)
SIGMA(3, K) = SBUF(M+3)
M = M+4
50 CONTINUE
DO 60 K = 1, NSVAL
EPSLN(1, K) = SBUF(M)
EPSLN(2, K) = SBUF(M+1)
EPSLN(3, K) = SBUF(M+3)
M = M + 4
60 CONTINUE
C
GET X AND Y FOR JACOBIAN EVALUATION
C
CALL ACCELC(2, IPNELC, KINT, IREF, BUF, LEN, IERR, FRR)
IF(IERR.NE.0) GO TO 807
M = 1
DO 200 L=1,LENX,J
   J = L+1
   K = L+2
   X(M) = BUF(L)
   Y(M) = BUF(J)
   Z(M) = BUF(K)
   M = M+1
200 CONTINUE

C CALCULATE FORCES AT THE GAUSS POINTS
C
DO 250 L=1,NUOF
   DO 250 K=1,NSVAL
      BF(K,L) = 0.0
250 CONTINUE

C LOOP OVER THE GAUSS POINTS
C
DO 300 K=1,NSVAL
   PSI = GFL(I,K)
   ETA = GFL(2,K)

C EVALUATE SHAPE FUNCTIONS AT THE GAUSS POINTS
C
   IF(ISTYP.EQ.2) CALL EU2D40(PSI,ETA,K,SHPF,DSHFL,
      DSHPGX,DSHPGY,DETJ,X,Y,IERR)
   IF(ISTYP.EQ.4) CALL EU2DPO(PSI,ETA,K,SHPF,DSHFL,
      DSHPGX,DSHPGY,DETJ,X,Y,IERR)
   IF(IERR.NE.0) GO TO 809
300 CONTINUE

WRITE(10,855)
   DO 320 K=1,NSVAL
      WRITE(10,854) K, (SIGMA(J,K),J=1,NSIG)
320 WRITE(10,860)
   DO 330 K=1,NSVAL
      WRITE(10,854) K, (EPSLN(J,K),J=1,NSIG)
330 WRITE(10,860)
   DO 340 J=1,NSIG
      SE(I) = SE(I) + SIGMA(J,K)*EPSLN(J,K)*DETJ
340 CONTINUE

C CALCULATE SENSITIVITY VECTOR
C
   DPSIT(I) = - SE(I)
350 GO TO 820

C WRITE ERROR MESSAGES TO THE SCREEN
C
801 PRINT 871, IERR
   GO TO 820
807 PRINT 878, IERR
   GO TO 820
809 PRINT 876, IERR
   GO TO 820
C
820 CONTINUE
C
C
854    FORMAT(1X,'2X,5X,3(E16.8,2X))
855    FORMAT(1X,'GP',5X,'SIGMAX(GP)',8X,'SIGMAY(GP)',8X,
        'SIGMAXY(GP)')
860    FORMAT(1X,'GP',5X,'EPSLNX(GP)',8X,'EPSLNY(GP)',8X,
        'EPSLNXY(GP)')
871    FORMAT(1X,'ACCFES RETURNED WITH ERROR ',I4)
876    FORMAT(1X,'EU2DF0 RETURNED WITH ERROR ',I4)
C
C
C
RETURN
END
SUBROUTINE CP16(NF,I,ILCN,PSIBTB)
CP***************************************************************
CP CP16: BRANCHES TO THE APPROPRIATE ELEMENT SUBTYPE
CP***************************************************************
CP CP* DESCRIPTION:
CP CP* 'CP16' BRANCHES TO THE APPROPRIATE ELEMENT SUBTYPE
CP CP* TO CALCULATE THE COMPLIANCE AND COMPLIANCE DESIGN
CP CP* SENSITIVITY OF THE PLATE BENDING ELEMENT 16.
CP CP* NOTE: THIS DOES NOT TAKE INTO ACCOUNT ANY MEMBRANE
CP CP* STIFFNESS.
CP CP***************************************************************
CP CP* NT COUNTER FOR FINITE DIFFERENCE
CP CP* I EXTERNAL ELEMENT NO. BEING PROCESSED
CP CP* ILCN INTERNAL LOAD CASE NO.
CP CP* PSIBTB COMPLIANCE OF A BENDING PLATE, USED FOR
CP CP* CALCULATING THE FINITE DIFFERENCE
CP CP***************************************************************
CP INCLUDE 'AEGSDR.INC' IMPLIC.SPC'
CP INCLUDE 'AEGSDR.INC' ELDES.MON'
CP INCLUDE 'AEGSDR.INC' SVEC.T.MON'
C DIMENSION PSIBTB(500)
C CP** BRANCH TO THE APPROPRIATE ELEMENT SUBTYPE
C IF(ISTYP.EQ.1) CALL CP1601(NF,I,ILCN,PSIBTB)
C IF(ISTYP.EQ.2) CALL CP1602(NF,I,ILCN,PSIBTB)
C RETURN
C END
SUBROUTINE CP1601(NF, I, ILCN, PSIBTR)
CP*******************************************************************************
CP CP1601: CALCULATES COMPLIANCE AND DESIGN SENSITIVITY
CP*******************************************************************************
CP DESCRIPTION:
CP CP CP1601' CALCULATES COMPLIANCE AND DESIGN SENSITIVITY
CP FOR A TRIANGULAR BENDING ELEMENT.
CP*******************************************************************************
CP NF COUNTER FOR FINITE DIFFERENCE
CP I EXTERNAL ELEMENT NO. BLING PROCESSED
CP ILCN INTERNAL LOAD CASE NO.
CP PSIBTR COMPLIANCE, USED FOR CALCULATING FINITE DIFFERENCE
CP*******************************************************************************
C INCLUDE 'CAEGSDR.INCJ IMPLICIT.SPC'
C INCLUDE 'CAEGSDR.INCJ ACCIPN.MON'
C INCLUDE 'CAEGSDR.INCJ CNIL.MON'
C INCLUDE 'CAEGSDR.INCJ ELEDES.MON'
C INCLUDE 'CAEGSDR.INCJ VECTR.MON'
C DIMENSION X(3), Y(3), PSIB(2), SBUF(100), (CFE(500), EF(3,6),
C SIGMA(6), EPSL(6), BUF(100), (EBUF(6), EDBUF(100),
C PSIBTR(500), CD(3,6), Z(3)
C DATA NT/J, IREF/1, MPT/1/
C CPE(I) = 0.
C PSIBTR(I) = 0.
C C GET PROPERTIES
C CALL ACCEPR(2, IPNFR, IPTAB, 0, BUF, LEN, IERR)
C IF(IERR .NE. 0) GO TO 809
C PB(NF) = BUF(20)
C IF(NF .GT. 1) GO TO 65
C C GET ELEMENT STRESSES AND STRAINS
C CALL ACCCFES(2, IPMFES, KINT, IREF, ILCN, SBUF, LEN, IERR)
C IF(IERR .NE. 0) GO TO 801
C M = 1
C DO 50 J=1, NDOF
C SIGMA(J) = SBUF(J)
C 50 CONTINUE
C M = 7
C DO 60 J=1, NDOF
C EPSL(J) = SBUF(M)
C M = M+1
C 60 CONTINUE
C C GET DISPLACEMENTS FOR PSI CALCULATION
C DO 70 J=1, NUMPL
C CALL ACCCNB(2, IPNCND, INTNN(J), I, ILCN, LFBUF, LEN, IERR)
C 70 CONTINUE
C
IF(IERR.NE.0) GO TO 802
DO 70 K=1,NDOF
   CD(J+K) = CBUF(K)
70 CONTINUE
C GET EQUIVALENT FORCES AT THE ELEMENT NODES FOR PSI CALC.
C CALL ACCEEN(2,IPLEN,KINF,IREF,ILCN,EDBUF,LEN,IER)
IF(IERR.NE.0) GO TO 808
M = 1
DO 80 J=1,NUNE
   DO 80 K=1,NDOF
      EF(J+K) = EDBUF(M)
      M = M+1
80 CONTINUE
IF(NT.GT.1) GO TO 350
C GET THE JACOBIAN
C CALL ACCELC(2,IPNEC,KINT,IREF,BUF,LENB,IER)
IF(IERR.NE.0) GO TO 807
M = 1
DO 200 J=1,LENB,E
   X(M) = BUF(J)
   Y(M) = BUF(J+1)
   Z(M) = BUF(J+2)
   M = M+1
200 CONTINUE
DETJ = EUTRIA(X,Y)
C CALCULATE SENSITIVITY VECTOR
C DO 340 J=1,3
   CPE(I) = CPE(I) + SIGMA(J)*EPSI(N(J))*DETJ
340 CONTINUE
DPSIB(I) = - CPE(I)
C CALCULATE PSI(B) - INTEGRAL OF FORCE*DISPLACEMENT IN Z
C DO 400 J=1,NUNE
   PSIB(J) = PSI(J) + EF(J+3)*CD(J+3)
400 CONTINUE
GO TO 820
C WRITE ERROR MESSAGES TO THE SCREEN
C 801 PRINT 871, IERR
   GO TO 820
802 PRINT 872, IERR
   GO TO 820
807 PRINT 877, IERR
   GO TO 820
808 PRINT 879, IERR
   GO TO 820
809 PRINT 880, IERR
   GO TO 820
C 820 CONTINUE
C
C

052 FORMAT(1X,'NODE',6X,'DISP X',8X,'DISP Y',8X,'DISP Z',8X,'ROT X',9X,'ROT Y',9X,'ROT Z')
053 FORMAT(1X,'ACCEFES RETURNED WITH ERROR ',14)
072 FORMAT(1X,'ACCENU RETURNED WITH ERROR ',14)
077 FORMAT(1X,'ACCELC RETURNED WITH ERROR ',14)
079 FORMAT(1X,'ACCEEN RETURNED WITH ERROR ',14)
080 FORMAT(1X,'ACCEPK RETURNED WITH ERROR ',14)
C
C
RETURN
END
SUBROUTINE CP1602(NT, ILCN, PSIBTR)
*
*  CF*  CF1602: CALCULATES COMPLIANCE AND DESIGN SENSITIVITY
*  CF*  DESCRIPTION:
*  CF*  'CP1602' CALCULATES COMPLIANCE AND DESIGN SENSITIVITY
*  CF*  FOR A FOUR NOVE PAVING ELEMENT.
*  CF*
*
*  CF*  NT  COUNTER FOR FINITE DIFFERENCE
*  CF*  I   EXTERNAL NO. BEING PROCESSED
*  CF*  ILCN INTERNAL LOAD CASE NO.
*  CF*  PSIBTR COMPLIANCE
*  CF*
*
C INCLUDE 'CAEGSDR, INCJ IMPLIC, SPC'
INCLUDE 'CAEGSDR, INCJ ACCPIN, MON'
INCLUDE 'CAEGSDR, INCJ 'N1L, MON'
INCLUDE 'CAEGSDR, INCJ ELEDES, MON'
INCLUDE 'CAEGSDR, INCJ SVECTR, MON'
C DIMENSION X(4), Y(4), PSIB(2), SBUF(100), CPF(500), ET(4,6), Z(4), 
  SIGMA(6,4), EPSLN(6,4), BUF(100), CFUB(6), 
  EBUF(100), PSIBTR(500), CD(4,6)
C DATA K/3/, IREF/1/, M/1/
C CF(E) = 0.,
PSIBTR(I) = 0.,
C GET PROPTEIES
C CALL ACCEPR(2, IP, IPR, IPTAB, 0, BUF, LEN, IERR)
IF(IERR, NE, 0) GO TO 999
PB(N) = BUF(25)
IF(N, GT, 1) GO TO 65
C GET ELEMENT STRESSES AND STKNVHS
C CALL ACCFES(2, IP, IES, HIN, IREF, ILCN, SBUF, LEN, IERR)
IF(IERR, NE, 0) U010 801
M = 1
DO 50 K=1, NSVAL
  J = M+1
  L = M+2
  SIGMA(1,K) = SBUF(M)
  SIGMA(2,K) = SBUF(J)
  SIGMA(3,K) = SBUF(L)
  M = M+6
50 CONTINUE
DO 60 K=1, NSVAL
  J = M+1
  L = M+2
  EPSLN(1,K) = SBUF(M)
EPSSLN(2,K) = SBUF(J)
EPSSLN(3,K) = SBUF(L)
M = M+6
60 CONTINUE
C GET DISPLACEMENTS FOR PSI CALCULATION
C
65 DO 70 J=1,NUNPE
CALL ACCMU(2,1,PMCN,INTMH(J),1,ILCN,FBUF,LNF,IF) ERR
IF(IERR.NE.0) GO TO 802
DO 70 K=1,NUOF
CD(J,K) = (FBUF(K)
70 CONTINUE
C GET EQUIVALENT FORCES AT THE ELEMENT NODES FOR PSI (CALC.
C
CALL ACCEHE(2,IPNFR,KNINT,IREF,ILCN,FBUF,LNC,IF,IF) ERR
IF(IERR.NE.0) GO TO 808
M = 1
DO 80 J=1,NUNFE
DO 80 K=1,NDOF
EF(J,K) = EGBUF(M)
M = M+1
80 CONTINUE
IF(NT.GT.1) GO TO 350
C GET THE JACOBIAN
C
CALL ACCEL(2,IPNLR,KNINT,IREF,LNC,FBUF,LNC,IF) ERR
IF(IERR.NE.0) GO TO 807
M = 1
DO 200 J=1,LNF,3
X(M) = BUF(J)
Y(M) = BUF(J+1)
Z(M) = BUF(J+2)
M = M+1
200 CONTINUE
DETJ = AREA(X,Y)
DETJ = DETJ/4.0
C C CALCULATE SENSITIVITY VECTOR
C
DO 340 J=1,3
DO 340 K=1,NSVAL
CPE(I) = CPE(I) + SIGM(A(J,K)*EPSSLN(J,K)*DETJ
340 CONTINUE
DPSIB(I) = - CPE(I)
C C CALCULATE PSI(R) - INTEGRAL OF FORCE*DISPLACEMENT IN R
C
350 DO 400 J=1,NUNPE
PSIDIB(I) = PSIDIB(I) + EF(J,3)*CD(J,3)
400 CONTINUE
GO TO 820
C C WRITE ERROR MESSAGES TO THE SCREEN
C
801 PRINT 871, IERR
GO TO 820
802 PRINT 872, IERR
GO TO 820
807 PRINT 877, IERR
GO TO 820
808 PRINT 879, IERR
GO TO 820
809 PRINT 880, IERR
GO TO 820
C
C
820 CONTINUE
C
C
852 FORMAT(1X,'NODE',6X,'DISP X','8X','DISP Y','8X','SISF
  *8X','ROT X','9X','ROT Y','9X','ROT Y')
853 FORMAT(1X,I4,6(2X,E12.4))
871 FORMAT(1X,'ACCFES RETURNED WITH ERROR ',I4)
872 FORMAT(1X,'ACCNND RETURNED WITH ERROR ',I4)
873 FORMAT(1X,'ACCELc RETURNED WITH ERROR ',I4)
874 FORMAT(1X,'ACCEEN RETURNED WITH ERROR ',I4)
C
C
C
RETURN
END
SUBROUTINE CPS11I(FSI1T,111,1REF)

