Finite Element Formulation and Solution of Contact-Impact Problems in Continuum Mechanics

Authors

Hughes, Thomas

Taylor, Robert

Sackman, Jerome

Publication Date

1974-07-01

REPORT NO.
UCB/SESM-74/08

STRUCTURAL ENGINEERING
AND STRUCTURAL MECHANICS

FINITE ELEMENT FORMULATION AND SOLUTION OF CONTACT-IMPACT PROBLEMS IN CONTINUUM MECHANICS

by

THOMAS J. HUGHES
ROBERT L. TAYLOR
JEROME L. SACKMAN

MAY 1974

DEPARTMENT OF CIVIL ENGINEERING
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA

Structures and Materials Research
Department of Civil Engineering

Report No. 74-8

FINITE ELEMENT FORMULATION AND SOLUTION OF CONTACT AND IMPACT PROBLEMS IN CONTINUUM MECHANICS

by

Thomas J. Hughes
Robert L. Taylor
Jerome L. Sackman

This report was prepared under subcontract number
N62399-73-C-0023

sponsored by

Naval Civil Engineering Laboratory
Port Hueneme, California

as part of NHTSA Interagency Agreement Number
DOT-HS-289-3-550 IA

Funded by National Highway Traffic
Safety Administration

Department of Transportation

May 1974

TABLE OF CONTENTS

Acknowledgment

This report was prepared under Subcontract number N62399-73-C-0023 sponsored by Naval Civil Engineering Laboratory, Port Hueneme, California as part of NHTSA interagency agreement number DOT-HS-289-3-550 IA which was funded by the National Highway Traffic Safety Administration, Department of Transportation.

The opinions, findings and conclusions expressed in this publication are those of the authors and not necessarily those of the Naval Civil Engineering Laboratory or the National Highway Traffic Safety Administration.

Introduction

In this report we consider the general problem of contact and impact between two bodies. The report is divided into three basic parts. These parts describe: (I) The general theory of contact-impact problems, (II) A numerical scheme for the analysis of contact-impact problems, and (III) The description of computer program FEAP 74 for the solution of contact-impact problems. In an appendix we include the program subroutines and general input description for FEAP 74.

In Sections 1 to 6, Part I, we deal with spatial aspects of the theory and in Section 7, Part I, we deal with temporal aspects. This splitting of the theory is motivated by the way we intend to numerically solve the equations, i.e., the finite element method spatially and a finite difference method temporally.

Part II considers a numerical implementation of the theory given in Part I. Section 9 deals with spatial notions of the numerical problem and Section 10 the temporal. The solution scheme for the resulting algebraic problem is discussed in Sections 11 and 12.

The computer program FEAP 74 was modified to incorporate the numerical contact-impact model. The program modifications and capabilities together with two numerical examples are contained in Part III.

Finally, in the appendix we give listings for the contact subroutines together with the data input instructions.

PART I

VARIATIONAL FORMULATION OF CONTACT-IMPACT PROBLEMS IN CONTINUUM MECHANICS

1. Preliminaries

Our conventions on indices are as follows:

Superscripts indicate to which body an entity pertains. Summation is to take place only when explicitly indicated.

Latin subscripts range over 1,2,3, while Greek subscripts range over 1,2. The summation convention is assumed to hold for both.

A body $\mathcal{B}$ is a nice connected region of $\mathbb{R}^3$ with a piecewise smooth boundary $\partial\mathcal{B}$. A contact* problem is a boundary value problem, or an initial-boundary value problem, in which two bodies, $\mathcal{B}^1$ and $\mathcal{B}^2$, interact according to the principles of mechanics. Thus the primary kinematic axiom of a contact problem is that configurations $\mathscr{b}^1$ and $\mathscr{b}^2$, of $\mathcal{B}^1$ and $\mathcal{B}^2$, respectively, do not penetrate each other, i.e.,

$$
\begin{aligned}
(\mathscr{b}^1)^\circ \cap \mathscr{b}^2 &= \emptyset, \\
\mathscr{b}^1 \cap (\mathscr{b}^2)^\circ &= \emptyset,
\end{aligned}
\tag{1}
$$

where $(\mathscr{b}^\alpha)^\circ$ denotes the interior of $\mathscr{b}^\alpha$, $\alpha = 1,2$.

On the other hand the unique conditon which characterizes contact problems is that material points on the boundaries of $\mathcal{B}^1$ and $\mathcal{B}^2$ may coalesce during the motion of the bodies. Thus we say $\mathcal{B}^1$ and $\mathcal{B}^2$ are in contact if $\partial\mathscr{b}^1 \cap \partial\mathscr{b}^2 \neq \emptyset$, and we define the contact surface $\mathscr{c}$ by

---

* It is usual for the term contact to have a static connotation while the term impact has a dynamic connotation. We shall use contact in the general sense to include static as well as dynamic phenomena.

$$\mathfrak{c} = \partial b^1 \cap \partial b^2 \,. \tag{2}$$

If $\mathcal{B}^1$ and $\mathcal{B}^2$ are never in contact then $\mathfrak{c} = \emptyset$ for all configurations b^1 and b^2, and in this case an initial-boundary value problem for $\mathcal{B}^1$ and $\mathcal{B}^2$ reduces to one in which $\mathcal{B}^1$ and $\mathcal{B}^2$ may be treated separately. Thus a non-trivial contact problem is one in which $\mathfrak{c} \neq \emptyset$ for at least one instant during the motion of $\mathcal{B}^1$ and $\mathcal{B}^2$. The picture (Fig. 1) illustrates these notions.

Equation (1) implies that $\mathfrak{c}$ is a material surface with respect to both bodies, i.e., one which is not crossed by material particles. From this we may deduce the interface conditions on $\mathfrak{c}$.

Let $\underset{\sim}{x}$ be a <u>persistent</u> <u>point</u> of $\mathfrak{c}$ (one at which joining or releasing of the bodies is not instantaneously occurring) and $\underset{\sim}{v}$ be the velocity of $\underset{\sim}{x}$ ($\underset{\sim}{v} = \dot{\underset{\sim}{x}}$). Note that only the normal part of $\underset{\sim}{v}$ is independent of the parametrization of $\mathfrak{c}$. Let $\underset{\sim}{v}^1$ and $\underset{\sim}{v}^2$ be the velocities of the material particles located at the points $\underset{\sim}{x}^1$ and $\underset{\sim}{x}^2$, contained in ∂b^1 and ∂b^2, respectively, such that $\underset{\sim}{x} = \underset{\sim}{x}^1 = \underset{\sim}{x}^2$ at the present instant. Then since $\mathfrak{c}$ is material and $\underset{\sim}{x}$ is persistant

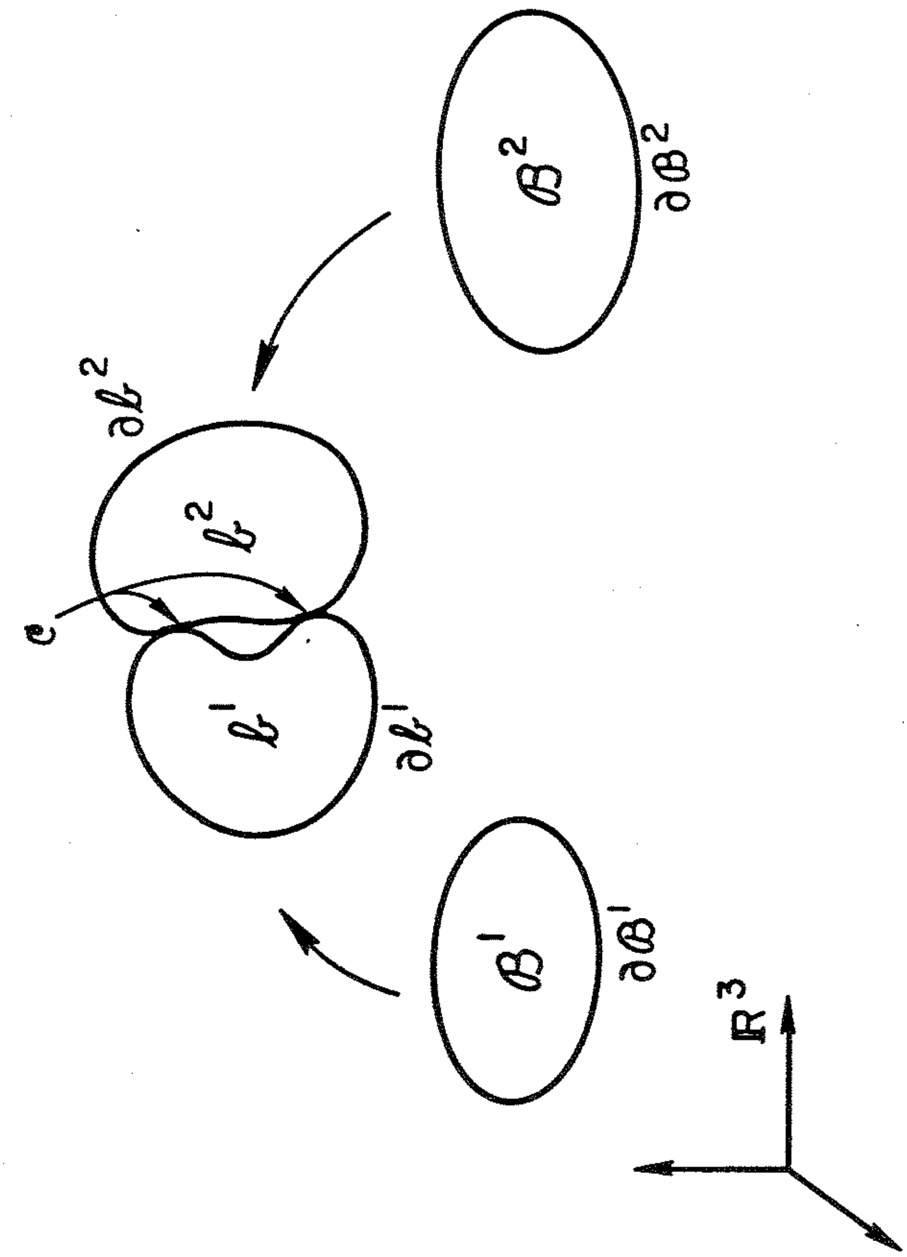
$$\underset{\sim}{v} \cdot \underset{\sim}{n} = \underset{\sim}{v}^1 \cdot \underset{\sim}{n} = \underset{\sim}{v}^2 \cdot \underset{\sim}{n} \;, \tag{3}$$

where $\underset{\sim}{n}$ is a unit normal vector to $\mathfrak{c}$ at $\underset{\sim}{x}$. From this it follows that a necessary condition for momentum to be balanced at $\underset{\sim}{x}$ is that

$$(\underset{\sim}{t}^1 + \underset{\sim}{t}^2) \cdot \underset{\sim}{n} = \underset{\sim}{0} \;, \tag{4}$$

where $\underset{\sim}{t}^\alpha$ is the Cauchy traction vector with respect to ∂b^α .

In addition we assume that no tensile tractions can occur on $\mathfrak{c}$,



$\mathcal{B}^2$
$\partial\mathcal{B}^2$
∂b^2
b^2
c
b^1
∂b^1
$\mathcal{B}^1$
$\partial\mathcal{B}^1$
$\mathbb{R}^3$

Figure 1

$$\underset{\sim}{t}^{\alpha} \cdot \underset{\sim}{n}^{\alpha} \leq 0 \ , \tag{5}$$

where $\underset{\sim}{n}^{\alpha}$ is the outward unit normal vector to $\partial\mathcal{B}^{\alpha}$. This condition excludes the possibility of the two bodies being glued together. Conditions (1-5) characterize our notion of a contact problem.

Note that thus far we have said nothing about the tangential parts of $\underset{\sim}{v}^{\alpha}$ and $\underset{\sim}{t}^{\alpha}$. These remaining conditions are determined by the frictional nature of the contact. We shall study two simple cases.

Case I: If we assume that points, once in contact, move with $\mathfrak{c}$ until released, we have that

$$\underset{\sim}{v}^{1} = \underset{\sim}{v}^{2} \ , \tag{6}$$

and therefore

$$\underset{\sim}{t}^{1} + \underset{\sim}{t}^{2} = \underset{\sim}{0} \ . \tag{7}$$

For this model we say that a no-slip, or perfect friction, condition is achieved on $\mathfrak{c}$. Thus condition (5) and equations (6) and (7) are the interface conditions for this case.

Case II: We may create the interface conditions for a frictionless, sliding contact by asserting that the tangential part of each $\underset{\sim}{t}^{\alpha}$ is identically zero,

$$\underset{\sim}{t}^{\alpha} - (\underset{\sim}{t}^{\alpha} \cdot \underset{\sim}{n}^{\alpha})\underset{\sim}{n}^{\alpha} = \underset{\sim}{0} \ . \tag{8}$$

Eq. (8), along with (3-5), are the interface conditions for this case.

2. Variational Theorems

We will formulate a variational theorem for the contact problem of finite elastodynamics. We point out, however, that our treatment is entirely general and could be used in conjunction with any field theory, as the only unique feature of the formulation involves the handling of interface conditions. At the same time finite elastodynamics, though lending itself to a clean and simple variational statement, is a case of wide practical interest.

We shall first obtain a variational theorem for the usual initial-boundary value problem of finite elastodynamics by a trivial generalization of some work done by S. Nemat-Nasser [1].

For notational simplicity let $\mathcal{A}$ denote $\partial\mathcal{B}$, and let $d\mathcal{A}$ and $d\mathcal{B}$ denote area and volume forms for $\mathcal{B}$ and $\mathcal{A}$, respectively. Let $\mathcal{A}_T \subset \mathcal{A}$ be that part of $\mathcal{A}$ where surface tractions are prescribed, and denote by $\bar{\underset{\sim}{T}}$ the Piola - Kirchhoff traction vector representing these prescribed tractions. Call ρ_o the density of $\mathcal{B}$ in the initial configuration, $\underset{\sim}{F}$ the extrinsic body force vector and let $\underset{\sim}{x} = \underset{\sim}{x}_t(\underset{\sim}{X})$ represent the position at time t of the material particle located at $\underset{\sim}{X}$ in the initial configuration. For convenience we take $\mathcal{B}$ to be the initial configuration. We denote by $\partial\underset{\sim}{x}/\partial\underset{\sim}{X}$ the deformation gradients and by $\Phi(\partial\underset{\sim}{x}/\partial\underset{\sim}{X})$ the strain energy density. Then if $\underset{\sim}{x}$ satisfies the kinematic boundary conditions

$$\underset{\sim}{x} = \bar{\underset{\sim}{x}} \tag{9}$$

on $\mathcal{A}_x \subset \mathcal{A}$, where

$$\mathcal{A}_x \cup \mathcal{A}_T = \mathcal{A},$$
$$\mathcal{A}_x \cap \mathcal{A}_T = \emptyset,$$

the functional Π defined by

$$\Pi(\underset{\sim}{x}) = \int_0^t \Big\{ \int_{\mathcal{B}} \big(\Phi(\partial \underset{\sim}{x}/\partial \underset{\sim}{X}) - \rho_0 \dot{\underset{\sim}{x}} \cdot \dot{\underset{\sim}{x}}/2 - \rho_0 \underset{\sim}{F} \cdot \underset{\sim}{x} \big) d\mathcal{B} - \int_{\mathcal{A}_T} \underset{\sim}{x} \cdot \underset{\sim}{\bar{T}}\, d\mathcal{A} \Big\} dt \,, \tag{10}$$

is stationary, i.e., its first variation vanishes

$$0 = \delta\Pi(\underset{\sim}{x}, \delta\underset{\sim}{x}) = \int_0^t \Big\{ \int_{\mathcal{B}} \big(\rho_0 (\ddot{\underset{\sim}{x}} - \underset{\sim}{F}) - \mathrm{DIV}\, \underset{\sim}{P} \big) \cdot \delta\underset{\sim}{x}\, d\mathcal{B} + \int_{\mathcal{A}_T} (\underset{\sim}{T} - \underset{\sim}{\bar{T}}) \cdot \delta\underset{\sim}{x}\, d\mathcal{A} \Big\} dt \,, \tag{11}$$

subject to the constraint on variations $\quad \delta\underset{\sim}{x}_t = \delta\underset{\sim}{x}_0 = \underset{\sim}{0} \,,$ on $\mathcal{B}_x$ (12)

if and only if the equations of motion and traction boundary conditions are satisfied

$$\rho_0 (\ddot{\underset{\sim}{x}} - \underset{\sim}{F}) = \mathrm{DIV}\, \underset{\sim}{P} \,, \qquad \text{in } \mathcal{B} \,, \tag{13}$$

$$\underset{\sim}{T} = \underset{\sim}{\bar{T}} \,, \qquad \text{on } \mathcal{A}_T \,, \tag{14}$$

where $\underset{\sim}{P} = \partial\Phi/\partial(\partial\underset{\sim}{x}/\partial\underset{\sim}{X})$ is the first Piola - Kirchhoff stress tensor, $\underset{\sim}{T} = \underset{\sim}{N} \cdot \underset{\sim}{P}$ is the Piola - Kirchhoff traction vector, and $\underset{\sim}{N}$ is the outward unit normal vector to $\mathcal{A}$. The solution to the initial-boundary value problem must also satisfy the given initial conditions

$$\left.\begin{aligned} \underset{\sim}{x} &= \underset{\sim}{x}_0 \\ \underset{\sim}{v} &= \underset{\sim}{v}_0 \end{aligned}\right\} \quad \text{in } \mathcal{B} \,. \tag{15}$$

To interpret this variational theorem for two (non-interacting) bodies set

$$\mathcal{B} = \mathcal{B}^1 \cup \mathcal{B}^2 \,,$$

$$\mathcal{A} = \mathcal{A}^1 \cup \mathcal{A}^2 \,, \quad \text{etc.} \,,$$

and write

$$\Pi(\underset{\sim}{x}) = \Pi^{1}(\underset{\sim}{x}^{1}) + \Pi^{2}(\underset{\sim}{x}^{2}) .$$

The next step is to add to Π terms manifesting the interface conditions on $\mathcal{C}$ and to stipulate the constraints under which the vanishing of the first variation of the appended functional corresponds to a solution of the contact problem. To do this we must consider further the kinematics and geometry of $\mathcal{C}$.