C INCLUDE 'EAGSDR.INCJ IMPLIC.END'
INCLUDE 'EAGSDR.INCJ ACCIPN.MON'
INCLUDE 'EAGSDR.INCJ 1N1L,MON'
INCLUDE 'EAGSDR.INCJ KLEDK.MON'

C DIMENSION DATN(50),CFBUF(6)

C CALCULATE FSI(B) FOR PLATE - FORCF*1DISPLACEMENT

C PRINT *,
PRINT *,ENTER NODE WHERE LOAD IS APPLIED
READ(5,1004) NEXT
PRINT *,
PRINT *,ENTER LOAD DIRECTION(1:X,...,6:Z)
READ(5,1004) LDIR

C GET INTERNAL NODE NUMBERS

C CALL ACCNOD(2,IFNMOD,HEXI2,1DATN,IERR)
IF(IERR.NE.0) GO TO 806
NINT = DATN(4)

C GET DISPLACEMENT AT NODE

C CALL ACCNU(2,IFNCND,NINT,1REF,IL1,CFBUF,LEN,IERR)
IF(IERR.NE.0) GO TO 802
DISP = CFBUF(LDIR)

C FSI1T = 40000*DISP

C WRITE ERROR MESSAGES

802 PRINT 872, IERR
GO TO 820
806 PRINT 877, IERR
GO TO 820

820 CONTINUE

C FOR(1X,'ACCNOD RETURNED WITH ERROR ',IERR)
872 FORMAT(1X,'ACCNOD RETURNED WITH ERROR ',IERR)
877 FORMAT(1X,'ACCNOD RETURNED WITH ERROR ',IERR)
1004 FORMAT(I4)

C RETURN

END
SUBROUTINE DISP(PSIB, NT, NELM)
CP************************************************************************************
CP* DISP: BRANCHES TO THE APPROPRIATE ELEMENT TYPE FOR TH.
CP* CP************************************************************************************
CP* DESCRIPTION:
CP* CP* 'DISP' BRANCHES TO THE APPROPRIATE ELEMENT TYPE FOR
CP* CP* CALCULATION OF THE DISPLACEMENT CONSTRAINT AND THE
CP* CP* DISPLACEMENT SENSITIVITY VECTOR.
CP* CP************************************************************************************
CP* PSIB DISPLACEMENT AT X
CP* NT COUNTER FOR FINITIF DIFFERENCE
CP* NELM TOTAL NO. OF ELEMENTS
CP* CP************************************************************************************
C INCLUDE 'AEGSDR.INC' IMPLICIT, SPC
C INCLUDE 'AEGSDR.INC' ACCIPN, MON
C INCLUDE 'AEGSDR.INC' CNL, MON
C INCLUDE 'AEGSDR.INC' ELFUS, MON
C INCLUDE 'AEGSDR.INC' SVCTR, MON
C EQUIVALENCE (NDAT(97), INBS), (NDAT(98), IDBL)
C DIMENSION DATH(50), PSIB(2), CBUF(6)
C DATA IREF/1/
C SDFSTR = 0.0
C SDFSIT = 0.0
C SDFPSIB = 0.0
C SDFPSIH = 0.0
C IF(NT.GT.1) GO TO 10
C SET ADJOINT LOAD CASE NUMBER
C L2 = LCS + NC
C SET ORIGINAL LOAD CASE NUMBER
C L1 = 1
C SETUP POINTERS
C CALL ACCELM(1, IPNLM, IDBS, 1, 0, IERR)
C IF(IERR.NE.0) GO TO 800
C CALL ACCFES(1, IPNFES, IDBL, 1, L1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 802
C CALL ACCCND(1, IPNNND, IDBL, 1, L1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 802
C CALL ACCEL(1, IPNLLC, IDBS, 1, 0, 0, IERR)
C IF(IERR.NE.0) GO TO 807
CALL ACCHAT(1,IPNMAT,IDBS,1,0,0,IELR)
IF(IERR,NR,0) GO TO 808
CALL ACCEPR(1,IPHEPR,IDBS,1,0,0,IERR)
IF(IERR,NR,0) GO TO 809
C
C LOOP THROUGH THE BUFFERS TO GET STRESSES, STRAINS, MOMENTS, ETC.
C
DO 100 I=1,NLEM
C
C GET ELEMENT DESCRIPTORS
C
CALL ACCELM(2,IPHELM,I,IREP,IFD,IERR)
IF(IERR,NR,0) GO TO 800
C
IF(NY,GT,1) GO TO 720

C BRANCH TO THE APPROPRIATE ELEMENT TYPE
C
IF(IFY.EQ.11) CALL DISP11(N1,I,L1,L2,L11)
IF(IFY.EQ.5) CALL DISP05(N1,I,L1,L2,L11)
IF(IFY.EQ.16) CALL DP16(N1,I,L1,L2,L11)

C
SDPSIT = SDPSIT+DPSIT(I)
SDPSIB = SDPSIB+DPSIB(I)
SDPSIH = SDPSIH+DPSIH(I)
SDPSIB = SDPSIB+DPSIB(I)

100 CONTINUE
C
WRITE(10,859)
DO 710 I=1,NLEM
710 WRITE(10,857) I,PSIT(I),DPSIB(I),DPSIH(I),DPSIB(I)
WRITE(10,861) SDPSIT,SDPSIB,SDPSIH,SDPSIB
C
C CALCULATE PSI(J) - ORIGINAL DISPL AT NODE WHERE ADJU LOAD APPLIED
C
720 PRINT *, *
PRINT *, 'ENTER NODE NUMBER WHERE ADJU LOAD IS APPLIED'
READ(5,1004) NEXT
LDIR = 3
C
C GET INTERNAL NODE NUMBER
C
CALL ACCMOD(2,IPNMOD,NEXT,2,IATN,IERR)
IF(IERR,NR,0) GO TO 806
NINT = IATN(4)
C
C GET DISPLACEMENT AT NODE
C
CALL ACCCMOD(2,IPNCM,NI1,IREF,IL1,ClBUF,LFN,IERR)
IF(IERR,NR,0) GO TO 802
DIS = ClBUF(LDIR)
C
PSIB(NT) = -DIS
C
WRITE(10,858) PSIB(NT)
WRITE 858, PSIB(NT)
C
IF(NY,EQ,1) GO TO 720
WRITE(10,**)**
WRITE(10, 856) NEXT
C
C CLEAN-UP EVERYTHING
C
730 CALL ACCELH(4, IPNLCS, 4, IERR)
    IF(IERR.NE.0) GO TO 800
    CALL ACCFES(4, IPNFES, 0, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 801
    CALL ACCNOD(4, IPNOD, 0, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 802
    IF(NT.GT.1) GO TO 750
    CALL ACCLCS(4, IPNLCS, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 805
750 CALL ACCNOD(4, IPNOD, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 806
    CALL ACCELH(4, IPHEV, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 807
    CALL ACCMAT(4, IPMAT, 0, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 808
    CALL ACCEFR(4, IPNEFR, 0, 0, 0, IERR)
    IF(IERR.NE.0) GO TO 809
C
    GO TO 820
C
C WRITE ERROR MESSAGES TO THE SCREEN
C
800 PRINT 870, IERR
    GO TO 820
801 PRINT 871, IERR
    GO TO 820
802 PRINT 872, IERR
    GO TO 820
805 PRINT 875, IERR
    GO TO 820
806 PRINT 877, IERR
    GO TO 820
807 PRINT 878, IERR
    GO TO 820
808 PRINT 879, IERR
    GO TO 820
809 PRINT 876, IERR
    GO TO 820
C
820 CONTINUE
C
856 FORMAT(1X, '*** ADJOIN LOAD IS APPLIED AT NODE', 14)
857 FORMAT(13, 4X, 4(E16.8, 4X))
858 FORMAT(1X, 'FSIB=', E16.8)
861 FORMAT(1X, /, 1X, 'TOTAL=', 4(E16.8, 4X))
862 FORMAT(1X, 'KLEMEN ', 14)
870 FORMAT(1X, 'ACCEL RETURNED WITH ERROR', 14)
871 FORMAT(1X, 'ACCFES RETURNED WITH ERROR', 14)
872 FORMAT(1X, 'ACCCNU RETURNED WITH ERROR', 14)
875 FORMAT(1X, 'ACCLCS RETURNED WITH ERROR', 14)
876 FORMAT(1X, 'ACCEFR RETURNED WITH ERROR', 14)
877 FORMAT(1X, 'ACCCNU RETURNED WITH ERROR', 14)
878    FORMAT(1X,'ACCEL RETURNED WITH ERROR ',14)
879    FORMAT(1X,'ALCMAT RETURNED WITH ERROR ',14)
1004   FORMAT(I4)
2000   FORMAT(A)
C
C
RETURN
END
SUBROUTINE DISPOS(MF, I, L1, L2, II, L)
CP*******************************************************************************
CP* DISPOS: CALCULATES THE DISPL. DESIGN SENSITIVITY IF A BEAM CP*
CP*******************************************************************************
CP* DESCRIPTION:
CP*
CP* 'DISPOS' CALCULATES THE DISPLACEMENTS UNDER CONSTR. DESIGN CP*
CP* SENSITIVITY OF A 1-D BEAM IN BENDING, WITH AN APPLIED CP*
CP* ELEMENT FORCE IN #/IN. SELF WEIGHT IS NEGLECTED. CP*
CP* TORSION IS ACTIVE.
CP*
CP*******************************************************************************
CP* NT COUNTER FOR FINITE DIFFERENCE
CP* I EXTERNAL ELEMENT NO. BEING PROCESSED
CP* L1 ORIGINAL EXTERNAL LOAD CASE NO.
CP* L2 EXTERNAL ADJOIN LOAD CASE NO.
CP* IL1 INTERNAL LOAD CASE NO. OF ORIGINAL LOAD
CP*
CP*******************************************************************************

INCLUDE 'CAEGSR. INC1 IMPLICEFC
INCLUDE 'CAEGSR. INC1 ACCIPN.MON'
INCLUDE 'CAEGSR. INC1 CNTL.MON'
INCLUDE 'CAEGSR. INC1 ELEDES.MON'
INCLUDE 'CAEGSR. INC1 SVECTR.MON'
COMMON/LCSDES/ULCS(90)

EQUIVALENCE (NUAT(14),IF),(NUAT(YB),IRBL)
DIMENSION X(J),Y(J),Z(J),USHEF(12),HAIN(50),BUF(100),
SHFF(12),CFH(3),GPLW(3),ALD(2,6),CD(2,6),WIFW(3),
C(6,2),H(6,2),T(3,3),TB(6,6),ALDL(2,6),ALDL(2,6),
COOR(3,3)

DATA GPLW/-.77459667, .00000000, .74596677/
DATA WTW/ .555555556, .00000000, .555555556/
DATA KT/3/, IREF/1/, HPT/1/

DFSIHG = 0.0
DFSIHG = 0.0
IF(1.GT.1) GO TO 50

REQUEST APPLIED FORCE IN LOAD/LENGTH

PRINT *, 'ENTER APPLIED LOAD IN FPCE/LENGTH'
READ(5,1001) AF

GET AREA MOMENT OF INERTIA ABOUT Y-AXIS

CALL ACCEPR(2,IPNEPR,IPTH,0,BUF,LEN,IERR)
IF(IERR,NL,0) GO TO 809
Y1 = BUF(5)
H = 2*BUF(9)
B = 2*BUF(10)
BH(NF) = H
BW(HI) = B
GET WEIGHT DENSITY AND MODULUS OF ELASTICITY

CALL ACCMAT(2,IPNMT,NIAC,MPF,BUF,LEN,IERR)
IF(IERR,NE,0) GO TO 808
GAMMA = 0.000
E = BUF(5)
V = BUF(7)
G = E/(2.00*(1.00+V))

IF(I.GT.1) GO TO 60

GET INTERNAL LOAD CASE NUMBER FOR ORIGINAL LOAD

CALL ACCLCS(1,IPNLCS,IDL,BL,1,0,0,IERR)
IF(IERR,NE,0) GO TO 805
CALL ACCLCS(2,IPNLCS,L1,2,DLCs,IERR)
IF(IERR,NE,0) GO TO 805
IL1 = DLCs(21)

GET DISPLACEMENTS AT ELEMENT ENDS

CALL ACCND(1,IPNCND,IDL,BL,1,1,0,0,0,IERR)
IF(IERR,NE,0) GO TO 802
DO 70 J=1,NUMF
    CALL ACCND(2,IPNCND,INTNM(J),1,IL1,CFBUF,LEN,IERR)
    IF(IERR,NE,0) GO TO 802
    DO 70 K=1,NODF
        CD(J,K) = CFBUF(K)
    70 CONTINUE

GET INTERNAL LOAD CASE NUMBER FOR ADJACENT LOAD

CALL ACCLCS(1,IPNLCS,IDL,BL,1,0,0,IERR)
IF(IERR,NE,0) GO TO 805
CALL ACCLCS(2,IPNLCS,L2,2,DLCs,IERR)
IF(IERR,NE,0) GO TO 805
ILCN = DLCs(21)

GET DISPLACEMENTS AT ELEMENT ENDS FOR ADJACENT LOAD

CALL ACCND(1,IPNCND,IDL,BL,1,1,0,0,0,IERR)
IF(IERR,NE,0) GO TO 802
DO 100 J=1,NUMF
    CALL ACCND(2,IPNCND,INTNM(J),1,ILCN,CFBUF,LEN,IERR)
    IF(IERR,NE,0) GO TO 802
    DO 100 K=1,NODDF
        ALD(J,K) = CFBUF(K)
    100 CONTINUE

EVALUATE DISPLS. AND CURVATURE AT THE GAUSS POINT USING
SHAPE FUNCTIONS - ONE PT. FOR CURV., THREE PT. FOR DISPL

GET X, Y, AND Z OF ELEMENT NODES

CALL ACCELNC(2,IPNLCE,KINF,INREF,BUF,LEN,IERR)
IF(IERR,NE,0) GO TO 807
M = 1
DO 200 J=1,9,3
    K = J+1
\[ L = J+2 \]
\[ X(H) = BUF(J+1) \]
\[ Y(H) = BUF(K) \]
\[ Z(H) = BUF(L) \]
\[ M = M+1 \]

200 CONTINUE
DO 210 J=1,3
COOR(1,J) = X(J)
COOR(2,J) = Y(J)
COOR(3,J) = Z(J)
210 CONTINUE

C FORM THE ELEMENT LOCAL COORDINATE SYSTEM FOR DISPLACEMENTS

C
IN3 = INFHM(3)
CALL EUHM(IN3,BETA,COOR,1,1,ERR)
CALL ZEROSP(TB,364IP)
DO 220 J=1,3
DO 220 K=1,3
TB(J+K) = T(J,K)
\[ TB(J+K+3) = T(J,K) \]
220 CONTINUE
CALL UHXA(B,TB,CD,6,2,6)
DO 230 J=1,2
DO 230 K=1,6
CDL(J,K) = C(K,J)
230 CONTINUE
CALL UHXA(B,ALN,6,2,6)
DO 232 J=1,2
DO 232 K=1,6
ALDL(J,K) = D(K,J)
232 CONTINUE

C CALCULATE ELEMENT LENGTH

C
DX = X(2)-X(1)
DY = Y(2)-Y(1)
DZ = Z(2)-Z(1)
EL = DSQRT(DX*DX+DY*DY+DZ*DZ)

C CHANGE LOCAL \( Y \)-ROTAIUM FROM POSITIVE TO NEGATIVE IF BEAM LIES ALONG X GLOBAL AXIS

C
IF(DX,L.T.0.001,.AND.DX,G.T.-0.001) GO TO 246
DO 240 J=1,HNUMPE
CDL(J,5) = -CDL(J,5)
ALDL(J,5) = -ALDL(J,5)
240 CONTINUE

C
F = -AF - GAMMA*B*H

C CALCULATE THE TWISTING ANGLES

C
WXY = DABS((CDL(2,4)-CDL(1,4))/EL)
AWXY = DABS((ALDL(2,4)-ALDL(1,4))/EL)

C EVALUATE SHAPE FUNCTIONS FOR DISPL. - THREE POINT QUADRATURE

C
B2 = B*B
B3 = B2*B
B4 = B3*B
H2 = H*H  
H3 = H2*H

C  DO 300 K=1,3  
   PSI = GPLW(K)  
   CALL EU3DISB(PSI, SHFF, DDSSHFF, 2, EL)  
   W = (SHFF(3)*CDL(1,3)+SHFF(5)*CDL(1,5)+SHFF(9)*CDL(2,3)+SHFF(11)*CDL(2,5))  
   AW = (SHFF(3)*ALDL(1,3)+SHFF(5)*ALDL(1,5)+SHFF(9)*ALDL(2,3)+SHFF(11)*ALDL(2,5))

C  EVALUATE SHAPE FUNCTIONS FOR CURV. - THREE POINT QUADRATURE

C  WXX = (DDSSHFF(3)*CDL(1,3)+DDSSHFF(5)*CDL(1,5)+DDSSHFF(9)*CDL(2,3)+DDSSHFF(11)*CDL(2,5))  
   *AWXX = (DDSSHFF(3)*ALDL(1,3)+DDSSHFF(5)*ALDL(1,5)+DDSSHFF(9)*ALDL(2,3)+DDSSHFF(11)*ALDL(2,5))

C  CALCULATE SENSITIVITY VECTORS

C  PJH = H3/3, DO - 420/10*(H2+B4/(4, DO/H2))  
   PZH = B4H2 - 420/2*H-K4/(12, DO/H3))  
   DFSISH = DFSISH+(-GAMMA*B*AW-(E*H3/12)*AWXX*WXX-  
   *PJH*WXY*WXY)*W(K)*(E/2, DO)  
   DFSISH = DFSISH+(-GAMMA*B*AW-(3*E*B*H2/12)*AWXX*WXX-  
   *PJH*WXY*WXY)*W(K)*(E/2, DO)

300  CONTINUE  
   DFSISH = DFSISH  
   DFSISH = DFSISH

C  GO TO 820

C  WRITE ERROR MESSAGES TO THE SCREEN

C  802  PRINT 872, IERR  
   GO TO 820

C  805  PRINT 875, IERR  
   GO TO 820

C  807  PRINT 878, IERR  
   GO TO 820

C  808  PRINT 879, IERR  
   GO TO 820

C  809  PRINT 876, IERR  
   GO TO 820

C

C  820  CONTINUE

C

C  851  FORMAT(/1X,'BEAM WIDTH B=',F8.5,2X,'H=AM DEPTH=',F8.5,2X  
   *,E9.3,2X,'IYY=',E9.3,2X,'GAMMA=',F6.5,2X,'APPLIED  
   #FORCE=',F8.5)

C  855  FORMAT(1X,'NODE=',I2,2X,'X=',E12.5,2X,'Y=',E12.5,2X,'Z=',  
   *E12.5,2X,'KX=',E12.5,2X,'RY=',E12.5,2X,'RZ=',E12.5)

C  860  FORMAT(1X,'BF=',E12.5,2X,'W=',E11.5,4X,'WXX=',E11.5,4X,'AWX=',  
   *E11.5,4X,'AWXY=',E11.5)

C  864  FORMAT(1X,'NO=X=',E12.5,2X,'AX=',E12.5,2X,'AY=',E12.5,2X,  
   *AZ=',E12.5,2X,'ARX=',E12.5,2X,'ARY=',E12.5,2X,'ARZ=',  
   *E12.5)

C  872  FORMAT(1X,'ACCDN RETURNED WITH ERROR ',I4)

C  875  FORMAT(1X,'ALGCIS RETURNED WITH ERROR ',I4)
976  FORMAT(1X, 'ACCEPR RETURNED WITH ERROR ', I4)
978  FORMAT(1X, 'ACCELC RETURNED WITH ERROR ', I4)
979  FORMAT(1X, 'ACLMAT RETURNED WITH ERROR ', I4)
1001 FORMAT(E12.5)

C

RETURN

END
SUBROUTINE DISP11(NT, I, L1, L2, IL1)

DESCRIPTION:

'DISP11' CALCULATES THE DISPLACEMENT SENSITIVITY FOR THE FOUR AND EIGHT NODE PLANE STRESS ELEMENT WITH TRACTION, SFLF WEIGHT NOT INCLUDED.

NT COUNTER FOR FINITE DIFFERENCE
I EXTERNAL ELEMENT NO. BEING PROCESSED
L1 ORIGINAL EXTERNAL LOAD CASE NO.
L2 EXTERNAL ADJOINT LOAD CASE NO.
IL1 ORIGINAL INTERNAL LOAD CASE NO. RETURNED TO DISP.FOR

INCLUDE 'CAGSDR, INC' IMPLIC, SFC'
INCLUDE 'CAGSDR, INC' ACCJFN, MON'
INCLUDE 'CAGSDR, INC' NTL, MON'
INCLUDE 'CAGSDR, INC' ELEDES, MON'
INCLUDE 'CAGSDR, INC' SVI, CTR, MON'
COMMON/LCSDES/LDLS(90)

EQUIVALENCE (MNAT(98), IDUL)

DIMENSION SHO(8), GPL(2, 4), BUM(100), X(8), Y(8), Z(8),
DGP(50), PSIB(2), SBUF(50), DSHFQ(4), DSHFG(8),
DSHPL(2, 8), SIGMA(6, 4), EPSLN(6, 4), HF(4, 8), SE(500)

DATA GPL/2* -57735027/, 57735027/, -57735027/, 57735027/,
2* -57735027/ -57735027/, 57735027/,
DATA KT/3/, IREF/1/, MIF/1/1/

JJ = L1

GET INTERNAL LOAD CASE NUMBER

CALL ACCLCS(1, PNL, S, IDBL, 1, 0, IERR)
IF (IERR .NE. 0) GO TO 805
CALL ACCLCS(2, IFNLCS, JJ, 2, 1, 5, IERR)
IF (IERR .NE. 0) GO TO 805
ILCN = DLCS(21)

SETUP POINTER FOR STRESS-STRAIN BUFFER

CALL ACCEFES(1, IPFES, ) NL, ILCN, 0, 0, IERR
IF (IERR .NE. 0) GO TO 801
IF (JJ .EQ. L1) IL1 = ILCN
SE(1) = 0.0

GET ELEMENT STRESSES AND STRAINS

CALL ACCEFES(2, IPFES, KIN, IREF, ILCN, SBUF, LEN, 1, IERK)
IF (IERR .NE. 0) GOTO 801
LOC = LEN - 1
M = 1
IF (JJ.EQ.L?) GO TO 55
DO 50 K=1,NSVAL
   SIGMA(1,K) = SBUF(M)
   SIGMA(2,K) = SBUF(M+1)
   SIGMA(3,K) = SBUF(M+3)
   M = M+4
50 CONTINUE
GO TO 55
55 M = 17
DO 60 K=1,NSVAL
   EPSLN(1,K) = SBUF(M)
   EPSLN(2,K) = SBUF(M+1)
   EPSLN(3,K) = SBUF(M+3)
   M = M+4
60 CONTINUE
65 IF (JJ.EQ.L?) GO TO 100
JJ = L2
GO TO 10
C GET X AND Y FOR JACOBIAN EVALUATION
C 100 CALL ACCEL(2,IPNEL+KIN,IREF,HUF,LENB,IERR)
IF(IERR.NE.0) GO TO 807
M = 1
DO 200 L=1,LENP,3
   X(L) = BUF(L)
   Y(L) = BUF(L+1)
   Z(L) = BUF(L+2)
   M = M+1
200 CONTINUE
C CALCULATE FORCES AT THE GAUSS POINTS
C DO 250 L=1,NDOF
DO 250 K=1,NSVAL
   BF(K,L) = 0.0
250 CONTINUE
C LOOP OVER THE GAUSS POINTS
C DO 300 K=1,NSVAL
   PSI = GFL(1,K)
   ETA = GFL(2,K)
C EVALUATE SHAPE FUNCTIONS AT THE GAUSS POINTS
C IF(ISTYP.EQ.2) CALL EU2DFG(PSI,ETA,KF,SHFX,SHFY,DSHFX,DSHFY,DETJ,X,Y,IERR)
* IF(ISTYP.EQ.4) CALL EU2DFG(PSI,ETA,KF,SHFX,SHFY,DSHFX,DSHFY,DETJ,XX,YY,IERR)
* IF(IERR.NE.0) UTD H09
300 CONTINUE
DO 340 J=1,NSIG
DO 340 K=1,NSVAL
   SE(I) = SE(I) + SIGMA(J,K)*EPSLN(J,K)*DETJ
340 CONTINUE
C CALCULATE SENSITIVITY VECTOR
DPSIT(I) = - SE(I)