Define two piecewise smooth, invertible maps $\underset{\sim}{\chi}^{1}$, $\underset{\sim}{\chi}^{2}$ by the condition

$$(\underset{\sim}{\chi}^{\alpha})^{-1} : \mathcal{C} \longrightarrow C^{\alpha} \subset \mathcal{A}^{\alpha} , \tag{16}$$

where each $\underset{\sim}{\chi}'^{\alpha}$ identifies points on the boundary of the initial configuration $\mathcal{B}^{\alpha}$ which map into the contact surface $\mathcal{C}$ at each instant of time. If $\underset{\sim}{x} \in \mathcal{C}$, then $\underset{\sim}{X}^{1} = (\underset{\sim}{\chi}'^{1})^{-1}(\underset{\sim}{x})$ and $\underset{\sim}{X}^{2} = (\underset{\sim}{\chi}'^{2})^{-1}(\underset{\sim}{x})$ are the positions of particles in $\mathcal{A}^{1}$ and $\mathcal{A}^{2}$, respectively, which have coalesced at $\underset{\sim}{x} \in \mathcal{C}$. It is clear what the $\underset{\sim}{\chi}'^{\alpha}$'s really are, viz., if $\underset{\sim}{x}^{\alpha} = \underset{\sim}{x}^{\alpha}_{t}(\underset{\sim}{X}^{\alpha})$, for all $\underset{\sim}{X}^{\alpha} \in \mathcal{B}^{\alpha}$, represents the motion of body $\mathcal{B}^{\alpha}$ from the original configuration $\mathcal{B}^{\alpha}$ to the present one $\mathcal{b}^{\alpha}$, then $\underset{\sim}{\chi}'^{\alpha}$ is the restriction of $\underset{\sim}{x}^{\alpha}$ to C^{α},

$$\underset{\sim}{\chi}'^{\alpha}(\underset{\sim}{X}^{\alpha}) = \underset{\sim}{x}^{\alpha}(\underset{\sim}{X}^{\alpha}) \quad , \tag{17}$$

for each $\underset{\sim}{X}^{\alpha} \in C^{\alpha}$, $\alpha = 1,2$. For the time being we consider the $\underset{\sim}{\chi}'^{\alpha}$'s as maps defined independently of the $\underset{\sim}{x}^{\alpha}$'s and consider (17) a constraint on possible motions.

We are interested in to what extent the relation

$$\underset{\sim}{x} = \underset{\sim}{\chi}^1(\underset{\sim}{X}^1) = \underset{\sim}{\chi}^2(\underset{\sim}{X}^2) \quad , \tag{18}$$

is smooth in time and analogously under what circumstances the variations of the $\underset{\sim}{\chi}^\alpha$'s are equal. In general the $\underset{\sim}{\chi}^\alpha$'s will not even be continuous in time since contact surfaces can be instantaneously created or destroyed. If we eliminate such exceptional instants and consider only persistent points, the bodies still may slide with respect to each other, as depicted in Fig. 2. Thus tangential velocities are seen to be unequal in general. However, when $\underset{\sim}{x}$ is persistent, the impenetrability condition (1) forces the normal velocity components to be equal, and concomitantly the normal components of variations of the $\underset{\sim}{\chi}^\alpha$'s are also equal

$$\delta\underset{\sim}{\chi}^1 \cdot \underset{\sim}{n} = \delta\underset{\sim}{\chi}^2 \cdot \underset{\sim}{n} \tag{19}$$

For sliding contact (Case II), Eq. (19) characterizes the constraint on variations of the $\underset{\sim}{\chi}^\alpha$'s eqivalent to the velocity constraint (3).

For no-slip contact (Case I),

$$\delta\underset{\sim}{\chi}^1 = \delta\underset{\sim}{\chi}^2 \quad , \tag{20}$$

is easily seen to be the condition on variations equivalent to Eq. (6). We shall see that Eqs. (19) and (20) lead to the proper interface conditions in the variational theorems.

Introduce vector valued Lagrange multipliers $\underset{\sim}{\tau}^\alpha$, and add

$$\mathcal{K} = -\sum_{\alpha=1}^{2} \int_0^t \int_{\mathcal{C}^\alpha} \underset{\sim}{\tau}^\alpha \cdot (\underset{\sim}{x}^\alpha - \underset{\sim}{\chi}^\alpha)\, d\mathcal{C}^\alpha dt \quad , \tag{21}$$

to the functional Π (Eq. 10). Note that when $\mathcal{C} \neq \emptyset$,

$$\mathcal{A}^\alpha = \mathcal{A}^\alpha_x \cup \mathcal{A}^\alpha_\tau \cup \mathcal{C}^\alpha \quad ,$$

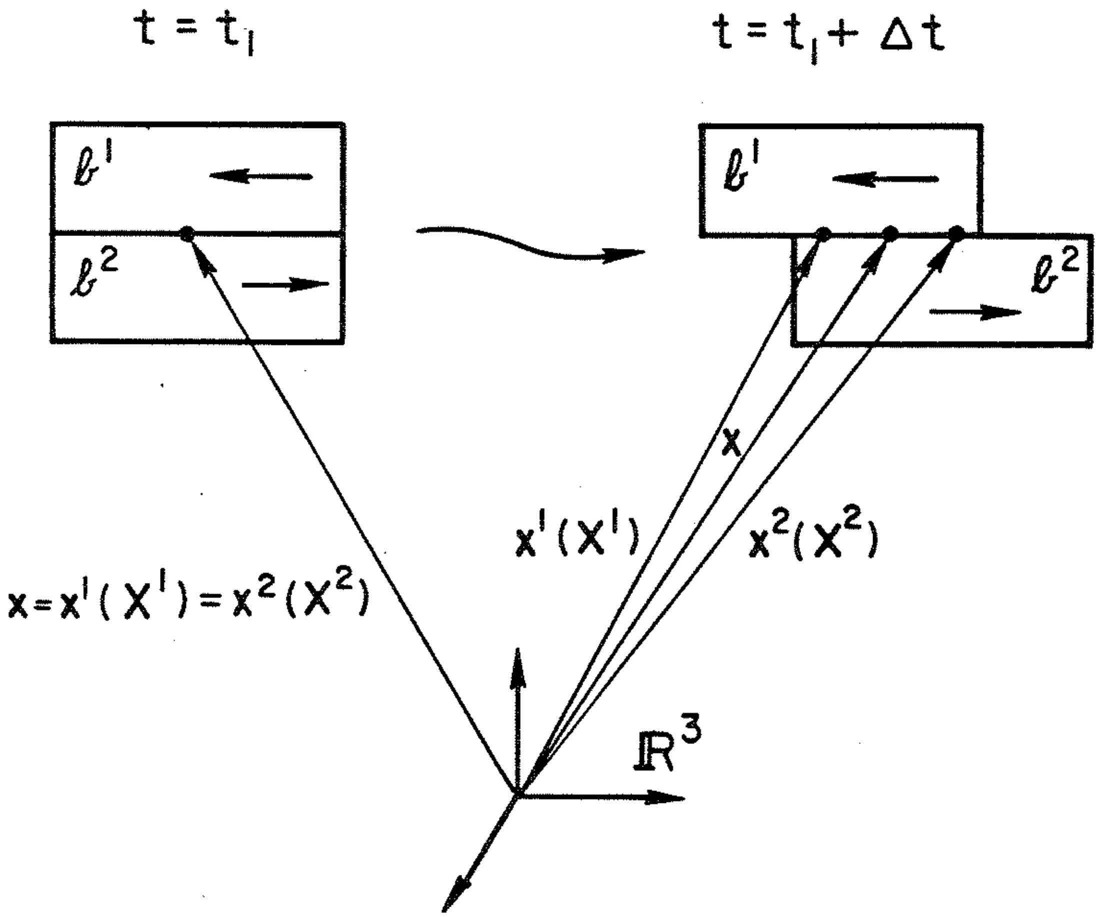


Figure 2

and assume for consistency's sake that

$$\mathcal{C}^{\alpha} \subset \mathcal{Q}^{\alpha}_{T} ,$$

$$\underset{\sim}{T} = \underset{\sim}{0} \qquad \text{on} \quad \mathcal{C}^{\alpha}. \tag{22}$$

This condition will preclude the ambiguous circumstance of non-zero tractions being specified on the contact area. Upon taking variations of $\mathcal{J} = \Pi + \mathcal{K}$ we get Eqs. (13), (14) and,

$$0 = -\sum_{\alpha=1}^{2} \int_{0}^{t} \int_{\mathcal{C}^{\alpha}} \{ \delta\underset{\sim}{\tau}^{\alpha} \cdot (\underset{\sim}{x}^{\alpha} - \underset{\sim}{\chi}^{\alpha}) + \tag{23}$$

$$+ \delta\underset{\sim}{x}^{\alpha} \cdot (\underset{\sim}{\tau}^{\alpha} - \underset{\sim}{T}^{\alpha}) - \delta\underset{\sim}{\chi}^{\alpha} \cdot \underset{\sim}{\tau}^{\alpha} \} d\mathcal{C}^{\alpha} dt \quad +$$

+ transversality condition.

The transversality condition is the classical terminology for variations associated with the domain $\mathcal{C}^{\alpha}$.

The first summand of (23) gives us (17) which insures that the $\underset{\sim}{x}^{\alpha}$'s map into $\mathcal{C}$ properly. The second summand identifies $\underset{\sim}{\tau}^{\alpha}$ as the Piola - Kirchhoff traction vector $\underset{\sim}{T}^{\alpha}$ on $\mathcal{C}^{\alpha}$. Let us investigate the third summand.

Consider first Case I and define

$$\delta\underset{\sim}{\chi} = \delta\underset{\sim}{\chi}^{\alpha} \quad , \ \alpha = 1,2 , \tag{24}$$

which makes sense because of Eq. (20). This condition is equivalent to insisting

$$\dot{\underset{\sim}{\chi}} = \dot{\underset{\sim}{\chi}}^{\alpha} \quad , \ \alpha = 1,2 ,$$

thus the first summand of (23) also implies (6) holds whenever we have a persistent point. Let j^α denote the Jacobian determinant associated with χ^α,

$$da = j^\alpha dA^\alpha. \tag{25}$$

Notice then that since $\underset{\sim}{T}^\alpha$ is the Piola - Kirchhoff traction vector, $(1/j^\alpha)\underset{\sim}{T}^\alpha$ is the corresponding Cauchy traction vector. With these we have for the third summand,

$$0 = \sum_{\alpha=1}^{2} \int_{A^\alpha} \delta\chi^\alpha \cdot \underset{\sim}{T}^\alpha dA^\alpha = \int_{a} \delta\chi \cdot ((1/j^1)\underset{\sim}{T}^1 + (1/j^2)\underset{\sim}{T}^2)da \quad , \tag{26}$$

which in words means the Cauchy traction vectors are in equilibrium. Thus the momentum balance, Eq. (7), is satisfied on a.

In Case II we only have that (19) holds, so define

$$\delta\chi(n) = \delta\chi^\alpha \cdot \underset{\sim}{n} \quad , \quad \alpha = 1,2 . \tag{27}$$

This requirement also insures that,

$$\dot{\chi}^1 \cdot \underset{\sim}{n} = \dot{\chi}^2 \cdot \underset{\sim}{n} \quad ,$$

thus the first summand of (23) implies (3). For this case the third summand takes the form,

$$\begin{aligned} 0 = \sum_{\alpha=1}^{2} \int_{A^\alpha} \delta\chi^\alpha \cdot \underset{\sim}{T}^\alpha dA^\alpha &= \int_{a} \delta\chi(n)((1/j^1)\underset{\sim}{T}^1 \cdot \underset{\sim}{n} + (1/j^2)\underset{\sim}{T}^2 \cdot \underset{\sim}{n})da \\ &+ \sum_{\alpha=1}^{2} \int_{A^\alpha} (\delta\chi^\alpha - \delta\chi(n)\underset{\sim}{n}) \cdot \underset{\sim}{T}^\alpha dA^\alpha . \end{aligned} \tag{28}$$

The integral over a gives us Eq. (4). The significance of the second integral hinges on the observation that $(\delta\chi^\alpha - \delta\chi(n)\underset{\sim}{n})$ is a tangent vector to a for each α. Thus the tangential part of each $\underset{\sim}{T}^\alpha$ is identically zero, which is equivalent to the shear free condition, Eq. (8), which we require for Case II.

A standard calculation enables us to write the transversality condition as,

$$0 = \sum_{\alpha=1}^{2} \int_{\partial\mathcal{C}^\alpha} (\delta\underset{\sim}{\chi}^\alpha \cdot \underset{\sim}{T}^\alpha)(\underset{\sim}{\tau}^\alpha \cdot (\underset{\sim}{x}^\alpha - \underset{\sim}{x}'^\alpha))\, d(\partial\mathcal{C}^\alpha) \quad , \tag{29}$$

where the transversal $\underset{\sim}{T}^\alpha$ is a unit vector field tangent to $\mathcal{C}^\alpha$, and perpendicular and pointing outward with respect to $\partial\mathcal{C}^\alpha$, Fig. 3. Thus (29) implies that

$$\underset{\sim}{\tau}^\alpha \cdot (\underset{\sim}{x}^\alpha - \underset{\sim}{x}'^\alpha) = 0 \quad \text{on} \quad \partial\mathcal{C}^\alpha, \ \alpha = 1,2 \tag{30}$$

Assuming continuity of the integrands of (21) on the closure of $\mathcal{C}^\alpha$, condition (30) is already implied by the first summand of (23). This assumption precludes $\underset{\sim}{\tau}^\alpha$ taking the form of a δ-distribution on $\partial\mathcal{C}^\alpha$.

Although this assumption is warranted here it may not be true when one employs certain approximate theories in mechanics. For instance consider the case where a Bernoulli-Euler beam is uniformly loaded and sits on a rigid parabolic surface (Fig. 4). At the contact points a, a', concentrated reactions must exist to balance shear forces. This example is actually from a completely different class of contact problems in that contact is made along a part of the interior rather than the boundary. Such problems as the contact of plates and shells also fall into this class. We could summarize such situations by the description -- m-dimensional contact of m-dimensional bodies, e.g., for the beam m=1, and for plates and shells m=2. The case under investigation in this paper (m=3) is an example of the (m-1)-dimensional contact of m-dimensional bodies.

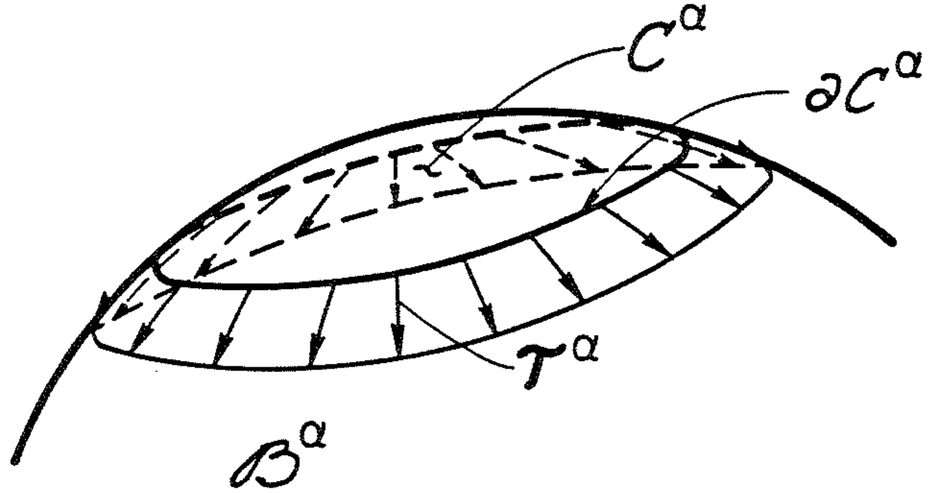


Figure 3

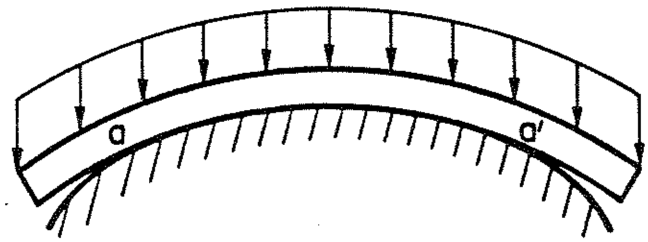


Figure 4

It is good to keep in mind cases such as that illustrated in Fig. 4 when considering specific boundary value problems.

A further point worth mentioning here is that the transversality condition will in general be an independent one in a numerical algorithm. For example, if the fields in the integrand of (21) are approximated by a family of trial functions, Eq. (23) only implies that some weighted integrals over the $\mathcal{C}^{\alpha}$'s vanish. The condition (29) requires that weighted integrals over the $\partial\mathcal{C}^{\alpha}$'s also vanish.

We now summarize our results in the following theorems:

<u>Theorem I</u>: Let (1), (2), (5), (9), (12), (15), and (20) hold. Then $\underset{\sim}{x}$ is a solution to the no-slip contact problem (Case I), that is, (6), (7), (13), (14) and (17) also hold if, and only if, $\delta\mathcal{G} = 0$ for arbitrary variations of $\underset{\sim}{x}^{\alpha}, \underset{\sim}{\chi}^{\alpha}$ and $\underset{\sim}{\tau}^{\alpha}, \alpha = 1,2$.

<u>Theorem II</u>: Let (1), (2), (5), (9), (12), (15), and (19) hold. Then $\underset{\sim}{x}$ is a solution to the sliding contact problem (Case II), that is, (3), (4), (8), (13), (14) and (17) also hold if, and only if, $\delta\mathcal{G} = 0$ for arbitrary variations of $\underset{\sim}{x}^{\alpha}, \underset{\sim}{\chi}^{\alpha}$ and $\underset{\sim}{\tau}^{\alpha}, \alpha = 1,2$.

3. Consideration of Theorems I and II as Computational Tools

Theorems I and II may be employed to generate numerical algorithms for the solution of contact problems. The basic idea is to represent $\underset{\sim}{x}^{\alpha}$, $\underset{\sim}{\dot{x}}^{\alpha}$ and $\underset{\sim}{\tau}^{\alpha}$ as the product of known functions on $\mathbb{R}^3$ with unknown parameters depending on time. Then Theorems I and II provide us with a method for generating an approximate system of equations (e.g., by the classical Ritz-Galerkin technique) in terms of these unknown parameters, which then can be solved incrementally and/or iteratively, subject to the side conditions of the theorems. The constraints (1) and (5) will both take the form of inequalities in actual computations, thus the ideas of optimization theory will probably be useful in the actual construction of a numerical algorithm.

The finite element method is a powerful technique for obtaining a system of approximate equations, and it is of interest to find out how amenable are Theorems I and II to a finite element formulation. Unfortunately the term $\mathcal{K}$ would result in a terrible mess if the integrand was represented by typical finite element functions. This is because the boundaries of the $\mathcal{C}^{\alpha}$'s are unknown and thus a parametric integration would bury the defining parameters of the $\mathcal{C}^{\alpha}$'s in the arguments of Heaviside functions representing the supports of the elements. Note that a classical Ritz-Galerkin approximation would not be subject to this pitfall, since the associated trial functions could be chosen to be real analytic and thus easily integrated parametrically to a relatively simple form. However, such a formulation is restricted to a geometrically simpler class of problems. Thus it is desirable to seek a generalization that will lend itself cleanly to a finite element formulation.

4. Variational Theorems Without Transversality Conditions

Let $\tilde{\mathcal{E}}^\alpha$ be a fixed part of $\mathcal{A}^\alpha_T$ such that

$$\tilde{\mathcal{E}}^\alpha \supset \mathcal{E}^\alpha \quad , \tag{31}$$

and

$$\mathbf{T} = \mathbf{0} \quad \text{on} \quad \tilde{\mathcal{E}}^\alpha \sim \mathcal{E}^\alpha . \tag{32}$$

Define a scalar valued function η^α on $\tilde{\mathcal{E}}^\alpha$ such that

$$\eta^\alpha(\mathbf{X}^\alpha) = 0 \qquad \text{if} \quad \mathbf{X}^\alpha \in \tilde{\mathcal{E}}^\alpha \sim \mathcal{E}^\alpha . \tag{33}$$

Let $\tilde{\varepsilon} \supset \varepsilon$, and define the maps $\tilde{\chi}^\alpha$ by the condition

$$(\tilde{\chi}^\alpha)^{-1} : \tilde{\varepsilon} \longrightarrow \tilde{\mathcal{E}}^\alpha \quad ,$$

where, as before, $\tilde{\chi}^\alpha$ represents $\mathbf{x}^\alpha$ on $\mathcal{E}^\alpha$; but on $\tilde{\mathcal{E}}^\alpha \sim \mathcal{E}^\alpha$ we place no physical interpretation on $\tilde{\chi}^\alpha$. Thus on $\tilde{\mathcal{E}}^\alpha$ we will always have that,

$$\eta^\alpha(\mathbf{x}^\alpha - \tilde{\chi}^\alpha) = \mathbf{0} \quad , \tag{34}$$

since $\mathbf{x}^\alpha = \tilde{\chi}^\alpha$ on $\mathcal{E}^\alpha$ and $\eta^\alpha = 0$ on the relative complement $\tilde{\mathcal{E}}^\alpha \sim \mathcal{E}^\alpha$.

Introduce vector valued Lagrange multipliers $\boldsymbol{\sigma}^\alpha$ and let $\mathcal{L} = \mathrm{II} + \mathcal{M}$ where

$$\mathcal{M} = - \sum_{\alpha=1}^{2} \int_0^t \int_{\tilde{\mathcal{E}}^\alpha} \boldsymbol{\sigma}^\alpha \cdot \eta^\alpha (\mathbf{x}^\alpha - \tilde{\chi}^\alpha) \, d\tilde{\mathcal{E}}^\alpha dt . \tag{35}$$

We require that the variations of $\tilde{\chi}^\alpha$ satisfy the same conditions as before, but now for all $\tilde{\mathcal{E}}^\alpha$:

$$\left.\begin{aligned} &\text{Case I:} \quad \delta\tilde{\chi} \overset{\text{def.}}{=} \delta\tilde{\chi}^\alpha \\ &\text{Case II:} \quad \delta\tilde{\chi}(n) \overset{\text{def.}}{=} \delta\tilde{\chi}^\alpha \cdot \mathbf{n} \end{aligned}\right\} \quad \text{on} \quad \tilde{\mathcal{E}}^\alpha \tag{36}$$

where $\underset{\sim}{n}$ is a unit normal vector to $\mathcal{E}$. Computing the first variation of $\mathcal{L}$ we have the usual conditions emanating from II and

$$\begin{aligned} 0 = -\sum_{\alpha=1}^{2} \int_0^t \int_{\tilde{\mathcal{E}}^\alpha} \{ & \delta\underset{\sim}{\sigma}^\alpha \cdot (\eta^\alpha(\underset{\sim}{x}^\alpha - \underset{\sim}{\chi}^\alpha)) \\ & + \delta\eta^\alpha(\underset{\sim}{\sigma}^\alpha \cdot (\underset{\sim}{x}^\alpha - \underset{\sim}{\chi}^\alpha)) \\ & + \delta\underset{\sim}{x}^\alpha \cdot (\eta^\alpha \underset{\sim}{\sigma}^\alpha - \underset{\sim}{T}^\alpha) \\ & - \delta\underset{\sim}{\chi}^\alpha \cdot (\eta^\alpha \underset{\sim}{\sigma}^\alpha) \} d\tilde{\mathcal{E}}^\alpha dt . \end{aligned} \tag{37}$$

The first summand gives us (34) and we define

$$\mathcal{C}^\alpha = \{ \underset{\sim}{X}^\alpha \in \tilde{\mathcal{E}}^\alpha : \underset{\sim}{x}^\alpha(\underset{\sim}{X}^\alpha) = \underset{\sim}{\chi}^\alpha(\underset{\sim}{X}^\alpha) \} . \tag{38}$$

The third summand defines $\eta^\alpha \underset{\sim}{\sigma}^\alpha$ as the Piola - Kirchhoff traction vector. Note that this insures that $\underset{\sim}{T} = \underset{\sim}{0}$ on $\tilde{\mathcal{E}}^\alpha \sim \mathcal{C}^\alpha$ since $\eta^\alpha = 0$ there. The fourth summand gives us the appropriate Cauchy traction condition across $\mathcal{C}$ for each case of (36). The second summand is identically satisfied on $\mathcal{C}^\alpha$ since $\underset{\sim}{x}^\alpha = \underset{\sim}{\chi}^\alpha$. On $\tilde{\mathcal{E}}^\alpha \sim \mathcal{C}^\alpha$ it tells us that $\underset{\sim}{\sigma}^\alpha$ is orthogonal to $\underset{\sim}{x}^\alpha - \underset{\sim}{\chi}^\alpha$, but this is of no physical interest.

Thus we can state the following theorems:

<u>Theorem I'</u>: Let (1), (2), (5), (9), (12), (15), (31), (32) and $(36)_1$ hold. Then $\underset{\sim}{x}$ is a solution to the no-slip contact problem (Case I), that is, (6), (7), (13), (14) and (17) also hold where $\mathcal{C}^\alpha$ is defined by (38), if $\delta\mathcal{L} = 0$ for arbitrary variations of $\underset{\sim}{x}^\alpha, \underset{\sim}{\chi}^\alpha, \eta^\alpha$ and $\underset{\sim}{\sigma}^\alpha$, $\alpha = 1,2$.

<u>Theorem II'</u>: Let (1), (2), (5), (9), (12), (15), (31), (32) and $(36)_2$ hold. Then $\underset{\sim}{x}$ is a solution to the sliding contact problem (Case II), that is, (3), (4), (8), (13), (14) and (17) hold where $\mathcal{C}^\alpha$ is defined by (38), if $\delta\mathcal{L} = 0$ for arbitrary variations of $\underset{\sim}{x}^\alpha, \underset{\sim}{\chi}^\alpha, \eta^\alpha$ and $\underset{\sim}{\sigma}^\alpha$, $\alpha = 1,2$.

The important feature of these theorems is that the regions $\tilde{\mathcal{E}}^{\alpha}$ are fixed. Thus transversality conditions are absent, and the theorems may be applied to finite element formulations. In fact one would naturally take $\tilde{\mathcal{E}}^{\alpha}$ to be a union of elements in $\mathcal{a}^{\alpha}$, large enough to contain $\mathcal{E}^{\alpha}$ throughout the motion.

Thus far our considerations have been quite general and, in fact, more general than would be required for the solution of particular classes of contact problems. In the next section we illustrate the many simplifications which can be made in the application of the preceeding theorems to a class of problems of wide practical interest.

5. Hertzian Contact Problems

We wish to characterize contact problems in which the contact surface is approximately planar and the bodies have undergone small deformations in the neighborhood of the contact surface.

Assume the following:

(1) $\underset{\sim}{n} \overset{\text{def.}}{=} n_i \underset{\sim}{e}_i \approx \underset{\sim}{e}_3$ on $\mathcal{C}$, where the n_i indicate components with respect to the standard basis $\{\underset{\sim}{e}_i\}_1^3$ for $\mathbb{R}^3$, (see Fig. 5).

(2) $j^\alpha \approx 1$, $\alpha = 1,2$, thus $\underset{\sim}{t}^\alpha \approx \underset{\sim}{T}^\alpha$ on $\mathcal{C}^\alpha$.

Assumptions (1) and (2) together imply that,

$$t_3^\alpha \approx \underset{\sim}{t}^\alpha \cdot \underset{\sim}{n} \approx \underset{\sim}{T}^\alpha \cdot \underset{\sim}{n} \approx T_3^\alpha \quad ,$$

and that,

$$(t_1^\alpha, t_2^\alpha, 0) \approx \underset{\sim}{t}^\alpha - (\underset{\sim}{t}^\alpha \cdot \underset{\sim}{n})\underset{\sim}{n} \approx \underset{\sim}{T}^\alpha - (\underset{\sim}{T}^\alpha \cdot \underset{\sim}{n})\underset{\sim}{n} \approx (T_1^\alpha, T_2^\alpha, 0) .$$

(3) Material points which eventually contact have, to the first order, the same initial coordinates z_1 and z_2. Explicitly we manifest this idea by requiring that the $\underset{\sim}{\chi}^\alpha$'s satisfy

$$\underset{\sim}{\chi}^1(z_1, z_2, X_3^1(z_1, z_2)) = \underset{\sim}{\chi}^2(z_1, z_2, X_3^2(z_1, z_2)) \quad . \tag{39}$$

This is depicted in Fig. 6. Since X_3^α are given functions which define the surfaces $\mathcal{C}^\alpha$, it follows from (39) that,

$$\delta \underset{\sim}{\chi}^1 = \delta \underset{\sim}{\chi}^2 \quad .$$

We term problems for which these assumptions hold Hertzian, since these assumptions are implicit in Hertz' classical theory [2] (see

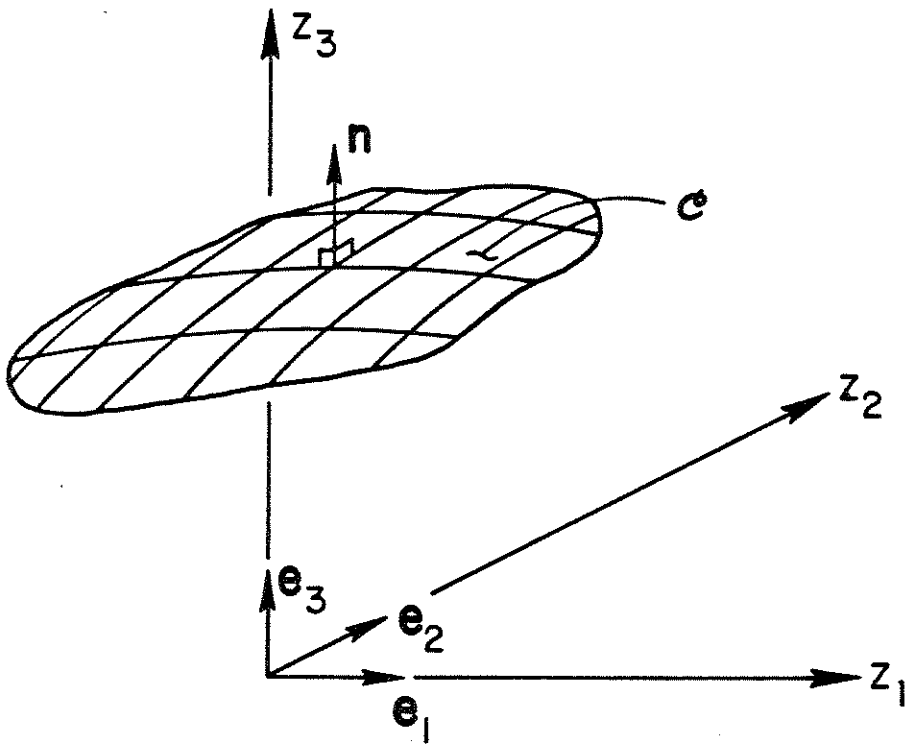


Figure 5

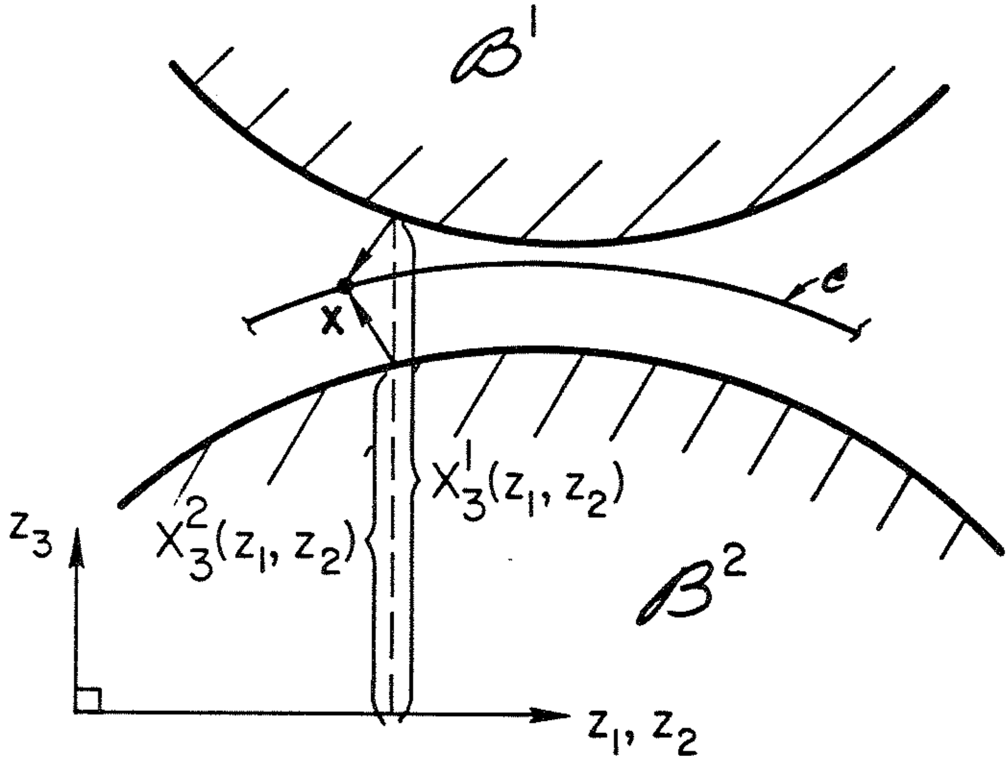


Figure 6

Goldsmith [3] for an excellent exposition of this work and also many applications of Hertz' theory to impact problems). It should be pointed out that the formulation we are about to give is still considerably more general than those to which Hertz' theory applies.