700 CONTINUE
C
GO TO 820
C WRITE ERROR MESSAGES TO THE SCREEN
C
801 PRINT 871, IERR
GO TO 820
805 PRINT 875, IERR
GO TO 820
807 PRINT 878, IERR
GO TO 820
809 PRINT 876, IERR
GO TO 820
C
820 CONTINUE
C
C
954 FORMAT(1X,12,2X,3(E16.8,2X))
955 FORMAT(1X,'GP','SIGMAX(GP)','SIGMAY(GP)','EPSLNX(GP)'
   *'EPSLNY(GP)'),8X,'
960 FORMAT(1X,'GP','SIGMAX(GP)','SIGMAY(GP)','EPSLNX(GP)'
   *'EPSLNY(GP)'),8X,'
971 FORMAT(1X,'ACCFES RETURNED WITH ERROR ',14)
975 FORMAT(1X,'ACCLCS RETURNED WITH ERROR ',14)
976 FORMAT(1X,'ACCELCS RETURNED WITH ERROR ',14)
978 FORMAT(1X,'ACCELCS RETURNED WITH ERROR ',14)
C
RETURN
END
SUBROUTINE DP16(NF,N1,L1,L2,IL1)
CP******************************************************************************
CP* CP16: BRANCHES TO THE APPROPRIATE ELEMENT SUBTYPE
CP*
CP******************************************************************************
CP* DESCRIPTION:
CP* 'CP16' BRANCHES TO THE APPROPRIATE ELEMENT SUBTYPE
CP* TO CALCULATE THE DISPLACEMENT DESIGN SENSITIVITY OF
CP* THE PLATE BENDING ELEMENT 16.
CP* NOTE: THIS DOES NOT TAKE INTO ACCOUNT ANY MEMBRANE
CP* STIFFNESS.
CP*
CP******************************************************************************
CP* NT COUNTER FOR FINITE DIFFERENCE
CP* I EXTERNAL ELEMENT NO. BEING PROCESSED
CP* L1 ORIGINAL EXTERNAL LOAD CASE NO.
CP* L2 EXTERNAL ADJOINING LOAD CASE NO.
CP* IL1 ORIGINAL INTERNAL LOAD CASE NO. - KE TURNED VALUE
CP*
CP******************************************************************************
CP* INCLUDE 'CAEGSRD.INC' IMPLICIT,SPC'
CP* INCLUDE 'CAEGSRD.INC' EIES.EES'
CP* INCLUDE 'CAEGSRD.INC' SVECTR.MON'
CP*
C** BRANCH TO THE APPROPRIATE ELEMENT SUBTYPE
C IF(ISTYP.EQ.1) CALL DP1601(NF,I,L1,L2,IL1)
IF(ISTYP.EQ.2) CALL DP1602(NF,I,L1,L2,IL1)
C
RETURN
END
SUBROUTINE DF1601(NF,1,L1,L2,L1)
*********************************************************************************
C**
C** DF1601: CALCULATES THE DISPLACEMENT CONSTRAINT DESIGN
C**
C******************************************************************************
C** DESCRIPTION:
C**
C** 'DF1601' CALCULATES THE DISPLACEMENT CONSTRAINT DESIGN
C** SENSITIVITY VECTOR FOR A TRIANGULAR PLATE BENDING
C** ELEMENT.
C******************************************************************************
C**
C** NT COUNTER FOR FINITE DIFFERENCE
C** I EXTERNAL ELEMENT NO. BEING PROCESSED
C** L1 ORIGINAL EXTERNAL LOAD CASE NO.
C** L2 EXTERNAL ADJOIN LOAD CASE NO.
C** IL1 ORIGINAL INTERNAL LOAD CASE NO. - RETURN: RETURN VALUE
C******************************************************************************
C**
C** INCLUDE 'AEAGSR,INC' IMPLIC.SFC'
C** INCLUDE 'AEAGSR,INC' ALCIP.MON'
C** INCLUDE 'AEAGSR,INC' CNTL.MON'
C** INCLUDE 'AEAGSR,INC' EPLDES.MON'
C** INCLUDE 'AEAGSR,INC' KV:CR.MON'
C** COMMON/LCSDES/MLCS(90)
C**
C** EQUIVALENCE (MV(98),IDBL)
C**
C** DIMENSION SIGMA(6),EPSLN(6),BUFF(100),BUF(100),CPE(500),
C** X(3),Y(3),Z(3)
C**
C** DATA NT/3/,IREF/1/
C**
C** JJ = L1
C** CPE(1) = 0.0
C**
C*** GET INTERNAL LOAD CASE NUMBER
C**
C10 CALL ACCLCS(L1,JNLLS,1DBL,1,0,IERR)
C IF(IERR.NE.0) GO TO 805
C CALL ACCLCS(J,JLNC,1,2,DLCS,IERR)
C IF(IERR.NE.0) GO TO 805
C ILCN = DLCS(21)
C**
C*** GET PROPERTIES
C**
C CALL ACCEPR(1,IFNEF,IPR,0,BUF,LFN,IERR)
C IF(IERR.NE.0) GO TO 809
C FB(NT) = BUF(75)
C**
C*** SETUP POINTER FOR STRESS-STRAIN BUFFER
C**
C CALL ACCES(1,IFNFS,IDBL,J,ILCN,0,0,IERR)
C IF (IERR.NF.0) GO TO 801
C IF (JJ.EQ.L1) IL1 = ILCN
C**
C*** GET ELEMENT STRESSES AND STRAINS
C**
CALL ACCELS(2,IPHES,KINT,IRFF,ILCN,SBUF,LEN,IERR)
IF(IERR,NE.0) GOTO H01
IF(JJ,EO,L2) GO TO 55
DO 50 K=1,NDOF
   SIGMA(K) = SBUF(K)
50 CONTINUE
GO TO 65
M = 7
DO 60 K=1,NDOF
   EPSLN(K) = SBUF(M)
   M = M+1
60 CONTINUE
IF(JJ,EO,L2) GO TO 100
JJ = L2
GO TO 10
C
C*** GET THE JACOBIAN
C
100 CALL ACCFES(2,IPHES,KINT,IRFF,ILCN,SBUF,LEN,IERR)
IF(IERR,NE.0) GO TO 897
ACCELC(2,IPHES,KINT,IRFF,ILCN,SBUF,LEN,IERR)
M = 1
DO 200 J=1,LEN+3
   K = J+1
   LL = J+2
   X(M) = BUF(J)
   Y(M) = BUF(K)
   Z(M) = BUF(LL)
   M = M+1
200 CONTINUE
C
DETJ = EUARIA(X,Y)
C
C*** CALCULATE SENSITIVITY VECTOR
C
DO 340 J=1,3
   CFE(I) = (PE(I) + SIGMA(J)*EPSLN(J)*DETJ)
340 CONTINUE
C
DFSIB(I) = -CFE(I)
GOTO 820
C
C*** WRITE ERROR MESSAGES TO THE SCREEN
C
801 PRINT 871, IERR
GO TO 820
805 PRINT 875, IERR
GO TO 820
807 PRINT 878, IERR
GO TO 820
809 PRINT 879, IERR
GO TO 820
C
820 CONTINUE
C
854 FORMAT(1X,I2,2X,3(E16.8,2X))
855 FORMAT(1X,'GP',5X,'SIGMAX(GP)',8X,'SIGMAY(GP)',8X, #'SIGMAXY(GP)')
860 FORMAT(1X,'GP',5X,'EPSLNX(GP)',8X,'EPSLNY(GP)',8X, #'EPSLNXY(GP)')
871 FORMAT(1X,'ACCFES KEFUNH WITH ERROR ',I4)
875 FORMAT(1X,'ACCELS RETURNED WITH ERROR ',I4)
878 FORMAT(1X, 'ACCELC RETURNED WITH ERROR', 14)
879 FORMAT(1X, 'ACCEFR RETURNED WITH ERROR', 14)

C
C
RETURN
END
SUBROUTINE DP1602(NF,J,LI,LJ,LK,LL)
CP***********************************************************************
CP* DP1602: CALCULATES THE DISPLACEMENT CONSTRAINT DESIGN
CP***********************************************************************
CP* DESCRIPTION:
CP* 'DP1602' CALCULATES THE DISPLACEMENT CONSTRAINT DESIGN
CP* SENSITIVITY VECTOR FOR A FOUR NODE PLATE BENDING
CP* ELEMENT.
CP***********************************************************************
CP* NF COUNTER FOR FINITE DIFFERENCE
CP* J EXTERNAL ELEMENT NO. BEING PROCESSED
CP* LI ORIGINAL EXTERNAL LOAD CASE NO.
CP* LJ EXTERNAL ADJOIN LOAD CASE NO.
CP* LL ORIGINAL INTERNAL LOAD CASE NO. - RETURNED VALUE
CP***********************************************************************
CP INCLUDE 'CAEGSR.INCJ IMPLIC.SPC'
CP INCLUDE 'CAEGSR.INCJ ACCIFN.MON'
CP INCLUDE 'CAEGSR.INCJ UNIL.MON'
CP INCLUDE 'CAEGSR.INCJ ELUFS.MON'
CP INCLUDE 'CAEGSR.INCJ SVECTR.MON'
CP COMMON/LCSDES/ULCS(90)
C EQUIVALENCE (NUAT(90),IDBL)
C DIMENSION SIGMA(6,4),EPSLN(6,4),BESU(100),BUF(100),CPE(500),
  X(4),Y(4),Z(4)
C DATA KT/S/,IREF/1/
C JJ = LI
C CPE(J) = 0.0
C C*** GET PROPERTIES
C CALL ACCEFR(2,JPHW,IPJAH,0,BUF,LFH,IFRR)
  IF(IERR.NE.0) GO TO 809
  PB(NF) = BUF(25)
C C*** GET INTERNAL LOAD CASE NUMBER
C 10 CALL ACCLCS(1,JPHLS,IBHL,1,0,IERR)
  IF(IERR.NE.0) GO TO 805
  CALL ACCLCS(2,JPHLSC,JJ,2,ILCS,1IERR)
  IF(IERR.NE.0) GO TO 905
  ILCN = DLCS(21)
C C*** SETUP POINTER FOR STRESS-STRAIN BUFFER
C CALL ACFECES(1,JPHESC,IMBL,1,ILCN,0,0,IERR)
  IF (IERR.NE.0) GO TO 801
  IF(JJ.EQ.LJ) LL = ILCN
C C*** GET ELEMENT STRESSES AND STRAINS
C
CALL ACCFES(2,IFNFEK,KINT,IRFF,ILNH,SBUF,LENH,IERR)
IF(IERR.NE.0) GO TO 801
M = 1
IF (JJ.EQ.L2) GO TO 55
DO 50 K=1,NSVAL
   J = M+1
   L = M+2
   SIGMA(1,K) = SBUF(M)
   SIGMA(2,K) = SBUF(J)
   SIGMA(3,K) = SBUF(L)
   M = M+6
50 CONTINUE
GO TO 65
55 M = 25
   DO 60 K=1,NSVAL
      J = M+1
      L = M+2
      EPSLN(1,K) = SBUF(M)
      EPSLN(2,K) = SBUF(J)
      EPSLN(3,K) = SBUF(L)
      M = M+6
60 CONTINUE
GO TO 100
65 IF (JJ.EQ.L2) GO TO 100
   JJ = L2
   GO TO 10
C
C*** GET THE JACOBIAN
C
100 CALL ACCLC(2,IPNELC,KINT,IREF,BUF,LENH,IERR)
IF(IERR.NE.0) GO TO 807
M = 1
DO 200 J=1,LENH,3
   K = J+1
   LL = J+2
   X(H) = BUF(J)
   Y(H) = BUF(K)
   Z(H) = BUF(LL)
   M = M+1
200 CONTINUE
C
DETJ = AREA(Q(X,Y))
DETJ = DETJ/4.0D0
C
C*** CALCULATE SENSITIVITY VECTOR
C
DO 340 J=1,3
   DO 340 K=1,NSVAL
      CFE(I) = CFE(I) + SIGMA(J,K)*EPSLN(J,K)*DETJ
340 CONTINUE
   DFRSIP(I) = -CFE(I)
C
GO TO 820
C
C*** WRITE ERROR MESSAGES TO THE SCREEN
C
801 PRINT 871, IERR
   GO TO 820
805 PRINT 875, IERR
   GO TO 820
807 PRINT 878, IERR
   GO TO 820
809    PRINT 879, IErr
       GO TO 820
C
820    CONTINUE
C
834    FORMAT(1X,'I2',2X,3(L16,8,2X))
835    FORMAT(1X,'GP',5X,'SIGMAX(GP)',8X,'SIGMAY(GP)',8X,
# 'SIGMAXY(GP)')
840    FORMAT(1X,'GP',5X,'EPSLNX(GP)',9X,'EPSLNY(GP)',9X,
# 'EPSLNXY(GP)')
871    FORMAT(1X,'ACCFES RETURNED WITH ERROR ','14)
875    FORMAT(1X,'ACCLCS RETURNED WITH ERROR ','14)
878    FORMAT(1X,'ACCELL RETURNED WITH ERROR ','14)
879    FORMAT(1X,'ACCEFR RETURNED WITH ERROR ','14)
C
C
RETURN
END
SUBROUTINE EUJDSB(PS1, SHP,F, UDSSH,F, Kt, EL)
CP******************************************************************************************
CP*
CP* EUJDSB: BEAM SHAPE FUNCTIONS
CP*
CP******************************************************************************************
CP* PSI GAUSS POINT LOCATION, THE CENTER OF THE BEAM
CP* IS ZERO.
CP* SHPF STANDARD BEAM SHAPE FUNCTIONS - RETURNED VALUE
CP* D2SHPF SECOND DERIVATIVE OF THE SHAPE FUNCTIONS
CP* - RETURNED VALUE.
CP* Kt FLAG FOR RETURNING D2SHPF
CP* =1 ONLY RETURN SHP,F
CP* =2 RETURN BOTH SHP,F AND D2SHPF
CP* EL ELEMENT LENGTH
CP*
CP******************************************************************************************
C
INCLUDE 'AEGERC.H' IMPLICIT,SPC'
C DIMENSION SHPF(12), D2SHPF(12)
C
EL2 = EL*EL
EL3 = EL2*EL
X = PSI + (EL/2)
X2 = X*X
X3 = X2*X
C
CALCULATE THE SHAPE FUNCTIONS
C
SHPF(1) = 1-X/EL
SHPF(2) = 1-(3*X2/EL2)+(2*X3/EL3)
SHPF(3) = SHPF(2)
SHPF(4) = SHPF(1)
SHPF(5) = X-(2*X2/EL)+(X3/EL2)
SHPF(6) = SHPF(5)
SHPF(7) = X/EL
SHPF(8) = (3*X2/EL2)-(2*X3/EL3)
SHPF(9) = SHPF(8)
SHPF(10) = SHPF(7)
SHPF(11) = (-X2/EL)+(X3/EL2)
SHPF(12) = SHPF(11)
C
IF(KT.EQ.1) 60 TO 900
C
CALCULATE THE SECOND DERIVATIVE OF THE SHAPE FUNCTIONS
C
D2SHPF(1) = 0.000
D2SHPF(2) = (-6/EL2)+(12*X/XL3)
D2SHPF(3) = D2SHPF(2)
D2SHPF(4) = 0.000
D2SHPF(5) = (-4/EL)+(6*X/EL2)
D2SHPF(6) = D2SHPF(5)
D2SHPF(7) = 0.000
D2SHPF(8) = (6/EL2)-(12*X/EL3)
D2SHPF(9) = D2SHPF(8)
D2SHPF(10) = 0.000
D2SHPF(11) = (-2/EL)+(6*X/EL2)
D2SHPF(12) = D2SHPF(11)
C
900  CONTINUE
   C
   C
   RETURN
   END
SUBROUTINE GETSEN(PSIB,NT,HELM,NNAME)
C*******************************************************************************
C* GETSEN: BRANCHES TO THE APPROPRIATE CONSTRAINT TYPE.
C*******************************************************************************
C* DESCRIPTION:
C* 'GETSEN' BRANCHES TO THE APPROPRIATE CONSTRAINT TYPE.
C* THE AVAILABLE CONSTRAINT TYPES ARE:
C* COMP - COMPLIANCE
C* DISP - DISPLACEMENT
C* STRESS
C*******************************************************************************
C* PSIB  THE CONSTRAINT VALUE IS RETURNED FOR FINITE
C*       DIFFERENTIAL EVALUATION
C* NT    COUNT FOR THE FINITE DIFFERENCE
C* HELM  TOTAL NUMBER OF ELEMENTS IN FINITE ELEMENT
C* MODEL, THIS IS IN IFAD DATA BASE, AND IS
C* RETRIEVED WHEN DATA BASE IS UPERLD.
C* NNAME  NAME OF FINITE ELEMENT ANALYSIS - CHARACTER
C*        INTERGER CONVERSION.
C*******************************************************************************
C INCLUDE 'CAEGSR,INCJ INPLIC,SPC'
INCLUDE 'CAEGSR,INCJ ACCIPN,MOPN'
INCLUDE 'CAEGSR,INCJ CCHIL,MOPN'
INCLUDE 'CAEGSR,INCJ KELIPN,MOPN'
INCLUDE 'CAEGSR,INCJ BVECYR,MOPN'
C
COMMON /MLMARY/ IARRY(60000)
CHARACTER NAME*8,A*4,B*4
C
DIMENSION IFNAME(2),IDBFTK(4),IERDLYV(2),PSIB(2)
C
DATA ISIZE/60000/
C
NT = NT+1
C
60 CALL ACCFNU(7,IFNAME,1,0,0,IERK)
   IF(IERR,NE,0) GO TO 800
C
C OPEN THE IFAD DATABASE
C
CALL INENF(ISIZE,IENDEV,IPNAME,1MPTH,3,ISTAT)
   IF (ISTAT,NE,0) GO TO 810
   A = CHAR(IPNAME(1))
   B = CHAR(IPNAME(2))
   NAME = A//B
C
C BRANCH TO THE APPROPRIATE CONSTRAINT ROUTINE
C
100 IF(ICT.EQ.1) CALL COMP(PSIB,NT,HELM)
   IF(ICT.EQ.2) CALL DISP(PSIB,NT,HELM)
   IF(ICT.EQ.3) CALL STRESS(PSIB,NT,HELM,NNAME)
   GO TO 820
C
C WRITE ERROR MESSAGES TO THE SCREEN
C
800 PRINT 875, IERR
   GO TO 820
810 PRINT 876, IERR
   GO TO 820
C
C CLOSE IFAD DATABASE
C
820 CALL INEXIT(IDEPTH, IERR)
   IF(IERR .NE. 0) GO TO 850
C
   GO TO 900
C
850 PRINT 877, IERR
C
900 CONTINUE
C
C FORMAT STATEMENTS
C
875 FORMAT(1X,'ACCFNW RETURNED WITH ERROR ',I4)
876 FORMAT(1X,'IENIK RETURNED WITH ERROR ',I4)
877 FORMAT(1X,'INEXIT RETURNED WITH ERROR ',I4)
1000 FORMAT(2A4)
1001 FORMAT(F8.5)
1004 FORMAT(A8)
C
C
RETURN
END
SUBROUTINE LC16(X,Y,Z,XL,YL,ZL,TM)
C************************************************************************
C* LC16: TRANSFORMS GLOBAL COORDINATES TO LOCAL COORDINATES FOR '601'
C************************************************************************
C* DESCRIPTION:
C* 'LC16' TRANSFORMS THE TRIANGULAR GLOBAL COORDINATES
C* OF ELEMENT 1601 TO LOCAL COORDINATES
C************************************************************************
C* X GLOBAL X COORDINATE
C* Y GLOBAL Y COORDINATE
C* Z GLOBAL Z COORDINATE
C* XL LOCAL X COORDINATE
C* YL LOCAL Y COORDINATE
C* ZL LOCAL Z COORDINATE
C* TM TRANSFORMATION MATRIX
C************************************************************************
C INCLUDE 'AEGSDR.INC' INPLC.SPC'
C INCLUDE 'AEGSDR.INC' ACPLN.MON'
C INCLUDE 'AEGSDR.INC' EDEFS.MO'
C
C DIMENSION X(3),Y(3),Z(3),TM(6,6),COOR(3,3),XL(3),YL(3),
C ZL(3),COOR(3,3),V12(3),V13(3),VH(3),TM(3,3)
C
C CHARACTER IREF*4
C DATA IREF/1/,IYTPE/1/
C
C*** INITIALIZE VARIABLES
C
C DO 40 I=1,3
C COOR(1,I) = X(I)
C COOR(2,I) = Y(I)
C COOR(3,I) = Z(I)
40 CONTINUE
C
C*** GET THE VECTORS PARALLEL TO THE 1-2 AND 1-3 SIDES
C
C CALL UMVEC(COOR(1,1),COOR(1,2),V12,IERR)
C CALL UMVEC(COOR(1,1),COOR(1,3),V13,IERR)
C
C OBTAIN NORMAL V12 X V13
C
C CALL UMVCRS(V12,V13,VH,0,IERR)
C
C OBTAIN LOCAL TRANSFORMATION MATRIX T13
C
C CALL EUTECS(VH,T,13,IERR)
C
C OBTAIN THE NODAL TRANSFORMATION MATRIX AS AN ASSEMBLY OF T13
C
C DO 50 K=1,3
C DO 50 J=1,3
TB(J,K) = T(J,K)
TB(J+3,K+3) = T(J,K)
50 CONTINUE
C GET LOCAL COORDINATES
C CALL UMXATB(I,COORD,COORDL,3,3)
DO 60 I=1,3
   XL(I) = COORDL(I,I)
   YL(I) = COORDL(2,I)
   ZL(I) = COORDL(3,I)
60 CONTINUE
C RETURN
END
SUBROUTINE SRST05(ILCN)
CP******************************************************************************
CP* SRST05: CALCULATES THE ADJOINT LOAD FOR THE BEAM ELEMENT
CP******************************************************************************
CP* DESCRIPTION:
CP* 'SRST05' CALCULATES THE ADJOINT LOAD FOR BENDING
CP* STRESS IN A 1-DIMENSIONAL BEAM
CP* ELEMENT AND CREATES A RESTART FILE SO THAT A RESTART
CP* OF THE FINITE ELEMENT MODEL CAN BE MADE. THE
CP* RESULTING DISPLACEMENTS CAN THEN BE USED TO CALCULATE
CP* THE STRESS CONSTRAINT DESIGN SENSITIVITY.
CP* THIS ROUTINE ONLY WORKS FOR BEAMS LYING IN THE
CP* X-GLOBAL OR Y-GLOBAL PLANE.
CP******************************************************************************
CP ILCN INTERNAL LOAD CASE NO. IF ORIGINAL LOAD
CP******************************************************************************
C INCLUDE 'TAEGRD1,INC' IMPLIC.SPL'
INCLUDED 'TAEGRD1,INC' ACCIPN,MON'
 INCLUDED 'TAEGRD1,INC' CNIT,MON'
 INCLUDED 'TAEGRD1,INC' ELEDES,MON'
 INCLUDED 'TAEGRD1,INC' SVCTR,MON'
C DIMENSION X(2),Y(2),Z(2),DQHFF(12),HATN(50),HUF(100),
* QHFF(12),CFBUF(6),GFLW(3),R1GF(12),CD(2,6),W'IW(J),
* IDIR(2),AL(12)
C DATA GFLW/-,77459667341403D0, .0D0,
* ,77459667341404D0/
DATA W1W/,55555555555555556D0, .588888888888888D0,
* ,55555555555555556D0/
DATA KT/2,1REF/1,MP/1/
C INITIALIZE VARIABLES
C ND = NDOF*NUMP
DO 10 J=1,ND
   AL(J) = 0.0
10 CONTINUE
C GET DEPTH AND WIDTH OF THE BEAM
C CALL ACCEFR(2,IPNUM,IPTR,0,HUF,LEN,IERR)
  IF(IERR,NE,0) GO TO 809
  H = 2*BUF(9)
  B = 2*BUF(10)
C GET MODULUS OF ELASTICITY
C CALL ACCMAT(2,IPNUM,NMAT,MP,1,HUF,LEN,IERR)
  IF(IERR,NE,0) GO TO 808
  E = BUF(5)
C GET X, Y, AND Z OF ELEMENT NODES
CALL ACCEL(2,IFMELC,KINT,IREF,_BUF,LEN,IERR)
IF(IERR.NE.0) GO TO 807
M = 1
DO 200 J=1,6+3
K = J+1
L = J+2
X(M) = BUF(J)
Y(M) = BUF(K)
Z(M) = BUF(L)
M = M+1
200 CONTINUE
C
C CALCULATE ELEMENT LENGTH
C
DX = X(2)-X(1)
DY = Y(2)-Y(1)
DZ = Z(2)-Z(1)
EL = DSRGT(DX*DX+DY*DY+DZ*DZ)
C
XMP = 1.0/EL.
C
DO 300 K=1,3
PSI = GPLW(K)
CALL EU3DSB(PSI,SHFP,DSHFP,EL)
C
C CALCULATE ADJOINT LOAD
C
ALGF(3) = -.5D*H*E*XH*DSHF(3)
ALGF(9) = -.5D*H*E*XH*DSHF(9)
IF(DX.EQ.0,ANU,DZ,ER,0) GO TO 250
ALGF(4) = 0.000
ALGF(5) = -.5D*H*E*XH*DSHF(5)
ALGF(10) = 0.000
ALGF(11) = -.5D*H*E*XH*DSHF(11)
GO TO 255
250 ALGF(4) = -.5D*H*E*XH*DSHF(5)
ALGF(5) = 0.000
ALGF(10) = -.5D*H*E*XH*DSHF(11)
ALGF(11) = 0.000
C
C SUM ADJOINT LOAD OVER GAUSS POINTS
C
255 DO 260 J=1,ND
AL(J) = AL(J)+ALGF(J)
260 CONTINUE
300 CONTINUE
C
N = -1
DO 400 J=1,MUHF
DO 350 K=1,MDF
IF(AL(K+J*N).EQ.0.0) GO TO 350
WRITE(11,864) IEXTNN(J),K,AL(K+J*N)
350 CONTINUE
N = N+5
400 CONTINUE
C
C WRITE ERROR MESSAGES TO THE SCREEN
C 802  PRINT 872, IERR
     GO TO 820
807  PRINT 878, IERR
     GO TO 820
808  PRINT 879, IERR
     GO TO 820
809  PRINT 876, IERR
     GO TO 820
C
C 820  CONTINUE
C
C 860  FORMAT(1X,'GP='I12.4X,'W='F11.5,4X,'WXX='E11.5,4X,'AH='*,
     *E11.5,4X,'AWXX='E11.5)
864  FORMAT(2(I4,2X),F16.4)
872  FORMAT(1X,'ACCCNL RETURNED WITH ERROR ',14)
876  FORMAT(1X,'ACCEPR RETURNED WITH ERROR ',14)
878  FORMAT(1X,'ACCELG RETURNED WITH ERROR ',14)
879  FORMAT(1X,'ACCMAT RETURNED WITH ERROR ',14)
1001 FORMAT(E12.5)
C
C
RETURN
END
SUBROUTINE SRST11(ILCN,IE)