We now show how these assumptions allow us to make simplifications in the preceding theorems.

Theorems I and II:

Due to assumption (3) the term $\mathcal{K}$ can be replaced by an integral over a region in the z_1, z_2 -plane. This region, say c, is the projection of $\mathcal{C}$ onto the z_1, z_2 -plane, and due to assumption (2) it coincides, to the first order, with the projections of the $\mathcal{C}^\alpha$'s. Thus $\mathcal{K}$ can be written

$$\mathcal{K} = -\sum_{\alpha=1}^{2} \int_0^t \int_c \underset{\sim}{\tau}^\alpha \cdot (\underset{\sim}{x}^\alpha - \underset{\sim}{\chi}^\alpha)\, dc\, dt\,. \tag{40}$$

Since, for Case I, we know that the momentum balance on $\mathcal{C}$ requires that

$$\underset{\sim}{\tau}^1 + \underset{\sim}{\tau}^2 = \underset{\sim}{0} \quad ,$$

we may make use of this relation immediately. Thus define

$$\underset{\sim}{\tau} = \underset{\sim}{\tau}^1 = -\underset{\sim}{\tau}^2 \quad ,$$

and substitute into (40). Employing (39), the integrand simplifies to

$$\underset{\sim}{\tau} \cdot (\underset{\sim}{x}^2 - \underset{\sim}{x}^1)\,. \tag{41}$$

The analog of (23) becomes

$$0 = \int_0^t \int_C \Big\{ \delta\boldsymbol{\tau}\cdot(\mathbf{x}^2 - \mathbf{x}^1) + \\ + \delta\mathbf{x}^1\cdot(\mathbf{T}^1 - \boldsymbol{\tau}) + \delta\mathbf{x}^2\cdot(\mathbf{T}^2 + \boldsymbol{\tau}) \Big\}\, dc\, dt \\ + \text{transversality condition}. \tag{42}$$

Thus the same conclusions of Theorem I can be drawn. However, from a numerical standpoint things are considerably different. First of all, since the $\boldsymbol{\lambda}^\alpha$'s are absent in this formulation, we do not get a uniquely defined ℓ ; $\mathbf{x}^1$ and $\mathbf{x}^2$ will not in general be the same pointwise. If the graph of ℓ is important it could be constructed by averaging $\mathbf{x}^1$ and $\mathbf{x}^2$, which, if the solution is any good, should be reasonably close pointwise. On the other hand, the $\boldsymbol{\lambda}^\alpha$'s being absent engenders a considerable saving in the number of equations to be solved and in their complexity.

The analogous case for Theorem II is constructed simply by setting

$$\tau_1 = \tau_2 = 0 \quad , \qquad \tau \overset{\text{def.}}{=} \tau_3 \ .$$

Then the integrand of $\mathcal{K}$ becomes

$$\tau\,(x_3^2 - x_3^1) \tag{43}$$

and (42) reduces to

$$0 = \int_0^t \int_C \Big\{ \delta\tau(x_3^2 - x_3^1) + \\ + \delta x_3^1(T_3^1 - \tau) + \delta x_3^2(T_3^2 + \tau) \\ + \delta x_\alpha^1\, T_\alpha^1 \quad + \delta x_\alpha^2\, T_\alpha^2 \Big\}\, dc\, dt \\ + \text{transversality condition}. \tag{44}$$

Hence the conclusions of Theorem II hold.

Thus in the case of Hertzian contact we can add the simplifications manifested in (41) and (43) to the conditions of Theorems I and II, respectively, and still garner the same conclusions.

<u>Theorems I' and II'</u>:

For these cases $\mathcal{M}$ can be written as an integral over $\tilde{c}$, the projection of $\mathcal{E}$:

$$\mathcal{M} = -\sum_{\alpha=1}^{2}\int_{0}^{t}\int_{\tilde{c}} \underset{\sim}{\sigma}^{\alpha}\cdot n^{\alpha}(\underset{\sim}{x}^{\alpha}-\underset{\sim}{\chi}^{\alpha})\, d\tilde{c}\, dt\, .$$

Due to the present geometric situation, it is appropriate to take

$$n^{1} = n^{2} \quad ,$$

and thus define

$$n = n^{\alpha} \quad , \qquad \alpha = 1,2\, .$$

Analagous to the considerations for Theorems I and II, the momentum balance across $\mathcal{E}$ motivates the simplification

$$\underset{\sim}{\sigma} \overset{\text{def.}}{=} \underset{\sim}{\sigma}^{1} = -\underset{\sim}{\sigma}^{2}\, .$$

With these and (39), the integrand of $\mathcal{M}$ can be written

$$\underset{\sim}{\sigma}\cdot n(\underset{\sim}{x}^{2}-\underset{\sim}{x}^{1})\, .$$

A further simplification can be made by setting*

$$\sigma_{3} = -n\, .$$

This eliminates one unknown function and, as we shall see, has the effect of satisfying (5) naturally. Thus the integrand of $\mathcal{M}$ becomes

---

* This is a standard ploy of optimization theory, see p. 82, [4].

$$\sigma_\alpha \eta(x_\alpha^2 - x_\alpha^1) - (\eta)^2(x_3^2 - x_3^1)\ , \tag{45}$$

and the analog of (23) is

$$\begin{aligned} 0 = \int_0^t \int_{\mathcal{C}} \{ & \delta\eta(\sigma_\alpha(x_\alpha^2 - x_\alpha^1) - 2\eta(x_3^2 - x_3^1)) \\ & + \delta\sigma_\alpha(\eta(x_\alpha^2 - x_\alpha^1) \\ & + \delta x_\alpha^1(T_\alpha^1 - \eta\sigma_\alpha) + \delta x_\alpha^2(T_\alpha^2 + \eta\sigma_\alpha) \\ & + \delta x_3^1(T_3^1 + (\eta)^2) + \delta x_3^2(T_3^2 - (\eta)^2)\}\, d\mathcal{C}\, dt . \end{aligned} \tag{46}$$

Summand two tells us that either $\eta = 0$ or $x_\alpha^1 = x_\alpha^2$, on $\mathcal{C}$. Suppose $\eta \neq 0$, then $x_\alpha^1 = x_\alpha^2, \alpha=1,2$. Summand one then gives us that $x_3^1 = x_3^2$ on $\mathcal{C}$. Thus we have

$$\eta(\underset{\sim}{x}^2 - \underset{\sim}{x}^1) = \underset{\sim}{0}\ , \qquad \text{on} \quad \mathcal{C}\ ,$$

as required, and $\mathcal{C}$ is defined as the subset of $\mathcal{C}$ where $\underset{\sim}{x}^1 = \underset{\sim}{x}^2$.

The last four summands give the momentum balance conditions, as usual, and, in addition, the last two summands imply that the normal tractions are compressive (since $(\eta)^2 \geq 0$). Thus we have the conclusions of Theorem I' and condition (5).

The analogous set up for Theorem II' is accomplished by setting $\sigma_\alpha = 0$ in (45) yielding

$$-(\eta)^2(x_3^2 - x_3^1) \tag{47}$$

for the integrand of $\mathcal{M}$. With this Eq. (46) becomes

$$0 = \int_0^t \int_{\tilde{C}} \Big\{ -2\,\delta\eta\,(\eta\,(x_s^2 - x_s^1)) \quad + $$
$$+ \; \delta x_\alpha^1\, T_\alpha^1 \qquad + \; \delta x_\alpha^2\, T_\alpha^2$$
$$+ \; \delta x_s^1\,(T_s^1 + (\eta)^2) \quad + \; \delta x_s^2\,(T_s^2 - (\eta)^2)\Big\}\, d\tilde{\sigma}\, dt \,.$$

In this case we achieve the conclusion of Theorem II' and condition (5).

Thus to Theorems I' and II' we can delete condition (5), add the simplifications manifested in (45) and (47), and achieve the conclusions of Theorems I' and II', respectively, plus condition (5).

6. Contact Problems for One, Two and Three-dimensional Bodies

The previous work needs only trivial modification to be made applicable to contact problems involving bodies of different dimensions. There are many cases of considerable interest which fall into this category. For example, models consisting of a shell and a plate, or a solid and a plate, are useful for the study of head impact. The modifications necessary are essentially interpretative. An example illustrates this assertion.

Consider the frictionless Hertzian contact of a three-dimensional solid and a two-dimensional plate. Let $\mathcal{B}^1$ represent the solid and $\mathcal{B}^2$ the plate. In evaluating Π, the $\mathcal{B}^1$ part is as before while the $\mathcal{B}^2$ part would manifest the particular plate theory used. The contact term $\mathcal{K}$ (or $\mathcal{M}$) would be exactly as before. However note that τ (or $\bar{\tau}$) is, in this case, also identifiable with part of the two-dimensional "volume" of the plate, rather than its boundary. Taking variations, everything is as before except that the term $\tau\,\delta x_3^2$ (or $-(\eta)^2\,\delta x_3^2$) contributes to the transverse momentum equation of the plate, rather than to its boundary conditions. The interpretation of τ (or $-(\eta)^2$) is thus two-fold, i.e., it is the normal component of the traction vector with respect to $\mathcal{B}^1$, as before, and it is also the equivalent normal "body force" with respect to $\mathcal{B}^2$, manifested by the interaction with $\mathcal{B}^1$ (Fig. 7).

This interpretation is general, namely, for one and two-dimensional bodies the contact force is an equivalent "body force" which contributes to the momentum equations, rather than the boundary conditions. With this interpretation in mind, the construction of variational theorems, analogous to the ones constructed in Sections 2, 4 and 5, for the class of one, two and three-dimensional contact problems, is just a formal deductive exercise involving only appropriate definitions for Π.

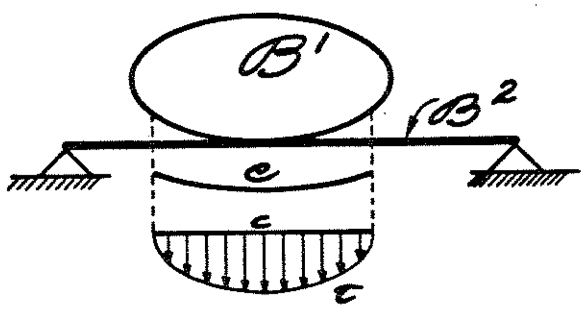


Figure 7

7. Impact

The previous sections deal with spatial aspects of contact problems. In this section we investigate temporal considerations, i.e., those phenomena which are unique to dynamic contact or impact. To manifest the problem encountered in such situations consider the following hypothetical situation. Assume that we are in the process of numerically solving some impact problem and suppose that it is discovered as we monitor the motion of the bodies that they impact somewhere in the time interval (t_1, t_2). At time t_1 we know the states of both bodies and we know that somewhere between t_1 and t_2 they have coalesced over a portion of their boundaries. Assume for the moment we know the geometry of the contact surface $\mathfrak{c}$. The question which arises then is what is the state of $\mathfrak{c}$ at time t_2, i.e., what are the velocity and traction vectors on $\mathfrak{c}$? It is necessary to know this information to carry forth the step forward time integration. The question though seems improperly posed without specifying considerable data about the nature of the impact. To get a handle on things, we will initially formulate a simple one-dimensional problem involving the impact of two elastic rods. Although this problem is trivial, it provides considerable insight into the general nature of impact of continuum bodies. Since we are interested in the state of $\mathfrak{c}$ (in this case a point) immediately after impact, whether the rods are finite or semi-infinite is immaterial.

Assume that the pre-impact states of the two bodies are given by the following data:

$$\mathcal{B}^{\alpha} : \quad v_-^{\alpha} , \quad (\partial x / \partial X)_-^{\alpha} , \quad P_-^{\alpha} \quad ; \; \alpha = 1,2 . \tag{48}$$

At impact the rods coalesce at $\mathfrak{t}$, and for some finite time interval thereafter (at least) $x \in \mathfrak{t}$ is persistent. At the moment of impact shock waves begin to propagate in each body. The space-time picture is depicted in Fig. 8. As discussed in section 1, since $\mathfrak{t}$ is material and x is persistent, we have

$$v \overset{\text{def.}}{=} v_+^1 = v_+^2 \quad , \qquad P \overset{\text{def.}}{=} T_+^1 = -T_+^2 \quad , \tag{49}^*$$

for the post-impact state (t_+). In addition to (49), the well known shock conditions must hold across the wave fronts:

$$\begin{aligned} [v^\alpha] + U^\alpha[(\partial x/\partial X)^\alpha] &= 0 \;, \\ \rho_0^\alpha U^\alpha [v^\alpha] &= [P^\alpha] \;, \end{aligned} \tag{50}$$

where U^α is the material velocity of the shock in $\mathcal{B}^\alpha$, and [ ] is the wave-front jump operator which assigns to a function the difference in its values behind and in front of the wave, i.e., $[f(X,t)] = f(X^-,t) - f(X^+,t)$ where X is a material point denoting the location of the wave-front. As can be deduced from Fig. 8, the states into which the shocks initially propagate are the pre-impact states given by (48), and the state at $\mathfrak{t}$, immediately after the shocks pass, is given by the post-impact state (49). These observations in conjunction with (50) yield,

$$\begin{aligned} v_-^\alpha - v + U^\alpha\{(\partial x/\partial X)_-^\alpha - (\partial x/\partial X)_+^\alpha\} &= 0 \;, \\ \rho_0^\alpha U^\alpha (v_-^\alpha - v) + P_-^\alpha - P &= 0. \end{aligned} \tag{51}^{**}$$

*For convenience we choose the initial state to be the pre-impact state, thus we need not distinguish between Cauchy and Piola tractions.

**A consistency condition for these equations is that $v_-^1 - v_-^2 > 0$. Otherwise the impact would not occur.

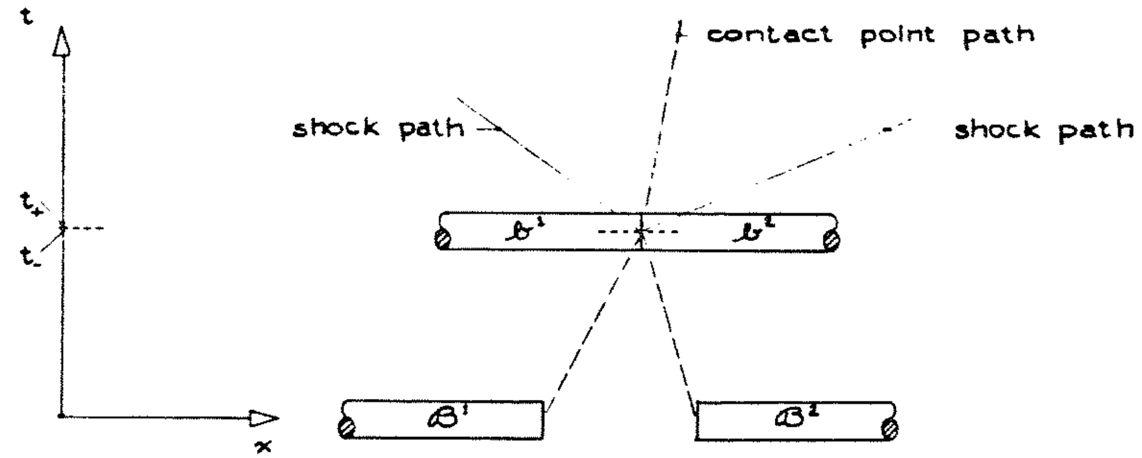

t
contact point path
shock path
shock path
t_+
t_-
b^1
b^2
$\mathcal{B}^1$
$\mathcal{B}^2$
x

Figure 8

The four Eqs. (51) and constitutive equations relating P^α to $(\partial x/\partial X)^\alpha$ yield a formally deterministic system of six equations in the six unknowns v, P, U^α, $(\partial x/\partial X)^\alpha$. Thus we see that the desired quantities v and P depend on the pre-impact states and material properties of both $\mathcal{B}^\alpha$. The precise form of this relationship depends upon the constitutive equations of the bodies. As a simple example, assume we have linear constitutive equations $P^\alpha = E^\alpha\{(\partial x/\partial X)^\alpha - 1\}$, E^α constant, and let the pre-impact state be given by

$$\begin{aligned} v_-^\alpha &= V^\alpha \,, \\ (\partial x/\partial X)_-^\alpha &= 1 \,, \\ P_-^\alpha &= 0 \,. \end{aligned} \tag{52}$$

These conditions, when inserted in Eqs. (51), lead to:

$$\begin{aligned} v &= \frac{\rho_0^2 U^2 V^2 - \rho_0^1 U^1 V^1}{\rho_0^2 U^2 - \rho_0^1 U^1} \,, \\ P &= \frac{V^2 - V^1}{\left(\frac{U^2}{E^2} - \frac{U^1}{E^1}\right)} \,, \\ (U^\alpha)^2 &= E^\alpha/\rho_0^\alpha \,. \end{aligned} \tag{53}$$

Note that the denominators in Eqs. $(53)_{1,2}$ present no problems since $\rho_0^\alpha > 0$, $E^\alpha > 0$ and $+1 = \operatorname{sgn} U^2 = -\operatorname{sgn} U^1$.
This result is also appropriate whenever the intensity of the impact is small enough such that the non-linear constitutive equation can be replaced by its linear approximation about the pre-impact state. In this case E^α is a tangent modulus evaluated at the pre-impact strain $\{(\partial x/\partial X)_-^\alpha - 1\} = 0$. To further simplify, consider the case when both rods have identical properties (i.e., $\rho_0 = \rho_0^\alpha$, $E = E^\alpha$, $\alpha = 1, 2$). Then

$$v = \frac{V^1 + V^2}{2} \quad ,$$

$$P = \rho_0 U (V^2 - V^1)/2 \quad ,$$

$$(U)^2 = E/\rho_0 \ . \tag{54}$$

In Eqs. (54) U is positive, and since consistency requires $V^1 - V^2 > 0$, P is compressive.

Thus for the one-dimensional case at least the problem of computing the post-impact state is easily achieved. The solution of (51) for the fully non-linear case can be automated as part of a numerical algorithm. Although this problem is trivial, it serves to indicate that the post-impact problem, the solution of which is essential in a numerical algorithm, is one of wave propagation.

In the analysis of higher dimensional bodies the solution of the post-impact problem becomes greatly complicated due to the geometric variety of impact conditions. However, considerable simplifications can be taken advantage of if one keeps in mind the nature of the discrete problem. For instance, if a certain portion of the boundaries of two bodies have coalesced in $\mathfrak{e}$, each interior point of $\mathfrak{e}$, at which the tangent plane is well defined, may be treated, to the first order, as a point on the mating surface of two impacting half-spaces. As long as time steps are kept small enough, the local behavior is well represented. The post-impact problem for the general case, analogous to (51), can be automated as part of the numerical algorithm, and for many simple cases can be solved explicitly.

With these notions in mind, let us return to the case of main interest in this report, namely three-dimensional continuum bodies. We shall consider only the case of a frictionless contact surface (Case II), and leave the solution of the post-impact problem for the no-slip case (Case I), which is more difficult, for future work. With the proper interpretations, the one-dimensional rod formulation (Eqs. (48-54)) suffices to completely characterize this case. This is so because no tangential motions or stresses may be communicated across a frictionless surface, and thus we need only consider the configuration of normal incidence. In this case the requisite constitutive functionals in (51) would be those relating P^{α}, the normal Piola stress, to the normal component of strain, holding all other components of strain fixed at the pre-impact values. For example, in the linear isotropic case, E^{α} (Young's modulus) in Eqs. (53,54) would be replaced by $\lambda^{\alpha}+2\mu^{\alpha}$ ($\lambda^{\alpha}, \mu^{\alpha}$ are the Lamé and shear moduli, respectively) and the propagation velocity would be that of dilatational waves.

PART II

A NUMERICAL SCHEME FOR ANALYSIS OF CONTACT-IMPACT PROBLEMS

8. Numerical Solution of Contact-Impact Problems

In performing numerical computations based on the above described variational formulation for contact-impact problems we have employed three distinct levels of approximation: (1) a spatial discretization of the bodies and contact surfaces, (2) a temporal discretization to determine the response of the discretized bodies, and (3) a numerical solution for the resulting system of nonlinear algebraic equations.

In the following sections we shall restrict our attention to the Hertzian contact problem described in Section 5. Significant numerical difficulties are encountered in the solution of impact problems; to complicate the problem further by introducing the additional steps necessary to determine the contact surface maps for the full kinematically nonlinear case is left for a future study. While this is a simple impact problem in terms of determining the contact surface and the full power of the preceding theory is neither necessary nor exploited in its solution, many of the features of the general problem are employed here.

9. Spatial Discretization of the Bodies and Contact Surface

The bodies $\mathcal{B}^1$ and $\mathcal{B}^2$ are discretized using standard finite element methods, (e.g., see [5]). In order to facilitate the computation of a discrete Hertzian contact surface the nodes of $\mathcal{B}^1$ are arranged so that they align with the nodes of $\mathcal{B}^2$. This is consistent with the notions of condition 3 of Section 5 and ensures that during determination of the approximation to the contact surface contiguous nodes of the two bodies will meet. Thus, the simulation of the contact surface is trivial. The development of a numerical model for Hertzian contact problems is based upon the form of Theorem II' which uses (47) for the integrand of $\mathcal{M}$. For numerical computations we introduce the displacement vector $\underset{\sim}{u}$ such that

$$\underset{\sim}{x} = \underset{\sim}{X} + \underset{\sim}{u} \quad . \tag{55}$$

For a compatible finite element displacement field the integrand of $\mathcal{M}$ can be approximated by taking $\eta^2(\underset{\sim}{z},t)$ as the product of $\mathcal{E}^2(t)$ and $\delta(\underset{\sim}{z}-\underset{\sim}{z}_i)$ (i.e., Dirac delta functions in space). This corresponds to taking η^2 as "concentrated nodal loads" which are the generalized forces of the contact pressure. With this discretization we can describe pseudo contact elements between each pair of candidate contact nodes. Let these nodes be denoted as $(\ )_i^\alpha$ and the generalized force as $(\mathcal{E}_i)^2$; then

$$\mathcal{M} = \int_0^t \sum_i (\mathcal{E}_i(t))^2 (u_{3i}^2(t) - u_{3i}^1(t) + X_{3i}^2 - X_{3i}^1)\,dt \tag{56}$$

where $\{i\}$ are the set of candidate contact nodes which span $\mathcal{E}$; u_{3i}^α are the nodal displacements in the z_3 direction and X_{3i}^α are the nodal coordinates of the candidate contact nodes.*

*We assume here that 3 is the direction nominally normal to the contact surfaces, e.g., see Fig. 5.

Use of the finite element method in Theorem II' with $\mathcal{M}$ given by (56) produces a set of nonlinear second order ordinary differential equations which together with the impenetrability conditions define the discretized contact impact problem. These equations take the form:

$$\underset{\sim}{M}\,\underset{\sim}{\ddot{u}} + \underset{\sim}{K}(\underset{\sim}{u}) = \underset{\sim}{R} \quad , \tag{57}$$

where $\underset{\sim}{M}$ is the usual finite element mass matrix, $\underset{\sim}{K}$ represents the elastic stiffness forces together with the contact terms, $\underset{\sim}{R}$ is the set of generalized forces resulting from boundary tractions and $\underset{\sim}{u}$ is the set of time dependent nodal displacements (which also include the $(\varepsilon_i)^2$). For inelastic materials Theorem II' can be extended by treating the first variation as a Galerkin method (principle of virtual work) and replacing the elastic constitutive model by more general theories, e.g., viscoelastic, elasto-plastic, viscoplastic, etc. In this case

$$\underset{\sim}{K}(\underset{\sim}{u}) \longrightarrow \underset{\sim}{K}(\underset{\sim}{u},\underset{\sim}{\dot{u}}) \tag{58}$$

in (57).

10. Temporal Discretization

A temporal discretization of the second order ordinary differential equations which result from a finite element spatial discretization of the contact-impact problem is accomplished herein by using the Newmark family of methods [6]. The Newmark family of methods is a one-step integration method with two free parameters which can be used to control stability and numerical damping. The method is essentially a difference method in time. The behavior of the method for linear elasto-dynamics problems is discussed in [6,7]. The algorithm is given by

$$\underset{\sim}{u}_{n+1} = \underset{\sim}{u}_n + \Delta t\, \dot{\underset{\sim}{u}}_n + (\tfrac{1}{2} - \beta)\Delta t^2 \ddot{\underset{\sim}{u}}_n + \beta \Delta t^2 \ddot{\underset{\sim}{u}}_{n+1} , \tag{59}$$

and $\dot{\underset{\sim}{u}}_{n+1} = \dot{\underset{\sim}{u}}_n + (1-\gamma)\Delta t\, \ddot{\underset{\sim}{u}}_n + \gamma \Delta t\, \ddot{\underset{\sim}{u}}_{n+1}$,

where $\underset{\sim}{u}_n = \underset{\sim}{u}(t_n)$, $\Delta t = t_{n+1} - t_n$, and β, γ are the two parameters. For linear problems $\gamma = .5 + \delta = .5$ produces no artificial viscosity and $\beta \geq \frac{1}{4}(1+\delta)^2$ produces unconditional stability (i.e., the method is stiffly stable). Such generalization is not possible for nonlinear problems and during solution it may be necessary to monitor the solution for any signs of instability. In (59) $\beta = 0$ produces an explicit method for $\underset{\sim}{u}_{n+1}$ and if $\underset{\sim}{M}$ is diagonal (lumped mass) with $\underset{\sim}{K}$ and $\underset{\sim}{R}$ independent of $\dot{\underset{\sim}{u}}$ the solution can be advanced without solving a large set of simultaneous equations; for all other cases the method is implicit and equations must be solved. Solution of $(59)_1$ for $\ddot{\underset{\sim}{u}}_{n+1}$ in terms of the solution at t_n and $\underset{\sim}{u}_{n+1}$ gives

$$\ddot{\underset{\sim}{u}}_{n+1} = \frac{1}{\beta \Delta t^2}(\underset{\sim}{u}_{n+1} - \underset{\sim}{u}_n) - \frac{1}{\beta \Delta t}\dot{\underset{\sim}{u}}_n - \left(\frac{1-2\beta}{2\beta}\right)\ddot{\underset{\sim}{u}}_n \tag{60}$$

which can also be used in $(59)_2$ to express the velocity in terms of the solution at t_n and $\underset{\sim}{u}_{n+1}$. Since in this process we divide by β and Δt it is no longer possible to consider zero β or zero time steps.

11. Solution of the Nonlinear Algebraic Problem

Use of the Newmark method in (57) (including (58)) yields the set of nonlinear algebraic equations:

$$\frac{1}{\beta\Delta t^2}\underset{\sim}{M}\,\underset{\sim}{u}_{n+1} + \underset{\sim}{K}(\underset{\sim}{u}_{n+1}, \underset{\sim}{u}_n, \dot{\underset{\sim}{u}}_n, \ddot{\underset{\sim}{u}}_n) = \underset{\sim}{R}_{n+1} + \underset{\sim}{M}\,\underset{\sim}{A}_n \quad , \tag{61}$$

where

$$\underset{\sim}{A}_n = \frac{1}{\beta\Delta t^2}\,\underset{\sim}{u}_n + \frac{1}{\beta\Delta t}\,\dot{\underset{\sim}{u}}_n + \left(\frac{1-2\beta}{2\beta}\right)\ddot{\underset{\sim}{u}}_n \quad .$$

A Newton-Raphson iterative solution to this set of equations can formally be constructed, giving:

$$\left(\frac{1}{\beta\Delta t^2}\underset{\sim}{M} + \partial_u\underset{\sim}{K} - \partial_u\underset{\sim}{R}\right)\Delta\underset{\sim}{u}^{(i)} = \underset{\sim}{R} - \underset{\sim}{K}(\underset{\sim}{u}_{n+1}^{(i)}, \underset{\sim}{u}_n, \dot{\underset{\sim}{u}}_n, \ddot{\underset{\sim}{u}}_n) - \underset{\sim}{M}\,\ddot{\underset{\sim}{u}}_{n+1}^{(i)} \quad , \tag{62}$$

where $\partial_u\underset{\sim}{R}$ is the effect of loads varying with the deformation and

$$(\partial_u\underset{\sim}{K})_{ij} = \partial K_i / \partial u_j \quad , \tag{63}$$

is the tangent stiffness matrix. The coefficient to $\Delta\underset{\sim}{u}^{(i)}$ is generally called the Jacobian matrix of the Newton-Raphson iteration. The solution is advanced by taking

$$\underset{\sim}{u}_{n+1}^{(i+1)} = \underset{\sim}{u}_{n+1}^{(i)} + \Delta\underset{\sim}{u}^{(i)} , \tag{64}$$

and iterating until a norm of the solution satisfies

$$\|\Delta\underset{\sim}{u}^{(i)}\| \leq \epsilon\,\|\underset{\sim}{u}_{n+1}^{(i)}\| \quad , \tag{65}$$

where ϵ is some small positive error tolerance. In the work reported here the norm $\| \ \|$ is taken as the Euclidian norm

$$\| \underset{\sim}{x} \| = \left(\sum_i x_i^2 \right)^{1/2} , \tag{66}$$

and the load vector $\underset{\sim}{R}$ is assumed to be independent of $\underset{\sim}{u}$. For stable elastic materials the resulting tangent stiffness is then symmetric and positive definite, consequently, standard direct solution methods normally employed in the solution of linear finite element problems can be used. For inelastic materials or deformation dependent loads the tangent stiffness may be asymmetric. In these cases some special methods may be necessary to effect a solution.

12. Discretized Impact Conditions

In the previous numerical development $\tilde{c}$ has been defined by discrete points which correspond to nodes along the boundaries of $\mathcal{B}^1$ and $\mathcal{B}^2$. When, during the course of advancing the solution in time, any one of these points violates the impenetrability condition a re-solution must be obtained in which the $(\mathcal{E}_i)^2$ are now non-zero and the u_3^* satisfy the impenetrability condition. Some control and monitoring are required to effect this in a computer program. In addition to satisfying these conditions, the impact relations denoted in Section 7 must be invoked. In the present study these conditions are applied to the solution at the end of a time step in which points first go into contact. Accordingly we compute from (50)*

$$\dot{u}_+ = \frac{\rho_0^2 U^2 \dot{u}_-^2 - \rho_0^1 U^1 \dot{u}_-^1}{\rho_0^2 U^2 - \rho_0^1 U^1} , \qquad (67)$$

and assign this value to the appropriate node of $\mathcal{B}^1$ and $\mathcal{B}^2$.

To determine the solution vector $\underset{\sim}{u}$ at t_{n+1} we have solved the set of equations (61). As described above the shock conditions are then used to determine the value of the velocity at time t_{n+1} for all points which have come into contact during the time interval. In order to get a consistent solution at these points we must modify the accelerations and contact force to reflect the shock conditions. This is accomplished by re-solving the equilibrium conditions of $\mathcal{B}^1$ and $\mathcal{B}^2$ at point i. The expanded forms of the appropriate equations are:

*The $(\ )_-$ denotes a value which is computed before impact, whereas $(\ )_+$ denotes the value after impact.

$$M^1\ddot{u}^1_- + K^1(\underset{\sim}{u}) + (\varepsilon_i)^2_- = R^1 \ ,$$

and

$$M^2\ddot{u}^2_- + K^2(\underset{\sim}{u}) + (\varepsilon_i)^2_- = R^2 \ . \tag{68}$$

For nodes which have come into contact we must enforce the condition on acceleration

$$\ddot{u}^1_+ = \ddot{u}^2_+ = \ddot{u}_+ \ , \tag{69}$$

and compute the contact force $(\varepsilon_i)^2_+$. The solution for these is obtained from

$$M^1\ddot{u}_+ + K^1(\underset{\sim}{u}) + (\varepsilon_i)^2_+ = R^1 \ ,$$

and

$$M^2\ddot{u}_+ + K^2(\underset{\sim}{u}) + (\varepsilon_i)^2_+ = R^2 \ .$$

These are two equations in two unknowns which can be solved for the $\ddot{u}_+$ and $(\varepsilon_i)^2_+$. If $K^\alpha(\underset{\sim}{u})$ is independent of velocity the stiffness forces and R^α will remain unchanged during the impact, hence we can solve the simpler problem

$$M^1\ddot{u}_+ + (\varepsilon_i)^2_+ = M^1\ddot{u}^1_- + (\varepsilon_i)^2_-$$

$$M^2\ddot{u}_+ - (\varepsilon_i)^2_- = M^2\ddot{u}^2_- - (\varepsilon_i)^2_-$$

whose solution is

$$\ddot{u}_+ = \frac{M^1\ddot{u}^1_- + M^2\ddot{u}^2_-}{M^1 + M^2} \ ,$$

and

$$2(\varepsilon_i)^2_+ = 2(\varepsilon_i)^2_- + M^1(\ddot{u}^1_- - \ddot{u}_+) - M^2(\ddot{u}^2_- - \ddot{u}_+) \ .$$

This completes the numerical specification of the solution at t_{n+1}; this solution process is now repeated for each of the succeeding time steps.