C**************************************************************************
C SRST11: CALCULATES THE ADJOINT LOADS FOR THE PLANE STRESS ELEMENT
C**************************************************************************
C DESCRIPTION:
C
C SRST11' CALCULATES THE ADJOINT LOADS FOR A FOUR AND
C EIGHT NODE PLANE STRESS ELEMENT AND THEN CREATES A
C RESTART FILE SO THAT THE FINITE ELEMENT MODEL CAN BE
C RESTARTED SUCH THAT THE RESULTING STRESSES AND STRAINS
C CAN BE THEN USED TO CALCULATE THE STRESS CONSTRAINT
C DESIGN SENSITIVITY.
C
C**************************************************************************
C
C ILCN  INTERNAL LOAD CASE NO. OF ORIGINAL LOAD
C IE  EXTERNAL ELEMENT NO. CONSTRAINED
C**************************************************************************
C
C INCLUDE ' [AEGSDR.INC] IMPLICIT, SFC'
C INCLUDE ' [AEGSDR.INC] ACCIPN.MON'
C INCLUDE ' [AEGSDR.INC] LINL.MON'
C INCLUDE ' [AEGSDR.INC] EI, DES.MON'
C INCLUDE ' [AEGSDR.INC] VECTR, MON'
C
C DIMENSION SHF(8), GFL(2,4), SBUF(50), BUF(100),
C * DSHPGX(8), USHPGY(8), DSHPL(2,8) X(8), Y(8), Z(8),
C * DG(3), ALGF(16), ALF(16), E(3,3), D(9), C(3), B(3,16),
C * EM1(1)
C
C DATA GFL/2*-57735027, -57735027, -57735027,
C * 2*, 57735027, 57735027, 57735027/
C DATA KT/3/, IREF/1/, KPT/1/, KTYPE/1/
C
C** INITIALIZE VARIABLES
C
C ND = NUNPE*NUOF
C DO 10 J=1,NU
C 10 AL(J) = 0,
C IF(IE.GT.1) GO TO 15
C
C** GET X AND Y FOR JACOBIAN EVALUATION
C
C 15 CALL ACCELE(2,1PNE, KINF, IREF, NU, LFIN, IERR)
C IF(IERR.NE.0) GO TO 807
C M = 1
C DO 20 L=1,LENB;3
C J = L+1
C K = L+2
C X(M) = BUF(L)
C Y(M) = BUF(J)
C Z(M) = BUF(K)
C M = M+1
C 20 CONTINUE
C AREA = AREA(X,Y)
C XMP = 1.DO/AREA
C
C** CALCULATE ADJOINT LOAD
C C*** LOOP OVER GAUSS POINTS
C
IC = 0
M = 1
DO 80 KK=1,NSVAL
J = M+1
L = M+3
C C*** GET DERIVATIVES OF STRESS FUNCTION C
C C*** VON MISES: (=(SIGX**2+SIGY**2-SIGX*SIGY+3*SIGX**2)**.5
C
IF(IC.GT.0) GO TO 25
CALL ACCFES(2,IPNFS,KINT,IREF,ILIN,LET,N+1,LEL,1,IER)
IF(IERR.NE.0) GO TO 801
25 IF(IST.EQ.1) GO TO 30
VMS = (SBUF(M)**2+SBUF(J)**2-SBUF(N)*SBUF(J)+3)*
   SBUF(L)**2)**.5
DG(1) = .5*(2*SBUF(N)-SBUF(J))/VMS
DG(2) = .5*(2*SBUF(J)-SBUF(M))/VMS
DG(3) = (3*SBUF(L))/VMS
GO TO 40
C C*** PRINCIPAL STRESS
C 30 TMAX = (((.5*(SBUF(M)-SBUF(J)))**2+SBUF(L)**2)**.5
DG(1) = .5+.25*(SBUF(M)-SBUF(J))*(1/TMAX)
DG(2) = .5-.25*(SBUF(M)-SBUF(J))*(1/TMAX)
DG(3) = SBUF(L)*(1/TMAX)
C 40 M = M+4
IC = IC+1
PSI = GPL(1,KK)
ETA = GPL(2,KK)
IF(ISTYP.EQ.2) CALL EU2DLQ(PSI,FTA,KF,SHF,DSHFL, *
   DSHFX,DSHFY,NET,JX,Y,IERR)
   * IF(ISTYP.EQ.4) CALL EU2DFQ(PSI,ITA,KF,SHF,DSHPL, *
   DSHFX,DSHFY,NET,JX,Y,IERR)
   IF(IERR.NE.0) GO TO 809
C C*** GET ELASTICITY MATRIX E
C CALL EU2DSS(D,THP,EMT,NMAT,ITYMA1,ITYPE)
N = 1
DO 50 JJ=1,3
LL = N+1
LLL = N+2
E(JJ,1) = W(N)
E(JJ,2) = W(LL)
E(JJ,3) = D(LLL)
N = N+3
50 CONTINUE
C C*** GET STRAIN-DISPLACEMENT MATRIX H
C CALL SD11(B,DSHFX,DSHFY,NUNPE)
C C*** CALCULATE [DQ]*[E]*[D]*[H]
CALL UMXAB(DG;E,C+1,J3)
CALL UMXAB(C;F,ALGP;1,HU,3)
DO 60 JJ=1,NU
   ALGP(JJ) = ALGP(JJ) * XHP
C
C*** SUM ADJOIN LOAD OVER GAUSS POINTS (INTEGRATE OVER ELEM)
C
DO 70 JJ=1,NU
   AL(JJ) = AL(JJ) + ALGP(JJ) * DETJ
70    CONTINUE
C
C*** WRITE ADJOIN LOAD TO RESTART FILE
C
N = -1
DO 95 J=1,NUNP,
   95 CONTINUE
C
C WRITE ERROR MESSAGES TO THE SCREEN
C
GO TO 820
C
801   PRINT 871, IERR
   GO TO 820
807   PRINT 878, IERR
   GO TO 820
809   PRINT 876, IERR
   GO TO 820
C
C
820   CONTINUE
C
C
GO TO 820
C
871   FORMAT(1X,'ACCES RETURNED WITH ERROR ',(I4)
876   FORMAT(1X,'SHAPE FUNCTION ROUTINE RETURNED WITH ERROR ',
   878   FORMAT(1X,'ACCEL RETURNED WITH ERROR ',(I4)
2001   FORMAT(A)
2004   FORMAT(1X,I4,1X,J2,1X,E16.9)
C
C
RETURN
END
SUBROUTINE SRST16(ILCN, IE)