At this point it is important to compare the solution procedure for impact of a continuum discretized by a finite element method with the solution procedure for a physically discrete body, i.e., a body composed of mass points joined by massless elastic springs. Both problems may be described by algebraic equations of the form of (57). The impenetrability condition is also identical. The impact conditions, however, are different. For the discretized continuum the procedure is described above. The study of the impact of mass points is considered in elementary mechanics books, e.g. [8]. The impact of two mass points is described by impulsive motion such that at t_- the velocities of the two mass points are V_-^1 and V_-^2; after impact at time t_+, the two points have velocities V_+^1 and V_+^2. The two points will not in general stay in contact (i.e., $V_+^1 \neq V_+^2$) but will rebound. The conditions used to compute the V_+^1 and V_+^2 are: Balance of Momentum*

$$M^1 \{V^1\} + M^2 \{V^2\} = 0 , \tag{71}$$

and use of an equation involving the "coefficient of restitution", e :

$$\frac{V_+^2 - V_+^1}{V_-^1 - V_-^2} = e . \tag{72}$$

For $e=1$ energy is conserved whereas for $e=0$ the points "stick" and energy is dissipated. We must comment in passing that (72) is the energy

---

* $\{f(t)\} = f(t_+) - f(t_-)$.

equation in disguise. To see this we can write the jump conditions for energy as

$$\frac{1}{2} M^1 \{(V^1)^2\} + \frac{1}{2} M^2 \{(V^2)^2\} = \{\nu\} \quad . \tag{73}$$

The term $\{\nu\}$ can exist only if other energies are dissipated during the jump. We rewrite (73) by using

$$\frac{1}{2} \{(V^1)^2\} = [V^1] \langle V^1 \rangle \quad ,$$

where

$$\langle V^1 \rangle = \frac{1}{2} (V_+^1 + V_-^1) \quad . \tag{74}$$

Use of the momentum equation (71) then gives, after dividing by $M^1 \{V^1\}$

$$\langle V^1 \rangle - \langle V^2 \rangle = \frac{\{\nu\}}{M^1 \{V^1\}} \quad ,$$

or after recollecting terms and dividing by $(V_-^1 - V_-^2)$ we obtain:

$$\frac{V_+^2 - V_+^1}{V_-^1 - V_-^2} = 1 - \frac{\{\nu\}}{M^1 \{V^1\} (V_-^1 - V_-^2)} \quad . \tag{75}$$

The significance of the coefficient of restitution then is associated with the right hand side of (75).

It is clear from the above developments that the numerical simulation of the discretized continuum and the physically discrete system involve two distinct methods for treating the impact conditions. It is imperative then to associate the correct method for the problem at hand. In the present study we are interested in the impact of continua, and in this case we shall employ the discrete shock condition to effect the solution. This a priori assumes that the response we are computing involves a time scale associated with wave propagation problems. Consequently, we cannot expect the computation procedure for advancing the solution in time to be accurate if we take time steps greatly in excess of transit times through each body. In this context it may be important to consider an "explicit" time integration procedure in future work. The stability restrictions may be too severe to make this feasible.

PART III

FEAP 74 - A COMPUTER PROGRAM FOR SOLUTION OF CONTACT-IMPACT PROBLEMS

13. Development of a Contact-Impact Model for FEAP

In order to incorporate an ability to compute solutions to contact-impact problems using a finite element method as described above it is necessary to have available a computer program which can solve the nonlinear equations of motion given by (61). The computer program FEAP is a general program to solve finite element problems. The program has a capability of solving both quasistatic and dynamic problems and can incorporate several types of elements simultaneously. The nonlinear capabilities required for the solution of contact-impact problems have been incorporated into FEAP and currently includes the user options (see Input Instructions):

(1) Selection of quasistatic or dynamic option: The dynamic option will integrate the equations of motion using the one-step Newmark method to advance the solution in time. Quasistatic analysis is accommodated by any one-step algorithm. The algorithm employed is incorporated into each element routine and thus id defined by the developer of each element. Impact problems require description of the contact surface and wave speeds.

(2) Selection of the nonlinear method to advance the solution: Options include:

   (a) No iterations in each time step. Unbalanced forces at each time are added to the next time step.

   (b) Iterations in each time step to achieve a balance of force within each time step. In this option the user can select to reform the Jacobian matrix for each iteration or only at the first iteration in each time step.

In the impact problems solved to date it has been necessary to use the general form of the Newton-Raphson algorithm. This includes a complete formating and factoring of the Jacobian matrix for each iteration of each time step in the analysis. If the method described herein is to become computationally effective improvements in the computer program are paramount. Undoubtedly the most important aspect in reducing computer times is to introduce a substructuring system so that the highly nonlinear equations in the vicinity of the contact surface can be isolated from the remainder of the bodies. This will normally involve only a small number of equations in the total system of (62). The solution of a large finite element problem will generally concentrate the computer solution time in the forming and factoring of the tangent stiffness matrix. The fewer times that it is necessary to perform this costly step the more efficient the solution algorithm. Substructuring can be used then to restrict the part of the equations which must be formed and factored often, and thus greatly reduce the computer costs in analyzing impact problems.

The version of FEAP which can currently be used to analyze contact-impact problems includes, in addition to the nonlinear Newton-Raphson iterative algorithm, a new special contact-impact element and a new subroutine to describe impact surfaces and the discrete shock conditions described in Section 12. These are described in the following sections.

14. Contact Element for Hertzian Contact

The contact-impact element which has been developed is called ELMT05 and can be used along any coordinate direction. As developed it cannot be used along normals which are in non-coordinate directions. The development of the contact element assumes that within the framework of linear elasticity theory a node on $\mathcal{B}^1$ will impact on a node of $\mathcal{B}^2$. In using this contact element we shall assume that the contact surface on $\mathcal{B}^2$ is located at larger coordinate values than the contact surface of $\mathcal{B}^1$. The contact element is described by three nodes. Node 1 is associated with $\mathcal{B}^1$, Node 3 is associated with $\mathcal{B}^2$, and Node 2 is used as storage for the contact force $(\mathcal{E})^2$. The user can select the direction of contact motion by specifying the degree of freedom of the nodal unknowns to which the contact is to be measured; this must agree with the physical direction of the element (see Fig. 9). The degree of freedom for the contact element is specified during the MATERIAL data input and consists of a single card in I5 format. The element nodes are described along with all other elements according to Section 4 of the Input Instructions. The Node order as shown in Fig. 9 must be observed.

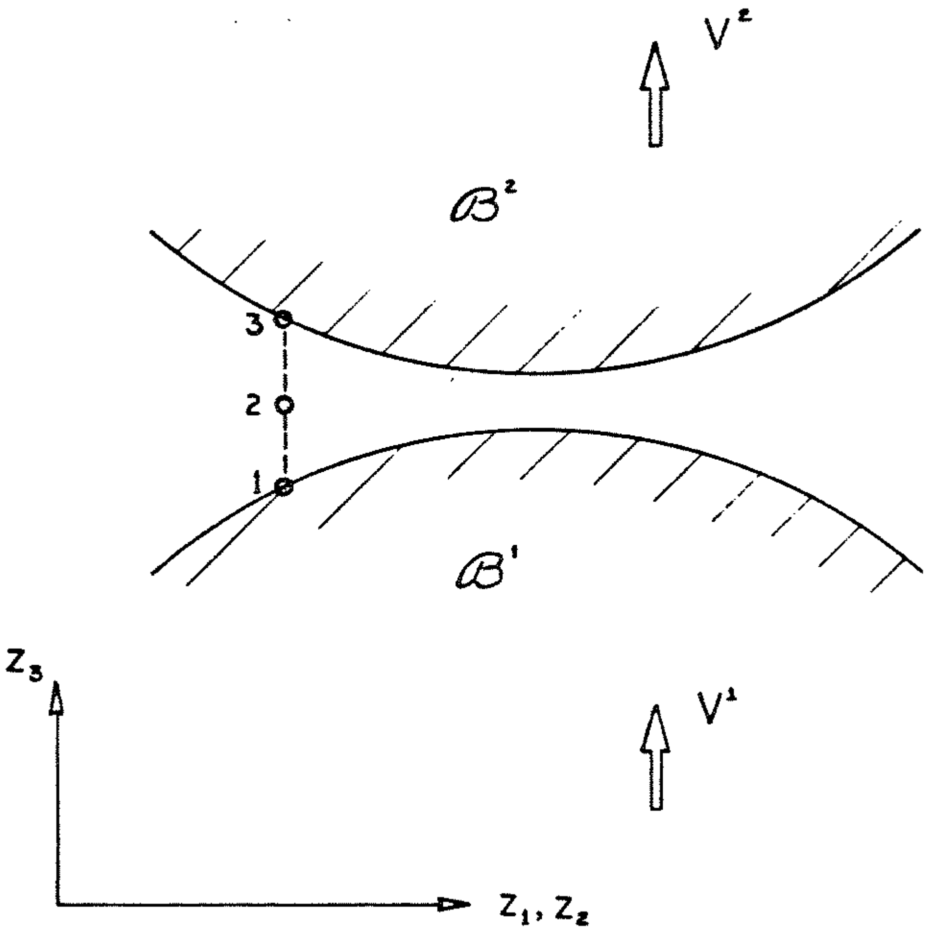


Figure 9

15. Impact Surface Description

The definition of the impact surface includes a list of all elements on the contact surface together with the degree of freedom describing the direction of contact motion (as described above). In addition, the product of mass density and wave velocity (always a positive number) for each body is input. This assumes, currently, that (1) each contact surface belongs to a linear material, and (2) the same material exists along all of the contact surface. This data need be prescribed only for impact problems, quasistatic contact problems do not require this data since no velocity or acceleration computations are performed in this class of problems. Data to be input for the impact surface is given in Table I.

Table I - Impact Surface Data

CARD 1) (6X,A6)		
COL. 7 to 12		Must contain CONTAC
CARD 2) (2F10.0)		
COL. 1 to 10	ρU	of body 1
COL. 11 to 20	ρU	of body 2
CARD 3) (I5)		
COL. 1 to 5		NLIST, number of elements on contact surface
CARD 4) (2I5)		
Repeat NLIST times		
COL. 1 to 5		Contact element number
COL. 6 to 10		Degree of freedom of this contact element

16. Example Problems

Two example problems are included to illustrate the characteristics of the methodology and the associated computer program described above for Hertzian contact problems. The first problem is a quasistatic contact problem which is used to demonstrate the ability of the computer program to compute an evolving contact surface. The second problem will demonstrate the ability of the program to properly model the temporal response of an impact problem.

To model a problem in which a contact surface will change under different load levels we consider two beams with an initial parabolic curvature. A symmetric configuration is analyzed and the resulting finite element model is shown in Fig. 10. Each element is nominally one unit by one unit. The gap at the load end is initially 0.5 units. The material properties used are $E = 500$ and $\nu = 0$. The load P is applied as shown and allowed to increase linearly in time. The problem then is to determine the contact surface at various load levels. In order to eliminate a singularity in the system of equations it was necessary to permanently attach the two nodes at the symmetry axis of the contact surface. All other nodes along the boundaries between the two bodies are assumed to be possible contact points and contact elements are assigned between vertical nodal pairs. The load was varied from 0.2 to 0.8 in increments of 0.1 and the computed contact surface and forces were computed. These are given in Table II. The deformed shape at a load of 0.4 is also shown in Fig. 10 as a dotted form. The attached node at the center has influenced the solution at loads above 0.3 since the contact pressure there is tensile (negative). The force is small and should not greatly affect the actual contact region computed. As the load increases the

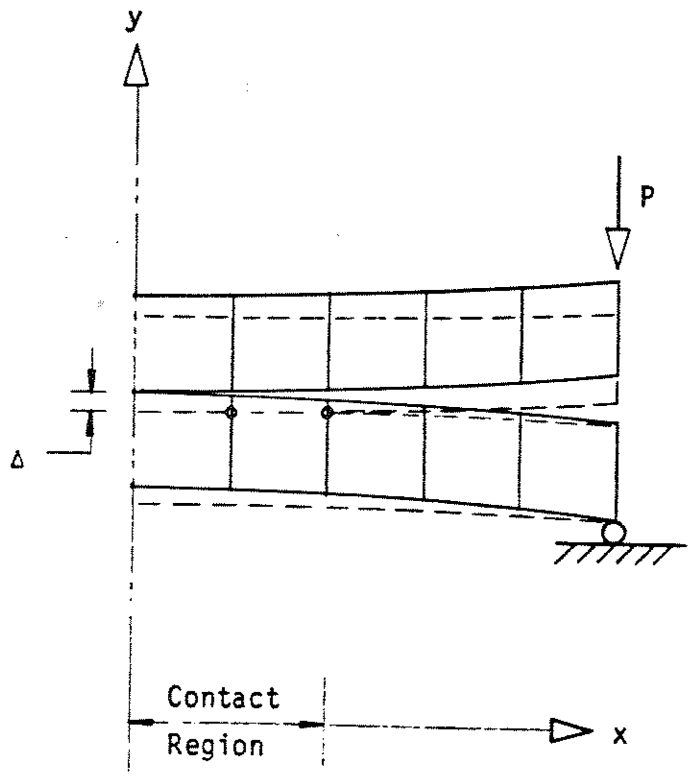


Figure 10

contact surface moves toward the load. This is conceptually correct since if the beams were modeled according to Euler-Bernoulli theory the contact force would be a point load which gradually moves from the center to the outer edge according to the relationship (using the above values for sizes and material properties)

$$X = 5\left(1 - \frac{1}{6P}\right) \quad .$$

This relation predicts that the contact point will be non-zero only after P exceeds 1/6. The finite element model is in qualitative agreement with this beam theory, but since shear deformations are included the finite element solution gives a distributed load on the contact surface. It is interesting to also note that the contact force over the center of the beams is zero, just as in the beam theory.

Table II - Contact Forces

LOAD	X-COORDINATE 0	1	2	3	4	5	BEAM THEORY-X
0.2	0.2	-	-	-	-	-	0.83
0.3	.07	.23	-	-	-	-	2.22
0.4	-.01	.07	.34	-	-	-	2.92
0.5	-0.00	.02	.23	.25	-	-	3.33
0.6	-.01	-	.09	.51	.01	-	3.61
0.7	-.01	-	.07	.45	.19	-	3.81
0.8	-.01	-	.05	.38	.38	-	3.96

This problem demonstrates that the computer program can model the evolution of a contact surface. Of particular importance is to note that as the load increases the program can both attach and detach a contact point. This is an essential requirement for the analysis of impact problems as is shown in the next problem.

As a simple example we consider the impact against a rigid wall of a finite, linear elastic rod traveling at constant velocity. The rod has a modulus of elasticity E of 100, and a mass density ρ of 0.1. The arrival velocity is taken to be 0.1 (units may be assigned in any convenient system). The rod is taken to be 10 units long and is divided into 10 elements plus one contact element as shown in Fig. 11. At time zero the rod is just arriving at the wall. The exact solution predicts a contact duration of 0.2 time units. This corresponds to the time required for a wave to travel from the contact point to the left end and back to the contact point at which time the rod will part from the wall. The problem was analyzed using FEAP with time steps of 0.01 unit (transit time across an element) and the rod remains in contact until time 0.20 units and has rebounded at time 0.21. Thus the program can predict accurately the contact duration of the rod. The finite element solution obtained is compared with the exact solution in Fig. 12. The agreement of stresses and contact force is good. The largest discrepancy exists in defining the shock front, which is "smeared" by the finite element method and ordinary differential equation solution method used here. This is the same type of solutions which are commonly obtained with numerical solutions of this type even without impact. Solutions such as the impact shocks generated are probably one of the most difficult responses to accurately calculate by a finite element method.

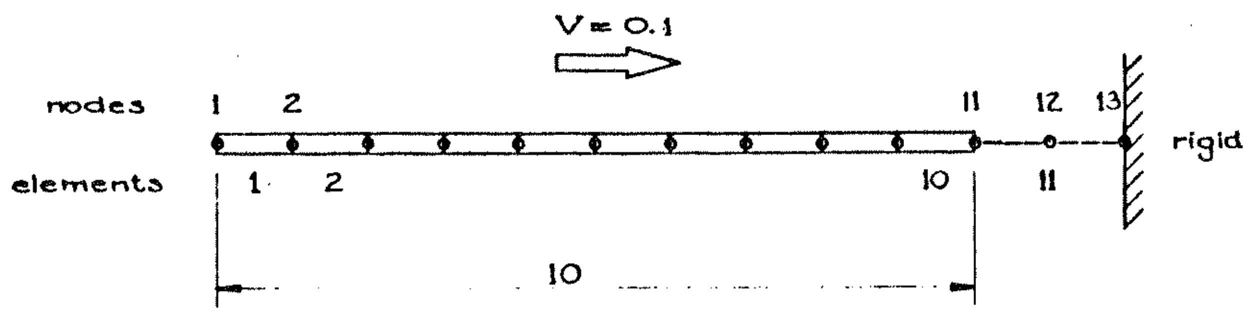
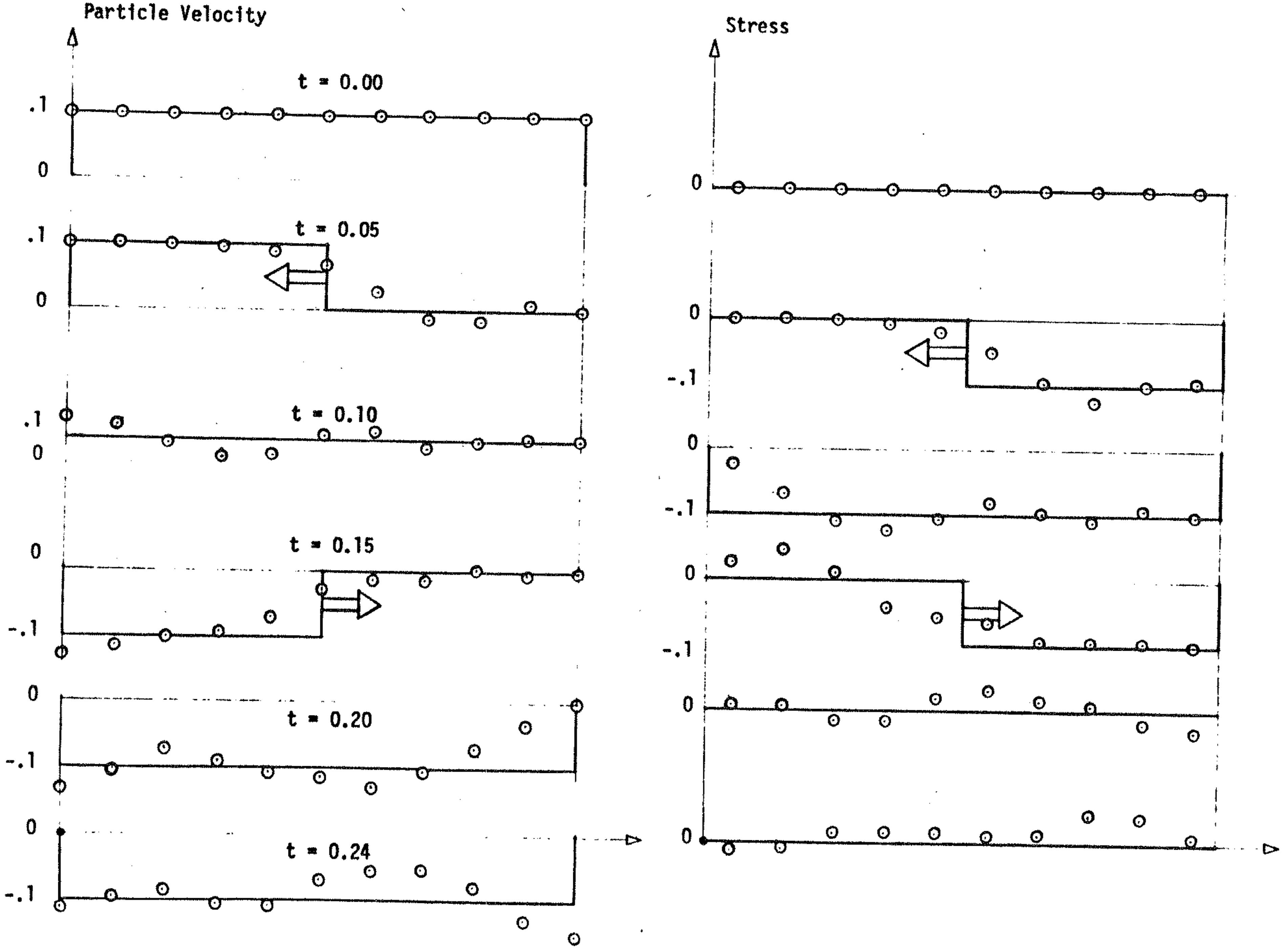


Figure 11

Comparison of Numerical Data with Exact Solution for Bar Impacting Rigid Wall



17. Closure and Recommendations for Future Work

In the preceding sections we have presented a theory for contact-impact problems together with the numerical development of a Hertzian contact-impact model. The computer program FEAP 74 has been modified to include the model and has successfully solved a contact problem and an impact problem. The work reported herein must be considered initiatory; the general theories and their numerical implementation have not been completed. The problem is of such a complicated nature and the literature existing prior to this study was so meager that we consider it fitting to document the work completed thus far.

We have attempted to qualify each stage of the development throughout the report, however, it may be fitting to reiterate future work which we consider to be essential for numerical models to be effective and efficient tools for predictive analyses.

(1) The restriction of Hertzian type contact must be removed. This involves the non-trivial task of finding appropriate numerical methods to handle the $\underset{\sim}{\chi}^{\alpha}$ maps.

(2) Improved methods for solving the set of nonlinear algebraic equations must be found. We have suggested two methods which should be considered: (a) Substructure the problem about the contact regime so that a more efficient forming and factoring of the tangent stiffness can be performed; and (b) Since the impact problem is a wave propagation problem an explicit time integration of the equations of motion should be explored. In complex situations the explicit integration method may have severe stability limitations which could make it unacceptable.

(3) Methods of utilizing the shock conditions need to be explored further. We have noted some peculiar anomalies when the bodies separate. These appear to be caused by a shock like separation phenomena.

(4) When the wave propagation property of the impact problem is ignored by taking time steps greatly in excess of the transit times in a body the computed response is meaningless. Under such situations the bodies rebound within a single time step. Currently the rebound velocity is much too large. When the shock conditions are used for a class of problems where the response desired is in the target instead of in the impactor, it may be expedient to take a large time step. Methods should be explored to accomplish this capability.

The above recommendations for future work should in no way minimize what has been accomplished by the present study. For the first time a contact-impact theory in the form of a variational problem has been presented in a general form. This formulation was motivated by the fact that numerical solutions would be obtained by a finite element method. In addition the necessary foundation for the numerical solution has been thought out and within this context a computer program has been developed for Hertzian contact-impact problems.

The implementations considered here have produced results which are hopeful signs for the eventual success of the more general impact problems.

References

1. S. Nemat-Nasser, On variational methods in finite and incremental elastic deformation problems with discontinuous fields, Quarterly Appl. Math., July 1972, 143.

2. H. Hertz, Über die Beruchrüng fester elastischer Körper, J. reine angew. Math. (Crelle), 92, 1881, 155.

3. W. Goldsmith, Impact, Edward Arnold, London, 1960.

4. Kowalik and Osborne, Methods for Unconstrained Optimization, Am. Elsevier, New York, 1968.

5. O. C. Zienkiewicz, The Finite Element Method in Engineering Science, McGraw-Hill, New York, 1971.

6. N. M. Newmark, A method of computation for structural dynamics, J. Engr. Mech. Division, ASCE, V. 85, No. EM3, July 1959, 67-94.

7. G. L. Goudreau and R. L. Taylor, Evaluation of numerical integration methods in elasto dynamics, Computer Methods in Appl. Mech. and Engng., 2, No. 1, 1972, 67-97.

8. J. L. Meriam, Dynamics, John Wiley & Son, Inc., New York, 1966.

APPENDICES

A. Input Instructions for Contact/Impact Problems

In order to analyze contact/impact problems in FEAP, users must prepare the data for a time dependent analysis. This will include the following Data Type Identification Cards (see Section 1, Appendix B):

FEAP 74
MATERIAL
NODAL
ELEMENT
CONTACT (for impact problems only)
loadings
INITIAL CONDITIONS (if non-zero)

and

VISCOE (for quasi-static contact problems)

or

IMPLICIT (for impact problems)

In performing the necessary solution to (62) the full Newton-Raphson method must be employed; this is controlled by the data in col. 76-80 of the first card following the VISCOE or IMPLICIT card, and consists of a negative number (negative uses full Newton-Raphson iteration with the absolute value of the number giving the number of iterations to be performed before going to the next time step). As an example of the required input data Table A shows the input data used for the impact problem reported in Section 16.

14.19.14 OUTPUT

```
    FEAP74 * * ONE DIMENSIONAL BAR CONTACT PROBLEM
   1      X
   1      U
   3
  13 NODAL
   1                      0.
  11                      10.
  13             111111   10.
  11 ELEMENTS
   1    1                 1  1
  1   2
  11    2
 11  12  13
   3 INITIAL CONDITIONS
    DISPLACEMENT
    VELOCITY
    ACCELERATIONS
   1    1                 .1
  11    0                 .1


  2 MATERIALS
  1 ELM03
  100.     1.         .01
  2 ELM05
  1
    CONTACT
  1        1.0        1.0E+30
 11     1

  1 IMPLICIT
  1.0        5    1    1   13    1   11    0    0   0.25                 -3
  .01       25    1    1   13    1   11    0    0   0.25                 -3
    STOP
  .01       10    1    1   13    9   11    0    0   0.25                 -3
```

B. Input Instructions for FEAP 74

The input instructions for the description of a finite element mesh, together with the initial and boundary conditions, is described by subroutine MANUAL listed on the following pages. The input of the contact surface for impact problems is described in Table I of this report.