CP**************************************************************
CP SRST16: CALCULATES THE ADJOINT LOADS FOR A TRIANGULAR
CP PLATE BENDING ELEMENT AND CREATES A POST-PROCESSING FILE
CP* THAT THE FINITE ELEMENT MODEL CAN BE POSTPROCESSED IN THE
CP RESULTING STRESSES AND STRAINS ARE THEN USED IN THE
CP STRESS CONVERGENCE DESIGN SENSITIVITY CALCULATION IN
CP THE 'ST1601' SUBROUTINE. STRESS TYPES 1=PRINCIPAL
CP 2 = VON MISES
CP**************************************************************
CP ILCN  INTERNAL LOAD CASE NO. OF THE ORIGINAL LOAD
CP* IE EXTERNAL (CONSTRAINED) ELEMENT NO.
CP**************************************************************

C INCLUDE 'CAEGSDR.INC' IMPLICIT.FPC'
INCLUDE 'CAEGSDR.INC' ACCIPN.MON'
INCLUDE 'CAEGSDR.INC' CNIL.MON'
INCLUDE 'CAEGSDR.INC' ELEM.S몬'
INCLUDE 'CAEGSDR.INC' SPECTR.MON'

C DIMENSION X(3), Y(3), Z(3), BUF(50), AL(18), SBUF(50), HG(3),
C* CG(3), XL(3), YL(3), ZL(3), L(3), M(9), E(7), E(3,3),
C* TB(6,6), SIG(3,3), ALM(18)

C DATA IREF/1/, ITYPE/1/

C DO 10 J=1,18
AL(J) = 0.000
10 CONTINUE
C
C*** GET X, Y AND Z
C
15 CALL ACCELL(2, PNEIC, KIN1, IREF, BUF, LENB), ENR)
IF (IERR .NE. 0) GO TO 807
M = 1
DO 20 L=1, LENB, 3
J = L+1
K = L+2
X(M) = BUF(J)
Y(M) = BUF(K)
Z(M) = BUF(L)
M = M+1
20 CONTINUE
C
C*** GET LOCAL COORDINATE SYSTEM
C CALL LC16(X,Y,Z,XL,YL,ZL,18)
C
C*** GET ELASTICITY MATRIX [E]
CALL EU2DS(D,THP,EMT,HMAT,TYMAT,ITYPE)
N=1
DO 25 JJ=1,3
  E(JJ,1) = D(N)
  E(JJ,2) = D(N+1)
  E(JJ,3) = D(N+2)
N = N+3
25 CONTINUE

C GET THE THICKNESS OF THE PLATE
C CALL ACCEPR(2,IFNEP,IPAR,0,BUF,LEN,IERR)
IF(IERR.NE.0) GO TO 809
THK = BUF(25)

C** GET STRESSES AT THE MUSCLE NODES OF TRIANGLE
C CALL SSMDV(XL,YL,ILCN,SIG)

C** CALCULATE THE ADJOINT LOAD WHERE O=[DG][E][B][H]*T/2
C** AL(18) = (TAREA/3)*(O(0,.5,.5)+O(.5,0,.5)+O(.5,.5,0)
C TAREA = EUTRIAX(X,Y)
DO 80 IT=1,3

C*** CALCULATE [DG] THE DERIVATIVE VECTOR
C IF(IST,3,1) GO TO 30
C
C*** VON MISES: G=SQRT(SIGX**2+SIGY**2-SIGX*SIGY+SIGXY**2)
C VMS = UDSRT(SIG(IT,1)**2+SIG(IT,2)**2+SIG(IT,3)**2* 
  SIG(IT,3)**2)
DG(1) = .5D0*(2.D0*SIG(IT,1)-SIG(IT,2))/VMS
DG(2) = .5D0*(2.D0*SIG(IT,2)-SIG(IT,1))/VMS
DG(3) = (3.D0*SIG(IT,3))/VMS
GO TO 40

C*** PRINCIPAL STRESS
C 30 Tmax = DSQRT((.5D0*(SIG(IT,1)-SIG(IT,2)))**2+SIG(IT,3)**2)
DG(1) = .5D0+.25D0*(SIG(IT,1)-SIG(IT,2))/TMAX
DG(2) = .5D0-.25D0*(SIG(IT,2)-SIG(IT,1))/TMAX
DG(3) = SIG(IT,3)/TMAX

C*** CALCULATE [C] = [DG] * [E]
C CALL UMXAR(UG,E,C,1,3,3)

C*** CALCULATE [AL] = [C] * [B] * T/2 * XMP
C CALL AL16(XL,YL,C,ALM,THK,IT)
DO 50 K=1,18
  AL(K) = AL(K) + (TAREA/3.0D0)*ALM(K)
50 CONTINUE

C CONTINUE

C*** WRITE ADJOINT LOAD TO RESTART FILE
C N = -1
DO 95 J=1,NUMPE

DO 90 K=1,NUDF
   IF(AL(K+J+N),FG,0.0) GO TO 90
   WRITE(11,2004) IEXTHN(J),K,AL(K+J+N)
90    CONTINUE
      N = N+5
95    CONTINUE
C GO TO 820
C WRITE ERROR MESSAGES TO THE SCREEN
C 801 PRINT 871, IERR
     GO TO 820
807 PRINT 878, IERR
     GO TO 820
809 PRINT 879, IERR
     GO TO 820
C C 820 CONTINUE
C C 871 FORMAT(1X,'ACCFES RETURNED WITH ERROR ',I4)
878 FORMAT(1X,'ACCEL RETURNED WITH ERROR ',I4)
879 FORMAT(1X,'ACCEFK RETURNED WITH ERROR ',I4)
2001 FORMAT(A)
2002 FORMAT(1X,I4,2X,'DISP=',E16.9)
2004 FORMAT(1X,I4,1X,I2,1X,E16.9)
C C RETURN
END
SUBROUTINE SMSN16(X,Y,LLN,SLC)
C**********************************************************************
C# SSMD16: CALCULATES STRESSES AT THE MIDSIDE NODES OF A 1XI.
C#
C# DESCRIPTION:
C#
C# 'SSMD16' CALCULATES STRESSES AT THE MIDSIDE NODES OF A
C# TRIANGLE 1601. [SIG] = [LE]*[B]*[DIS]*[I]/2. I IS THE
C# FINITE ELEMENT FORMULA FOR CALCULATING THE STRESS.
C# THE STRAIN-DISPLACEMENT MATRIX [B] IS LOCATED IN
C# COLUMNS 4,5, & 6 OF THE [W] MATRIX. LE IS THE
C# ELASTICITY MATRIX, 3X3 FOR ISOTROPIC MATERIAL, AND
C# [DIS] IS THE DISPLACEMENT VECTOR, WHERE THE DISPLS.
C# ARE TAKEN AT THE NODES OF THE TRIANGLE. BECAUSE
C# IFAD CALCULATES THE STRESS RESULTANTS, THE I/2 1XI
C# MUST BE INCLUDED IN THIS CALCULATION.
C#
C**********************************************************************
C#
C# X THE LOCAL ELEMENT X COORDINATE
C# Y THE LOCAL ELEMENT Y COORDINATE
C# ILCH THE INTERNAL I/2 CASE NUMBER FOR THE (INTERNAL)
C# SIG THE MIDSIDE NODE STRESSES - RETURN VALUE
C#
C**********************************************************************
C
INCLUDE 'LAESDR.INC' IMPLICIT,SPC'
INCLUDE 'LAESDR.INC' ACCIPN.MON'
INCLUDE 'LAESDR.INC' CNFL.NON'
INCLUDE 'LAESDR.INC' ELDES.MON'
C
EQUIVALENCE (NUAT(14),IFR)
C
DIMENSION X(3),Y(3),GPTS(3,3),M(18,7),CFBU(7),CN(3,7),
& DIS(9),D(9),L(3,3),EMT(27),HUB(100),LB(3,9),
& SIG(3,3),XL(6),YL(6)
C
DATA GPTS/0.000,5.00,5.00,5.00,0.000,5.00,5.00,5.00,0.000/
DATA ITYPE/1/
C
C** GET DISPLACEMENTS
C
DO 10 J=1,NUPHE
  CALL ACCCMN(2,IPNCHU,INTNH(J),1,LLN,CFBU,LEN,IERR)
  IF(IERR.NE.0) GO TO 802
  DO 10 K=1,LEN
  CD(J,K) = (CFBU(K)
10 CONTINUE
M = 1
DO 20 J=1,NUPHE
  DIS(M) = CD(J,3)
  DIS(M+1) = CD(J,4)
  DIS(M+2) = CD(J,5)
  M = M+3
20 CONTINUE
C
C** GET ELASTICITY MATRIX
CALL EU2DSS(UNIT, EMAT, NMAT, IYMAT, ITYPE)
N = 1
DO 30 JJ = 1, 3
E(JJ, 1) = D(N)
E(JJ, 2) = D(N + 1)
E(JJ, 3) = D(N + 2)
N = N + 3
30 CONTINUE
** GET MATERIAL THICKNESS
**
CALL ACCEPR(2, PNLPR, IPTAB, BUF, N, IERR)
IF (IERR .NE. 0) GO TO 809
THK = BUF(25)
** GET LOCAL X AND Y COORDINATES
**
CALL MOVESP(XL, X, 3*IPR)
CALL MOVESP(YL, Y, 3*IPR)
** CALCULATE STRESSES AT THE MIDSIDE NODES
**
DO 100 INT = 1, 3
** GET SHAPE FUNCTION AT MIDSIDE NODE
**
CALL SF1501(XL, YL, GFTS(1, INT), W, EB)
** [SIG] = [E]*[D]*[DIS]*T/2
**
DO 50 M = 1, 9
DO 50 K = 1, 3
GASH = 0.0D0
DO 40 J = 1, 3
GASH = GASH + I(K, J)*W(M, J)
40 CONTINUE
EB(K, M) = GASH
50 CONTINUE
DO 60 I1 = 1, 3
SIG(INR, I1) = 0.0D0
DO 60 M2 = 1, 9
SIG(INR, I1) = SIG(INR, I1) + EB(I1, M2) * W1S(M2)
60 CONTINUE
DO 70 I2 = 1, 3
SIG(INR, I2) = SIG(INR, I2) * THK / 2.0D0
70 CONTINUE
100 CONTINUE
GO TO 820
** WRITE ERROR MESSAGES TO THE SCREEN
**
802 PRINT 872, IERR
GO TO 820
809 PRINT 779, IERR
GO TO 820
**
820 CONTINUE
C
872 FORMAT(IX,'ACCEPR RETURNED WITH ERROR ',/4)
879 FORMAT(IX,'ACCEPR RETURNED WITH ERROR ',/4)
C
RETURN
END
SUBROUTINE SNMD16(X,Y,ILCN,EPN)
CP**********************************************************
CP* SNMD16: CALCULATES STRAINS AT THE MIDSIDE NODES OF A TRIANGLE 1601.
CP* DESCRIPTION:
CP* 'SNMD16' CALCULATES STRAINSES AT THE MIDSIDE NODES OF TRIANGLE 1601.
CP* THE FORMULAS FOR STRAIN ARE:
CP* EX = 1/E*(SIGX-V*SIGY)
CP* EY = 1/E*(SIGY-V*SIGX)
CP* EXY = SIGUX/Y
CP**********************************************************
CP* X  THE LOCAL ELEMENT X COORDINATE
CP* Y  THE LOCAL ELEMENT Y COORDINATE
CP* ILCN THE INTERNAL LOAD CASE NUMBER FOR THE ADJOINT LOAD
CP* EPN  THE MIDSIDE NODE STRAINS - RETURN VALUE
CP**********************************************************
C
INCLUDE 'CAEGSDR. INC' IMPLICIT,SPC
INCLUDE 'CAEGSDR. INC' ACC(IFN,MON'
INCLUDE 'CAEGSDR. INC' CNTL,MON'
INCLUDE 'CAEGSDR. INC' ELEVES,MON'

DIMENSION X(3),Y(3),GF15(3,3),W(18,7),CFBUF(7),LD(3,7),
*  DIS(9),EM1(27),BUF(100),EPN(3,3),SIG(3,3)

DATA MPT/1/

C** GET CONSTANS
C
CALL ACCHMAT(2,IPNMA,T,MP1,BUF,LEN,IERR)
IF(IERR.NE.0) GO TO 808
E = BUF(5)
V = BUF(7)
G = E/(2.DO*(1.DO+V))

C** GET STRESSES AT THE MIDSIDE NODES
C
CALL SNMD16(X,Y,ILCN,SIG)

C** CALCULATE STRAINS AT THE MIDSIDE NODES
C
DO 100 IT=1,3
  EPN(IT,1) = 1/E*(SIG(IT,1)-V*SIG(IT,2))
  EPN(IT,2) = 1/E*(SIG(IT,2)-V*SIG(IT,1))
  EPN(IT,3) = SIG(IT,3)/G
100 CONTINUE

GO TO 820

C
C** WRITE ERROR MESSAGES TO THE SCREEN
C
808 PRINT 878, IERRR
GO TO 820

820 CONTINUE

878 FORMAT(1X,'ACCMAT RETURNED WITH ERROR ',14)

RETURN
END
SUBROUTINE STRES05(NF,I,L,PSI00)
C*********************************************************************************************
CP
CP STRES05: CALC. STRESS CONSTRAINT AND SENSIT. FOR A BEAM
CP
CP*********************************************************************************************
CP
CP DESCRIPTION:
CP
CP 'STRES05' CALCULATES THE STRESS CONSTRAINT AND THE
CP DESIGN SENSITIVITY FOR THE 1-D BEAM ELEMENT IN
CP BENDING. INCLUDES TORSION.
CP
CP*********************************************************************************************
CP
CP NT  COUNTER FOR INFINITE DIFFERENCE
CP I  EXTERNAL ELEMENT NO. FOR ELEMENT BEING PROCESSED
CP L  LOAD CASES = EXTERNAL
CP PSI00 STRESS CONSTRAINT - RETURNED VALUE
CP
CP*********************************************************************************************
C
C INCLUDE 'CAEGSR,INCJ IMPLC.YC'
C INCLUDE 'CAEGSR,HINC ACIFN,MON'
C INCLUDE 'CAEGSR,INCJ CNTL,MON'
C INCLUDE 'CAEGSR,INCJ EULPES,MON'
C INCLUDE 'CAEGSR,INCJ KULCTR,MON'
C COMMON/ELSECL/ULCS(90)
C
C EQUIVALENCE (NUAT(14),IF),(NDAT(98),)NBL)
C
C DIMENSION X(3),Y(3),Z(3),DDSHF(12),WATN(50),1(2),BUFF(100),
C SHPF(12),CFBUFF(6),GFLW(3),ALD(2,6),CD(2,6),W1W(3),
C PSI00(600),K(6,2),B(6,2),1(3,3),TB(6,6),CDB(2,6),
C ALDL(2,6),CUOR(3,3)
C
C DATA GFLW/.745966692414183D0, .000,
C     .745966692414183D0/
C DATA W1W/ .555555555555555D0, .888888888888889D0,
C     .555555555555555D0/
C DATA KT/3,/IREF/1,/HPT/1/
C
C DPS100 = 0.000
C DPS100 = 0.000
C PSI00(I) = 0.000
C IF(I.GT.1) GO TO 50
C
C GET AREA MOMENT OF INERTIA ABOUT Y-AXIS
C
C 50  CALL ACCEPR(2,IPNEPR,IPTPR,0,BUF*LEN,IERR)
C IF(IERR.NE.0) GO TO 809
C I = BUF(5)
C H = 2*BUF(9)
C B = 2*BUF(10)
C W(N) = H
C I(N) = B
C
C GET WEIGHT DENSITY AND MODULUS OF ELASTICITY
C
C CALL ACCMAT(2,IPNMAT,IPMAT,0,BUF*LEN,IERR)
C IF(IERR.NE.0) GO TO 808
C
GAMMA = 0.0DO
E = 30500000.D0
Y = .3DO
G = E/(2.DO*(1.DO+Y))

C GET INTERNAL LOAD CASE NUMBER FOR ORIGINAL LOAD
60 CALL ACCLCS(1,PNLCS,IDBL,1,0,IERR)
   IF(IERR.NE.0) GO TO 805
CALL ACCLCS(1,PNLCS,L(1),2,DLCS,IERR)
   IF(IERR.NE.0) GO TO 805
IL1 = DLCS(21)

C GET DISPLACEMENTS AT ELEMENT ENDS
DO 70 J=1,NUN
   CALL ACCNCU(1,PNCHU,INNM(J),1,IL1,CBUF,LEN,IERR)
   IF(IERR.NE.0) GO TO 802
    DO 70 K=1,NUNF
   CD(J,K) = CBUF(K)
70 CONTINUE

C BYPASS ADJOINT LOAD CALCULATION
IF(NT.GT.1) GO TO 150

C GET INTERNAL LOAD CASE NUMBER FOR ADJOINT LOAD
C CALL ACCLCS(1,PNLCS,IDBL,1,0,IERR)
   IF(IERR.NE.0) GO TO 805
CALL ACCLCS(2,PNLCS,L(2),2,DLCS,IERR)
   IF(IERR.NE.0) GO TO 805
ILCN = DLCS(21)

C GET DISPLACEMENTS AT ELEMENT ENDS FOR ADJOINT LOAD
C CALL ACCNCU(1,PNCHU,IDBL,1,ILCN,0,0,IERR)
    DO 100 J=1,NUN
    CALL ACCNCU(1,PNCHU,INNM(J),1,ILCN,CBUF,LEN,IERR)
   IF(IERR.NE.0) GO TO 802
    DO 100 K=1,NUNF
   ALD(J,K) = CBUF(K)
100 CONTINUE

C*****************************************************************************
C EVALUATE DISPLS. AND CURVATURE AT THE GAUSS POINTS USING SHAPE FUNCTIONS -
C ONE PT. FOR CURV., THREE PT. FOR V'SPL
C*****************************************************************************
C GET X, Y, AND Z OF ELEMENT NODES
C 150 CALL ACCEL(2,PNELC,KINT,IREF,BUF,LEN,IERR)
   IF(IERR.NE.0) GO TO 807
   M = 1
    DO 200 J=1,3
    X(M) = BUF(J)
    Y(M) = BUF(K)
    Z(M) = BUF(LL)
H = H+1
CONTINUE
DO 210 J=1,3
    COOR(1,J) = X(J)
    COOR(2,J) = Y(J)
    COOR(3,J) = Z(J)
210    CONTINUE
C FORM THE ELEMENT LOCAL COORDINATE SYSTEM FOR DISPLACEMENTS
C
IN3 = INTNIN(3)
/ CALL EURTH(IN3,BETA,CONR,T,IFRR)
CALL ZEROSF(IB,36*IP)
DO 220 J=1,3
    DO 220 K=1,3
        TB(J,K) = T(J,K)
        TB(J+3,K+3) = TB(J,K)
220    CONTINUE
CALL UMXPRT(IB,CD,C,6,2,6)
CALL UMXPRT(IB,ALD,D,6,2,6)
DO 230 J=1,2
    DO 230 K=1,6
        CDL(J,K) = U(J,K)
        ALDL(J,K) = U(K,J)
230    CONTINUE
C CALCULATE ELEMENT LENGTH
C
DX = X(2)-X(1)
DY = Y(2)-Y(1)
DZ = Z(2)-Z(1)
EL = DBRST(DX*DX+DY*DY+DZ*DZ)
C CHANGE LOCAL Y-ROTATION FROM POSITIVE TO NEGATIVE IF
C BEAM LIES ALONG THE X GLOBAL AXIS
C
IF(DX,LT.0.001,AMNN,DY,GT.,-0.001) GO TO 245
DO 240 J=1,HUNPE
    CDL(J,4) = -CDL(J,4)
    CDL(J,5) = -CDL(J,5)
    ALDL(J,3) = -ALDL(J,3)
240    CONTINUE
C CALCULATE THE TWISTING ANGLES
C
WXY = DBRTS((CDL(2,4)-CDL(1,4))/EL)
AWXY = DBRTS((ALDL(2,4)-ALDL(1,4))/EL)
C EVALUATE SHAPE FUNCTIONS FOR HISP = THREE POINT QUADRATURE
C
B2 = B*8
B3 = B2*8
B4 = B2*B2
H2 = H*H
H3 = H2*H
C
DO 300 K=1,3
    PSI = OPLW(K)
    CALL EU3DSB(FS,SHFF,INDSHFF,2,EL)
    W = (SHFF(5)*CDL(1,3)+SHFF(5)*CDL(1,5)+SHFF(9)*
         CDL(2,3)+SHFF(11)*CDL(2,5))
I: WRITE ERROR MESSAGES

C

EVALUATE SHAPE FUNCTIONS FOR CURV. - THREE POINT QUADRATURE

C

WXX = (DDSHFF(3)*CDL(1,3)+UNSHF(5)*CDL(1,5)+

DSSHF(7)*CDL(2,3)+UNSHF(11)*CDL(2,5))

C

AWXX = (DSSHF(3)*ALDL(1,3)+UNSHF(5)*ALDL(1,5)+

DSSHF(7)*ALDL(2,3)+DSSHF(11)*ALDL(2,5))

C

CALCULATE SENSITIVITY VECTORS

C

XMP = 1.0D/EL

IF(N,GT,1) GO TO 250

STERM = .5D0*XMP*WXX

PJB = H5/3.0, -420*H5*H5/(12*D0*H5)

PFH = B*H5, -420*B2*H5/(12*D0*H5)

DSIMGH = DSIN64 + (-GAMMA*H3*A-H3/12)*WXX*WXX-

* PJB*XWXY*AWXY*WTW(K)*(EL/2,1,1)

C

DPSIHG = DSIMGH + (-GAMMA*H3*A-W-3.0D0*H5/12)*WXX*WXX

C

* WXX = PJB*XWXY*AWXY*WTW(K)*(EL/2,1,1)

C

IF(N,GT,1) GO TO 250

247 DPSIHG = DSIMGH + (-GAMMA*H3*A-W-3.0D0*H5/12)*WXX

C

* WXX = PJB*XWXY*AWXY*WTW(K)*(EL/2,1,1)

C

IF(N,GT,1) GO TO 300

C

CALCULATE PSI(i) - INTEGRAL OF STRESS FUNCTION FOR ELEMENT

C

STRESS = .5D0*W*W*XMP

PSIRB(i) = PSIRB(i) + STRESS*WTW(K)*(EL/2,1,1)

C

CONTINUE

IF(N,GT,1) GO TO 820

C

DPSIB(i) = DPSIB(i)

DPSIH(i) = DPSIHG

C

GO TO 820

C

WRITE ERROR MESSAGES TO THE SCRFEN

C

IERR GO TO 820

C

IERR GO TO 820

C

IERR GO TO 820

C

CONTINUE

C

FORMAT(1X,'***ADJOIN: LOAD IS APPLIED AT ELEMENT',I4)

C

FORMAT(/,1X,'BEAM WIDTH B=',F8.5,'X','BEAM DEPTH=',F8.5,'X'

*','E1=',E9.3,'X','E2=',E9.3,'X','GAMMA=',F6.5,'APPLIED FORCE

*=',F8.5)

C

FORMAT(1X,'NODE=',I2,2X,'X=',E12.5,2X,'Y=',E12.5,2X,'Z=',E12.5,2X)
*E12.5,2X,'RX=',*E12.5,2X,'KY=',*E12.5,2X,'RZ=',*E12.5)
860 FORMAT(1X,'GF=',*E12.5,4X,'W=',*E11.5,4X,'WXX=',*E11.5,4X,'AH=',*E11.5,4X,'AWX=',*E11.5)
864 FORMAT(1X,'NODE=',*E12.5,2X,'AX=',*E12.5,2X,'AY=',*E12.5,2X,'AZ=',*E12.5,2X,'ARX=',*E12.5,2X,'ARY=',*E12.5,2X,'ARZ=',*E12.5)
872 FORMAT(1X,'ACCCND RETURNED WITH ERROR ',*E12.5)
875 FORMAT(1X,'ACULCS RETURNED WITH ERROR ',*E12.5)
876 FORMAT(1X,'ALCEF R RETURNED WITH ERROR ',*E12.5)
878 FORMAT(1X,'ACELC RETURNED WITH ERROR ',*E12.5)
879 FORMAT(1X,'ACLMAT RETURNED WITH ERROR ',*E12.5)
1001 FORMAT(*E12.5)
1004 FORMAT(*I4)
C
C
RETURN
END
SUBROUTINE STRESI1(N1,I,L,FSIBT)

CP******************************************************************************************
CP* STRESI1: CALC. STRESS CONSTRAINT AND SENSIT. - PLANE STRESS
CP******************************************************************************************
CP* DESCRIPTION:
CP* 'STRESI1' CALCULATES THE STRESS CONSTRAINT AND THE
CP* DESIGN SENSITIVITY FOR A FOUR OR AND EIGHT HOLE PLANE
CP* STRESS ELEMENT WITH FRACTION, SELF WEIGHT NOT INCLUDED
CP******************************************************************************************
CP*
CP* N1 COUNTER FOR FINITE DIFFERENCE
CP* I EXTERNAL ELEMENT NO. BEING PROCESSED
CP* L EXTERNAL LOAD CASE NO.
CP* FSIBT STRESS CONSTRAINT
CP******************************************************************************************
C
INCLUDE 'AEGSDR.INC IMA',LIC','SFC','
INCLUDE 'AEGSDR.INC ACCIPN.MON'
INCLUDE 'AEGSDR.INC CNTL_MON'
INCLUDE 'AEGSDR.INC ELES_MON'
INCLUDE 'AEGSDR.INC SVECTR.MON'
COMMON/LCSUES/LCLS(99)

C
EQUIVALENCE (NDAT(98),IDBL)

C
DIMENSION X(8),Y(8),Z(8),SHFF(8),(PL(2),L(2),ILCN(2),
* DATN(50),BUF(100),PSIBT(500),SBUF(50),CBUF(6),
* BF(G+4),(EM(3),USHFX(8),BSBUF(50),H00X(4),Y(K),
* DSHFL(2),SIGMA(4),EPSLN(4),F(I,J),SF(500),
* DG(3),ALGF(16),AL(16),E(3),U(Y,L)(D),PSIBPE(50)
CHARACTER YESNO1

C
DATA GPL/24.57735027, .57735027,-.5/3.027,
* 24.57735027,-.57735027,.57735027/
DATA KI/3/,IREF/1/,HP1/1/,RTYPE/1/

C
SE(I) = 0.0
PSIBPE(I) = 0.0
IF(I,NE,1) GO TO 100
DO 50 JJ=1,2

C
GET INTERNAL LOAD CASE NUMBER

C
120 CALL ACCLCS(I,PHI(5),ML,1,0,IERR)
IF(IERR,NE,0) GO TO 805
CALL ACCLCS(2,IPNLCS,L(JJ),2,ILCLS,IERR)
IF(IERR,NE,0) GO TO 805
ILCN(JJ) = DCLS(21)
50 CONTINUE

C
SETUP POINTER FOR STRESS-STRAIN BUFFER

C
100 DO 165 JJ=1,2
CALL ACCFES(1,IPNFES,IDL,1,ILCN(JJ),0,0,IERR)

C
IF (IERR .NE. 0) GO TO 801
C GET ELEMENT STRESSES AND STRAINS
C CALL ACCFES(2,IFPNS,4,KINT,1,REF,ILCN(JJ),1,SBUF,LEN,1,IERR)
IF (IERR .NE. 0) GO TO 801
M = 1
IF (JJ .EQ. 2) GO TO 140
DO 130 K=1,NSVAL
   J = M+1
   LL = M+3
   SIGMA(1,K) = SBUF(M)
   SIGMA(2,K) = SBUF(J)
   SIGMA(3,K) = SBUF(LL)
   M = M+4
130   CONTINUE
GO TO 160
140   M = 17
DO 150 K=1,NSVAL
   J = M+1
   LL = M+3
   EPSLN(1,K) = SBUF(M)
   EPSLN(2,K) = SBUF(J)
   EPSLN(3,K) = SBUF(LL)
   M = M+4
150   CONTINUE
IF (JJ .EQ. 2) GO TO 170
IF (NT.GT.1) GO TO 170
165   CONTINUE
C GET X AND Y FOR JACOBIAN EVALUATION
C CALL ACCCEL(2,IPNEWC,KINT,1,REF,LENH,JERR)
IF (IERR .NE. 0) GO TO 807
M = 1
DO 200 J=1,LENH,3
   K = J+1
   LL = J+2
   X(M) = BUF(J)
   Y(M) = BUF(K)
   Z(M) = BUF(LL)
   M = M+1
200   CONTINUE
AREA = AREAO(X,Y)
XMP = 1.0D0/AREA
C CALCULATE FORCES AT THE GAUSS POINTS
C DO 250 J=1,NWOF
   DO 250 K=1,NSVAL
      BF(K,J) = 0.0
250   CONTINUE
C LOOP OVER THE GAUSS POINTS
C DO 300 K=1,NSVAL
   FS1 = CFL(1,K)
   ETn = CFL(2,K)
300   CONTINUE
C EVALUATE SHAPE FUNCTIONS AT THE GAUSS POINTS
C
IF (ISTYP, EQ, 2) CALL EU2DLO (PSI, ETA, K, SHF, PSHPL, DSHPGX, DSHPGY, DETJ, X, Y, XENK)  
* IF (ISTYP, EQ, 4) CALL EU2DPO (PSI, ETA, K, SHF, PSHPL, DSHPGX, DSHPGY, DETJ, X, Y, IFRR) 
* IF (IERR, NE, 0) CUTO 809  
300 CONTINUE  
WRITE (10, 855)  
DO 320 K = 1, NSVAL  
320 WRITE (10, 854) K, (SIGMA (J, K), J = 1, NSIG)  
IF (NT.GT.1) GO TO 345  
WRITE (10, 860)  
MU 330 K = 1, NSVAL  
330 WRITE (10, 854) K, (EPSLN (J, K), J = 1, NSIG)  
DO 340 J = 1, NSIG  
DO 340 K = 1, NSVAL  
SE (I) = SF (I) + SIGMA (J, K) * EPSLN (J, K) * DETJ  
340 CONTINUE  
C  
C CALCULATE SENSITIVITY VECTOR  
C  
DPSIT (I) = - SE (I)  
345 IF (I, NE, ICE (NC)) GO TO 820  
C  
C* CALCULATE PSI (B) - INTEGRAL OF STRESS FUNCTION FOR ELEMENT  
C  
IF (IST, EQ, 1) GO TO 360  
DO 350 K = 1, NSVAL  
VMS = (SIGMA (1, K) ** 2 + SIGMA (2, K) ** 2 - SIGMA (1, K) * SIGMA (2, K) + 3 * SIGMA (3, K) ** 2) ** .5  
* 350 PSIIBPE (I) = PSIIBPE (I) + VMS * DETJ  
GO TO 380  
360 DO 370 K = 1, NSVAL  
TMAX = (((.5 * (SIGMA (1, K) - SIGMA (2, K))) ** 2 + SIGMA (3, K) ** 2) ** .5  
* 370 PSIIBPE (I) = PSIIBPE (I) + (.5 * (SIGMA (1, K) + SIGMA (2, K))) * TMAX) * DETJ  
380 PSIIB (I) = PSIIBPE (I) * MP  
C  
GO TO 820  
C  
WRITE ERROR MESSAGES TO THE SCREEN  
C  
900 PRINT 870, IERR  
GO TO 820  
901 PRINT 871, IERR  
GO TO 820  
905 PRINT 875, IERR  
GO TO 820  
907 PRINT 878, IERR  
GO TO 820  
909 PRINT 876, IERR  
GO TO 820  
C  
C  
820 CONTINUE  
C  
850 FORMAT (1X, 'AJOIN: LOAD IS APPLIED AT ELEMENT', 14)  
851 FORMAT (1X, 'TYPE OF STRESS IS', 14)  
854 FORMAT (1X, 12, 2X, 3(E16.8, 2X))
855  FORMAT(1X,'GP',5X,'SIGMAX(GP)',8X,'SIGMAY(GP)',8X,*'SIGMAXY(GP)')
860  FORMAT(1X,'GP',5X,'EPSLNX(GP)',8X,'EPSLNY(GP)',8X,*'EPSLNXY(GP)')
870  FORMAT(1X,'ACCEL RETURNED WITH ERROR ',I4)
871  FORMAT(1X,'ACCFES RETURNED WITH ERROR ',I4)
875  FORMAT(1X,'ACCLCS RETURNED WITH ERROR ',I4)
878  FORMAT(1X,'ACCELIC RETURNED WITH ERROR ',I4)
1004 FORMAT(I4)
2001 FORMAT(AA)
C    RETURN
    END
SUBROUTINE STI601(N1,1,L,PSIBTB)

CP****************************************************************************************************************************
CP* STI601: DESIGN SENSITIVITY VECTOR FOR A BENDING PLATE
CP****************************************************************************************************************************
CP* DESCRIPTION:
CP* . 'STI601' COMPUTER THE DESIGN SENSITIVITY VECTOR FOR
CP* A TRIANGULAR BENDING PLATE ELEMENT.
CP****************************************************************************************************************************
CP* NT COUNTER FOR FINITE DIFFERENCE
CP* I EXTERNAL ELEMENT NO. NOT YET PROCESSED
CP* L EXTERNAL LOAD CASE NOS.
CP* FSIBTB STRESS CONSTRAINT FOR BENDING PLATE
CP* CP****************************************************************************************************************************
CP* INCLUDE 'CAEGSDR.INCJ IMPLIG.SPC'
CP* INCLUDE 'CAEGSDR.INCJ ACCPN.MON'
CP* INCLUDE 'CAEGSDR.INCJ CNTL.MON'
CP* INCLUDE 'CAEGSDR.INCJ ELFDES.MON'
CP* INCLUDE 'CAEGSDR.INCJ VECTR.MON'
CP* COMMON/LCSDES/LCS(90)
CP

C EQUIVALENCE (NMAT(98),IDTL)

C DIMENSION X(3),Y(3),Z(3),SIG(3,3),EPN(J,3),L(2),ILCH(2),
C * SE(500),BUF(100),PSIBTB(500),SBUF(50),FSIBTFE(500)

C DATA K=3/1,REF/1/MP/1/I,ITYFE/1/

C SE(I) = 0.000
C PSIBTFE(I) = 0.000
C FSIBTFE(I) = 0.000

C 20 DO 50 JJ=1,2

C GET INTERNAL LOAD CASE NUMBER
C
C CALL ACCLCS(1,JPNLCS,IDBL,1,0,IERR)
C IF(IERR.NE.0) GO TO 805
C CALL ACCLCS(2,JPNLCS,L(JJ),2,DLC,IERR)
C IF(IERR.NE.0) GO TO 805
C ILCh(JJ) = DLC(21)

C GET PROPERTIES
C
C CALL ACCEPR(2,IPHEPR, IPTAB,0,BUF,LFN,IFRR)
C IF (IERR.NE.0) GO TO 809
C FB(NT)=BUF(25)
C IF(NT,GT,1) GO TO 100
C 50 CONTINUE

C GET X AND Y FOR JACOBIAN EVALUATION

C 100 CALL ACCELC(1,IPNL,C,KINF,IREF,BUF,LFN,IFRR)
C IF(IERR.NE.0) GO TO 107
M = 1
DO 110 J=1,LENB,3
  K = J+1
  LL = J+2
  X(M) = BUF(J)
  Y(M) = BUF(K)
  Z(M) = BUF(LL)
  M = M+1
110 CONTINUE
C GET STRESSES FROM ORIGINAL LOAD CASE
C CALL SSMD16(X,Y,ILCN(1),SIG)
IF(NT,GT,1) GO TO 120
C GET STRAINS FROM ADJOINT LOAD
C CALL SSMD16(X,Y,ILCN(2),EPN)
C
120 AREA = EUTRIA(X,Y)
  XMP = 1.00/AREA
C START INTEGRATION LOOP
C DO 400 IT=1,3
C
  VMS = DSORT(SIG(IT,1)**2+SIG(IT,2)**2-SIG(IT,1)*SIG(IT,2)  *
        +3*SIG(IT,1)**2)  *
  TMAX = DSORT((.5*(SIG(IT,1)-SIG(IT,2))**2+SIG(IT,3)**2)
IF(NT,GT,1) GO TO 345
IF (I.NE.11:EC(NC)) GO TO 320
IF (IST,EQ,1) GO TO 300
IF (IST,GT,2) GO TO 320
C CALCULATE VON MISES STRESS SENSITIVITY TERM
C
  SE(I) = SE(I)+(AREA/3.00)*VMS*XMP/PB(N1)
GO TO 320
C CALCULATE PRINCIPAL STRESS SENSITIVITY TERM
C 300 SE(I) = SE(I)+(AREA/3.00)*(.500*(SIG(I),1)+SIG(IT,2)+2
  *TMAX))*XMP/PB(N1)
C CALCULATE THE SENSITIVITY VECTOR
C 320 DO 340 J=1,3
  SE(I) = SE(I) - (AREA/3.00)*SIG(IT,J)*EPN(J,J)
340 CONTINUE
PSIBT(I) = SE(I)
C 345 IF(I,NE,ICE(NC)) GO TO 400
C* CALCULATE PSI(B) - INTEGRAL OF STRESS FUNCTION G FOR ELEMENT
C IF(IST.EQ,1) GO TO 360
  PSIBT(I) = PSIBT(I) +(AREA/3.00)*VMS*XMP
GO TO 400
360 PSIBT(I) = PSIBT(I)+(AREA/3.00)*(.5*(SIG(IT,1)+SIG(IT,2))
        +TMAX))*XMP
400 CONTINUE
GO TO 820
WRITE ERROR MESSAGES TO THE SCREEN
GO TO 820
PRINT 870, IERR
GO TO 820
PRINT 871, IERR
GO TO 820
PRINT 875, IERR
GO TO 820
PRINT 876, IERR
GO TO 820
CONTINUE
GO TO 820
FORMAT(1X,'***ADJOIN LOAD IS APPLIED AT ELEMENT',/4)
FORMAT(1X,'***TYPE OF STRESS IS ',A4)
FORMAT(1X,12,3(8X,16.0,2X))
FORMAT(1X,'GF',5X,'SIGMAX(GP)',9X,'SIGMAY(GP)',9X,
'SIGMAZ(GP)')</nFORMAT(1X,'GF',5X,'EPSLNX(GP)',9X,'EPSLNY(GP)',9X,
'EPSLNXY(GP)')</nFORMAT(1X,'GF',5X,'EPSLNX(GP)',9X,'EPSLNY(GP)',9X,
'EPSLNXY(GP)')</nFORMAT(1X,'ACCEL RETURNED WITH ERROR ',/4)
FORMAT(1X,'ACCEFES RETURNED WITH ERROR ',/4)
FORMAT(1X,'ACCLCS RETURNED WITH ERROR ',/4)
FORMAT(1X,'ACCEPR RETURNED WITH ERROR ',/4)
FORMAT(1X,'ACCEL RETURNED WITH ERROR ',/4)
FORMAT(A4)
RETURN
END
C ELEMENT ATTRIBUTES COMMON

COMMON/ELEDES/IED(50), IINFN(32), IEXTN(32), IULR(4), IAF(32)
EQUIVALENCE (IED(1), IITYP), (IED(2), IITYP), (IED(3), NUMFE),
* (IED(4), NUDOF), (IED(5), MAXUDOF), (IED(6), NUMLL),
* (IED(7), ILIMP), (IED(8), IACTV),
* (IED(9), RREF), (IED(10), LXL),
* (IED(11), XINT), (IED(12), XILX), (IED(13), XINT),
* (IED(14), IESHA), (IED(15), XINT), (IED(16), IECOP1),
* (IED(17), ISOP), (IED(18), ICHOU), (IED(19), NOL),
* (IED(20), IMATP), (IED(21), IMATC), (IED(22), MXEL),
* (IED(23), NUMKEL), (IED(24), NMA1), (IED(25), IPTAR),
* (IED(26), ISPFAR), (IED(27), PEMA),
* (IED(28), NSVAL), (IED(29), NSIG), (IED(30), SIG)

COMMON/SVCTR/DPSIT(500), DPSIB(500), DPSIH(500), DPSIB(500),
* TH(2), BW(2), BH(2), PB(2), ICT, ISAC, LEC, NI, NC,
* IST, ICE(200)

C

C*** CURRENT IPNT POINTER STORAGE FOR ACCESS ROUTING CALLS

COMMON/ACCIPH/IPNHLH(2), IPNNOD(2), IPNNEC(2), IPNLHM(2), IPNLHM(2),
* IPNED(2), IPNLEC(2), IPNLH(2), IPNBER, IPNAT,
* IPNNCF(2), IPNAFL(3), IPNENE(2), IPNAND(2), IPNFR(2),
* IPNNSF(2), IPNBMR(2), IPNFS(2), IPNLCS(2), IPNSTK,
* IPNRES, IPNASE, IPNCF(2), IPNFR, IPNPK,
* IPNLHI(2), IPNULK, IPNCHU(2), IPNDF(2), IPNSGM(2),
* IPNHMF, IPNTH, IPNHSS(2), IPNSTN

C

C*** GLOBAL CONTROL PARAMETERS

COMMON/CNTL/ NETY(200), NUAT(300), NLINE, NWID4, NWID1, HAUX,
* IHEAPE(13), IAUX(73, 5)