The description of material properties for the contact element is described in Section 14. For material properties for other elements in FEAP special input instructions must be supplied.

```
      SUBROUTINE MANUAL                                                   MAN  1C
C                                                                         MAN  2C
C**** USER INSTRUCTIONS AND INPUT FORMATS FOR FEAP-74  *****************   MAN  3C
C                                                                         MAN  4C
C                                                                         MAN  5C
C     FINITE ELEMENT ANALYSIS PROGRAM                                     MAN  6C
C                                                                         MAN  7C
C      ****** I N D E X ******                                            MAN  8C
C                                                                         MAN  9C
C  SECTION    WORD       C O N T E N T                                    MAN 10C
C                                                                         MAN 11C
C    1.0                 DATA DIRECTORY WORDS                             MAN 12C
C    2.0      FEAP74     PROBLEM INITIATION AND CONTROL CARDS             MAN 13C
C    2.1      REMARK     REMARK CARD                                      MAN 14C
C    2.2      TITLE      TITLE CHANGE                                     MAN 15C
C    2.3      STOP       EXECUTION TERMINATION                            MAN 16C
C    3.0      MATERI     MATERIAL CHARACTERIZATION                        MAN 17C
C    4.0      NODAL      SEQUENTIAL NODAL GENERATION                      MAN 18C
C    4.1      GENERA     NON-SEQUENTIAL NODAL GENERATION                  MAN 19C
C    4.2      BOUNDA     BOUNDARY CODE INPUT                              MAN 20C
C    4.3      POLAR      POLAR TO CARTESIAN CONVERSION                    MAN 21C
C    5.0      ELEMEN     ELEMENT GENERATION                               MAN 22C
C    5.1      BLOCK      BLOCK MESH GENERATION                            MAN 23C
C    6.0      VECTOR     USER VECTOR INPUT                                MAN 24C
C    6.1      INITIA     INITIAL CONDITION                                MAN 25C
C    7.0      FORCE      GENERALIZED NODAL FORCE                          MAN 26C
C    7.1      BLOADS     SURFACE LOADS                                    MAN 27C
C    7.2      ELOADS     ELEMENT LOADS                                    MAN 28C
C    7.3                 PROPORTIONAL LOADS                               MAN 29C
C    8.0      SOLVE      INITIATION OF STATIC SOLUTION                    MAN 30C
C             RESOLV     STATIC SOLUTION                                  MAN 31C
C    8.1      EXPLIC     EXPLICIT DYNAMIC INTEGRATION                     MAN 32C
C    8.2      IMPLIC     IMPLICIT INTEGRATION                             MAN 33C
C             VISCOE     IMPLICIT TIME INTEGRATION                        MAN 34C
C    8.3      MESH       MESH CHECK                                       MAN 34C
C             PLOT       PLOT MESH                                        MAN 34C
C    8.4      FOURIE     FOURIER SERIES HARMONICS                         MAN 35C
C    9.0                 OUTPUT CONTROL                                   MAN 35C
C                                                                         MAN 36C
C        FEAP74 IS A GENERAL (F)INITE (E)LEMENT (A)NALYSIS (P)ROGRAM      MAN 37C
C        WHICH FURNISHES TO THE USER MESH INPUT/OUTPUT, ELEMENT ASSEMBLY  MAN 38C
C        AND SOLUTION OF EQUATIONS (LINEAR, IMPLICIT AND EXPLICIT TIME    MAN 39C
C        DEPENDENT, NONLINEAR), PRESCRIBED GENERALIZED NODAL FORCES,      MAN 40C
C        PRESCRIBED NODAL AND ELEMENT DATA, AND OUTPUT OF THE             MAN 41C
C        GENERALIZED DISPLACEMENTS AND FORCES.  ELEMENT MATRICES FOR TWO  MAN 42C
C        AND THREE DIMENSIONAL LINEAR ELASTICITY, SHELLS, PLATES, AND     MAN 43C
C        FIELD (LAPLACE EQUATION) PROBLEMS ARE AVAILABLE.  ALTERNATIVELY  MAN 44C
C        USERS MAY SUPPLY THEIR OWN ELEMENT LIBRARY BY PROVIDING A        MAN 45C
C        SUBROUTINE CALLED ELMTNN,  WHERE NN IS A TWO DIGIT NUMBER        MAN 46C
C        (01-10), IDENTIFYING THE ELEMENT SUBROUTINE. EACH ELEMENT        MAN 47C
C        SUBROUTINE HAS AT LEAST FOUR BASIC FUNCTIONS WHICH ARE           MAN 48C
C        DELINEATED BY A SWITCHING PARAMETER, ISW, IN THE SUBROUTINE.     MAN 49C
C                                                                         MAN 50C
```

```
C        ELMTNN(N,MA,NDIM,NDF,NEL,NEL1,NSTF,NSIZV,NVEC,MCT,DM,D,XYZ,        MAN 51C
C        IX,F,FORCE,ESTIF,U,VECT,ISW)                                       MAN 52C
C                                                                           MAN 53C
C                    N IS ELEMENT NUMBER.                                   MAN 54C
C                    MA IS THE MATERIAL NUMBER.                             MAN 55C
C                    NDIM IS SPATIAL DIMENSION, 1,2,OR 3.                   MAN 56C
C                    NDF IS NUMBER OF DEGREES OF FREEDOM PER NODE.          MAN 57C
C                    NEL IS THE NUMBER OF EXTERNAL NODES PER ELEMENT        MAN 58C
C                    NEL1 IS DIMENSION OF ELEMENT PROPERTY ARRAY.           MAN 59C
C                    NSTF IS THE SIZE OF THE ELEMENT STIFFNESS.             MAN 60C
C                    NSIZV IS THE SIZE OF UTILITY VECTORS.                  MAN 61C
C                    NVEC IS THE NUMBER OF UTILITY VECTORS.                 MAN 62C
C                    MCT IS A PRINTER LINE COUNTER.                         MAN 63C
C                    DM IS A PARAMETER FOR MATERIAL IDENTIFICATION.         MAN 64C
C                    D(L,1) IS MATERIAL PROPERTY MATRIX (63 CELLS).         MAN 65C
C                    XYZ(NDIM,1) ARE NODAL COORDINATES.                     MAN 66C
C                    IX(NEL1,1) ARE ELEMENT  PROPERTIES, NODES, ETC.        MAN 67C
C                    F(NDF,1) ARE NODAL GENERALIZED FORCES.                 MAN 68C
C                    FORCE(NSTF,2) IS ELEMENT FORCE VECTOR TO BE            MAN 69C
C                    COMPUTED. COLUMN 2 IS LUMPED MASS                      MAN 70C
C                    ESTIF(NSTF,NSTF) IS ELEMENT MATRIX TO BE               MAN 71C
C                    COMPUTED.                                              MAN 72C
C                    VECT(NSIZV,1) ARE PRESCRIBED NODAL OR ELEMENT          MAN 73C
C                    QUANTITIES. TEMPERATURES ETC.                          MAN 74C
C                    U(NDF,1) IS SOLUTION VECTOR.                           MAN 75C
C                    ISW IS SWITCHING PARAMETER,                            MAN 76C
C                                                                           MAN 77C
C                        ISW=1, ** MATERIAL CHARACTERIZATION**              MAN 78C
C                        ISW=2, ** CHECK ELEMENT FOR POSITIVE AREA *        MAN 79C
C                        ISW=3, ** ELEMENT STIFFNESS AND                    MAN 80C
C                                  LOAD COMPUTATION**                       MAN 80C
C                        ISW=4, ** ELEMENT STRESSES AND PRINTOUT**          MAN 81C
C                        ISW=5, ** ELEMENT LOAD COMPUTATION ONLY**          MAN 82C
C                        ISW=6, ** NONLINEAR GENERALIZED FORCES**           MAN 83C
C                        OTHER ISW MAY BE USED FOR SPECIAL PURPOSES.        MAN 84C
C                                                                           MAN 85C
C        USERS CAN GENERATE SURFACE LOADINGS BY PROVIDING SLDNN             MAN 86C
C        SUBROUTINES (WHERE NN IS A TWO DIGIT NUMBER BETWEEN 01 AND 05)     MAN 87C
C        THAT SPECIFY THE LOAD ROUTINE.  THE SUBROUTINE IS ACCESSED BY      MAN 88C
C        THE CALL TO                                                        MAN 89C
C                                                                           MAN 90C
C        SLDNN(NDIM,NDF,NDP,NPRES,IPRES,PR,XYZ,FS)                          MAN 91C
C                                                                           MAN 92C
C                    WHERE IN ADDITION TO QUANTITIES DEFINED ABOVE          MAN 93C
C                    FOR ELMTNN,                                            MAN 94C
C                    NDP IS THE DIMENSION OF LOADED SURFACE                 MAN 95C
C                    NPRES IS NUMBER OF LOADED NODES (MAX 8)                MAN 96C
C                    IPRES(8) ARE NODE NUMBERS OF LOADED NODES.             MAN 97C
C                    PR(8) ARE LOAD VALUES AT CORRESPONDING IPRES           MAN 98C
C                        NODES.                                             MAN 99C
C                    FS(6,8) ARE THE COMPUTED GENERALIZED (NODAL)           MAN100C
C                        FORCES FOR EACH DEGREE OF FREEDOM AT EACH          MAN101C
C                        IPRES NODE.                                        MAN102C
C                                                                           MAN103C
```

```
C       SEE SECTION 7.1 FOR DATA INPUT DETAILS.                                MAN104C
C                                                                              MAN105C
C                                                                              MAN106C
C       INTEGRATION TABLE IS ACCESSED BY THE CALL                              MAN107C
C                                                                              MAN108C
C         CALL  INTEGL(LIM,NC1,NDIM,LINT,STUW)                                 MAN109C
C                                                                              MAN110C
C                      STUW(4,M) INTEGRATION POINTS AND WEIGHTS.               MAN111C
C                      **NOTE** M MUST BE SET EXPLICITLY AND BE LARGER         MAN112C
C                      THAN OR EQUAL TO LINT.                                  MAN113C
C                      LINT - RETURNS WITH NUMBER INTEGRATION POINTS.          MAN114C
C                      NC1 = 0  RETURNS GAUSS POINTS AND WEIGHTS IN            MAN115C
C                           STUW.                                              MAN116C
C                           LIM = 1 TO 5 IS NUMBER OF GAUSS POINTS/DIR-        MAN117C
C                           ECTION.                                            MAN118C
C                      NC1 = 1  RETURNS A SPECIAL 3-D GAUSS FORMULA.           MAN119C
C                           SET LIM = 1 FOR 6 PT. CUBIC ACCURACY               MAN120C
C                           SET LIM = 2 FOR 14 PT. QUINTIC ACCURACY.           MAN121C
C                      NC1 = 2  RETURNS TRIANGULAR INTEGRATION FORMULA         MAN122C
C                           SET LIM = 1 FOR 1 PT. LINEAR ACCURACY.             MAN123C
C                           SET LIM = 2 FOR 3 PT. QUADRATIC ACCURACY.          MAN124C
C                           SET LIM = 3 FOR 7 PT. QUARTIC ACCURACY.            MAN125C
C                                                                              MAN126C
C                                                                              MAN127C
C     1.) DATA TYPE IDENTIFICATION CARDS (I5,1X,12A6).                         MAN128C
C                                                                              MAN129C
C       EACH DATA SEGMENT IS PREDEDED BY A CARD WHICH IDENTIFIES THE           MAN130C
C       TYPE OF DATA AND LIMITS ON THE AMOUNT OF DATA WHICH IMMEDIATELY        MAN131C
C       FOLLOWS THE CARD.  EXCEPT AS NOTED THE DATA SEGMENTS MAY APPEAR        MAN132C
C       IN ANY ORDER.  THE IDENTITY CARDS MAY ALSO AID THE USER IN             MAN133C
C       INTERPRETTING THE INPUT DATA CARDS.  AS SUPPLIED THERE ARE             MAN134C
C       TWENTY-FIVE DIFFERENT DATA IDENTIFICATION CARDS.  THESE ARE            MAN135C
C                                                                              MAN136C
C       COL  7 TO 12    IDENTITY(RESTRICTIONS)                                 MAN137C
C                                                                              MAN138C
C       FEAP74          START OF EACH PROBLEM (MUST PRECEDE ALL OTHER          MAN139C
C                       DATA).                                                 MAN140C
C       TITLE           CHANGE OUTPUT PAGE HEADINGS                            MAN141C
C       REMARK          COMMENTS ON OUTPUT                                     MAN142C
C       MATERI          MATERIAL CHARACTERIZATION.                             MAN143C
C       NODAL           NODAL CARDS                                            MAN144C
C       POLAR           POLAR CONVERSION. (PRECEDE BY NODAL,GENERA, OR         MAN145C
C                       BLOCK )                                                MAN146C
C       ELEMEN          ELEMENT CONNECTION CARDS.                              MAN147C
C       GENERA          GENERATE NODES IN A LINEAR PATH BY ANY INCREMNT        MAN148C
C       BLOCK           GENERATE ALL MESH DATA (BOTH NODAL AND ELEMENT)        MAN149C
C                       FOR A 2,(OR 3)DIMENSIONAL REGION WHOSE BOUNDARY        MAN150C
C                       MAY BE DEFINED BY 4(8)OR 8(20)COLLOCATED POINTS        MAN151C
C       BOUNDA          BOUNDARY CODE PRESCRIPTION (PRECEDE BY NODAL OR        MAN152C
C                       GENERATE OR BLOCK)                                     MAN153C
C       VECTOR          PRESCRIBED NODAL OR ELEMENT DATA (PRECEDE BY           MAN154C
C                       NODAL, OR GENERA AND ELEMEN, OR BLOCK)                 MAN155C
C       FORCE           NODAL GENERALIZED FORCES (PRECEDE BY NODAL OR          MAN156C
C                       GENERA OR BLOCK).                                      MAN157C
```

```
C       BLOADS         SURFACE LOADINGS (SAME AS FORCE).                       MAN158C
C       ELOADS         ELEMENT LOADINGS (SAME AS FORCE).                       MAN159C
C       MESH           CHECK CONSISTENCY OF MESH ONLY (SAME AS SOLVE)          MAN160C
C       PLOT           PLOT MESH (SAME AS SOLVE)                               MAN161C
C       INITIA         INITIAL CONDITION PRESCRIPTION FOR DYNAMIC              MAN162C
C                      ANALYSIS (PRECEDE BY NODAL, GENERA OR BLOCK)            MAN163C
C       SOLVE          COMPLETE FORMULATION AND SOLUTION FROM ELEMENTS         MAN164C
C                      (PRECEDE BY MATERI,NODAL OR GENERA, AND ELEMEN          MAN165C
C                      OR PRECEDE BY MATERI AND BLOCK )                        MAN166C
C       RESOLV         USE PREVIOUS PROBLEM DESCRIPTION WITH NEW LOAD          MAN167C
C                      ONLY (PRECEDE BY SOLVE AND NEW LOADING CARDS).          MAN168C
C       EXPLIC         DYNAMIC SOLUTION BY EXPLICIT INTEGRATION, (SAME         MAN169C
C                      AS SOLVE)                                               MAN170C
C       IMPLIC         IMPLICIT INTEGRATION OF DYNAMIC PROBLEMS                MAN171C
C                      (PRECEDE BY SAME DATA AS FOR SOLVE)                     MAN172C
C       VISCOE         QUASI-STATIC LINEAR VISCOELASTIC INTEGRATION            MAN173C
C                      (PRECEDE BY SAME DATA AS FOR SOLVE)                     MAN174C
C       FOURIE         FOURIER COMPOSITION (SAME AS SOLVE)                     MAN175C
C       ADDUP          ACCUMULATE FOURIER SOLUTION (AFTER FOURIE)              MAN176C
C       STOP           NORMAL EXIT (MUST FOLLOW ALL DATA)                      MAN177C
C                                                                              MAN178C
C       ***NOTE*** EACH IDENTIFIER IS PUNCHED STARTING IN COL 7 (LEFT          MAN179C
C           JUSTIFIED).                                                        MAN180C
C                                                                              MAN181C
C     EXCESS CARDS MAY EXIST BETWEEN EACH SECTION OF DATA, HOWEVER,            MAN182C
C     THE DATA TO BE USED MUST IMMEDIATELY FOLLOW THE TYPE CARD AND            MAN183C
C     MUST BE IN PROPER ORDER.  NO PARTICULAR ORDER OF THE TYPE                MAN184C
C     CARDS IS NECESSARY EXCEPT THAT THE FEAP74 CARD MUST ALWAYS BE            MAN185C
C     THE FIRST CARD IN EACH SET OF DATA, AND RESTRICTIONS MUST BE             MAN186C
C     OBSERVED.                                                                MAN187C
C                                                                              MAN188C
C                                                                              MAN189C
C    2.) PROBLEM INITIATION AND CONTROL CARDS                                  MAN190C
C                                                                              MAN191C
C       CARD 1. (6X,12A6)                                                      MAN192C
C                                                                              MAN193C
C       COL  7 TO 12    MUST CONTAIN WORD FEAP74                               MAN194C
C       COL 13 TO 78    OUTPUT PAGE HEADER                                     MAN195C
C                                                                              MAN196C
C       CARD 2. (I5,1X,3A6)                                                    MAN197C
C                                                                              MAN198C
C       COL  1 TO 5     NDIM - SPATIAL DIMENSION OF PROBLEM (1 TO 3)           MAN199C
C       COL  7 TO 12    NAMES TO BE PRINTED AS OUTPUT HEADERS TO               MAN200C
C       COL 13 TO 18    COORDINATES - IF BLANK SET TO 1,2,3 AS NEEDED.         MAN201C
C       COL 19 TO 24                                                           MAN202C
C                                                                              MAN203C
C       CARD 3. (I5,1X,6A6)                                                    MAN204C
C                                                                              MAN205C
C       COL  1 TO  5    NDF - NUMBER OF UNKNOWNS PER NODE (1 TO 6)             MAN206C
C       COL  7 TO 12    NAMES TO BE PRINTED AS OUTPUT HEADERS OF THE           MAN207C
C       COL 13 TO 18    GENERALIZED DISPLACEMENTS AND FORCES - IF              MAN208C
C           ........    BLANK SET TO 1,2,3,4,5,6 AS NECESSARY                  MAN209C
C       COL 37 TO 42                                                           MAN210C
C                                                                              MAN211C
```

```
C        CARD 4. (6I5,5F10.0)                                         MAN212C
C                                                                     MAN213C
C        COL  1 TO 5     NEN - MAXIMUM NUMBER OF NODES CONNECTED TO ANY MAN214C
C                        ELEMENT (1 TO 20).                           MAN215C
C        COL  6 TO 10    NEXTRA - INCREASES ELEMENT MATRIX SIZE FROM  MAN216C
C                        NDF*NEN TO NDF*NEN + NEXTRA                  MAN217C
C        COL 11 TO 15    IREC - COMPUTE GENERALIZED FORCE CHECK IF    MAN218C
C                        NONZERO (FOR TIME INVARIANT ANALYSIS ONLY)   MAN219C
C        COL 16 TO 20    MBAN - MAXIMUM EXPECTED BANDWIDTH. DEFAULT IS MAN220C
C                        SET TO 100. USED AS AN ERROR CHECK TO PREVENT MAN221C
C                        RUNNING WITH AN OBVIOUS ERROR.               MAN222C
C        COL 21 TO 25    IBUF - BUFFER SIZE FOR STORAGE OF HISTORY    MAN223C
C                        EFFECTS IN TIME DEPENDENT ANALYSIS. DEFAULT IS MAN224C
C                        IBUF = ISZDT/20                              MAN225C
C        COL 26 TO 30    NC1  - USER INTEGER CONSTANT                 MAN226C
C        COL 31 TO 40    CON1 - USER DEFINED CONSTANT                 MAN227C
C        COL 41 TO 50    CON2 - USER DEFINED CONSTANT                 MAN228C
C        COL 51 TO 60    CON3(1) - USER DEFINED CONSTANT              MAN229C
C        COL 61 TO 70    CON3(2) - USER DEFINED CONSTANT              MAN230C
C        COL 71 TO 80    CON3(3) - USER DEFINED CONSTANT              MAN231C
C                                                                     MAN232C
C      2.1)  *REMARK* USER COMMENTS ON OUTPUT. (6X,12A6)              MAN233C
C                                                                     MAN234C
C        SUBSEQUENT CARDS                                             MAN235C
C                                                                     MAN236C
C        COL  7 TO 12    MUST CONTAIN REMARK                          MAN237C
C        COL 13 TO 78    STATEMENTS TO BE OUTPUT , USE AS MANY REMARK MAN238C
C                        CARDS AS DESIRED. INSERT BEFORE ANY TYPE CARD. MAN239C
C                                                                     MAN240C
C      2.2) TITLE CHANGE ON OUTPUT (6X,12A6)                          MAN241C
C                                                                     MAN242C
C        COL  7 TO 12    MUST CONTAIN TITLE                           MAN243C
C        COL 13 TO 78    NEW TITLE DESCRIPTOR                         MAN244C
C                                                                     MAN245C
C      2.3) EXECUTION TERMINATION (6X,A4)                             MAN246C
C                                                                     MAN247C
C        COL  7 TO 10    MUST CONTAIN STOP. INSERT AFTER LAST PROBLEM. MAN248C
C                                                                     MAN249C
C                                                                     MAN250C
C     3.) MATERIAL CHARACTERIZATION (I5,1X,12A6)                      MAN251C
C                                                                     MAN252C
C        COL  1 TO  5    NUMMAT - NUMBER OF DIFFERENT MATERIAL CHARACT- MAN253C
C                        ERIZATIONS TO FOLLOW.                        MAN254C
C        COL  7 TO 12    MUST CONTAIN WORD MATERI                     MAN255C
C                                                                     MAN256C
C        THE FOLLOWING CARDS ARE SUPPLIED FOR EACH MATERIAL TO BE CHARAC MAN257C
C        TERIZED (MUST BE EXACTLY NUMMAT SETS OF CARDS)               MAN258C
C                                                                     MAN259C
C        CARD 1.) ELEMENT SELECTOR CARD (I5,1X,A5,11A6)               MAN260C
C                                                                     MAN261C
C        COL  1 TO  5    MATERIAL NUMBER (1 TO NUMMAT)                MAN262C
C        COL  7 TO 11    ELMNN - WHERE NN IS NUMBER OF ELEMENT CLASS (01 MAN263C
C                        TO 10) TO WHICH THE CHARACTERIZATION BELONGS. MAN264C
C        COL 12 TO 77    ALPHANUMERIC INFORMATION TO BE OUTPUT.       MAN265C
```

```
C                                                                               MAN266C
C       CARD 2.), ETC. ** USER DEFINED FOR EACH ELEMENT TYPE PROVIDED.          MAN267C
C                                                                               MAN267C
C       EXCESS BLANK CARDS ARE PERMISSIBLE BETWEEN EACH MATERIAL SET.           MAN267C
C                                                                               MAN268C
C     4.) NODAL CARDS (I5,1X,A6)                                                MAN269C
C                                                                               MAN270C
C       COL  1 TO  5    NUMNP - NUMBER OF NODAL POINTS                          MAN271C
C       COL  7 TO 12    MUST CONTAIN NODAL                                      MAN272C
C                                                                               MAN273C
C       SUBSEQUENT CARDS  LAST NODAL CARD MUST NOT BE GENERATED.                MAN274C
C       (I5,I15,3F10.0)                                                         MAN275C
C                                                                               MAN276C
C       COL  1 TO  5    NODE NUMBER                                             MAN277C
C       COL 15          1 IF 1 DISPLACEMENT IS SPECIFIED                        MAN278C
C       COL 16          1 IF 2 DISPLACEMENT IS SPECIFIED                        MAN279C
C       COL 17          1 IF 3 DISPLACEMENT IS SPECIFIED                        MAN280C
C       COL 18          1 IF 4 DISPLACEMENT IS SPECIFIED                        MAN281C
C       COL 19          1 IF 5 DISPLACEMENT IS SPECIFIED                        MAN282C
C       COL 20          1 IF 6 DISPLACEMENT IS SPECIFIED                        MAN283C
C       COL 21 TO 30    1 COORDINATE VALUE                                      MAN284C
C       COL 31 TO 40    2 COORDINATE VALUE * AS REQUIRED                        MAN285C
C       COL 41 TO 50    3 COORDINATE VALUE                                      MAN286C
C                                                                               MAN287C
C     NODAL CARDS MUST BE IN ORDER. MISSING NODES ARE INTERPOLATED              MAN288C
C     LINEARLY FROM INPUT NODES. IF SUCCEEDING CARDS HAVE IDENTICAL             MAN289C
C     BOUNDARY CODES, THIS BOUNDARY CODE WILL BE ASSIGNED TO THE INTER-         MAN290C
C     VENING NODES. IN ALL OTHER CASES THE BOUNDARY CODE IS SET TO ZERO         MAN291C
C     *TERMINATE ON NODE NUMNP OR A BLANK CARD*                                 MAN292C
C                                                                               MAN293C
C     4.1) NON SEQUENTIAL NODAL GENERATOR OPTION. (I5,1X,A6)                    MAN294C
C                                                                               MAN295C
C       COL  1 TO  5    NUMBER OF NODAL POINTS                                  MAN296C
C       COL  7 TO 12    MUST CONTAIN GENERA                                     MAN297C
C                                                                               MAN298C
C       SUBSEQUENT CARDS (2I5,I10,3F10.0)                                       MAN299C
C                                                                               MAN300C
C       COL  1 TO  5    NODE-NUMBER                                             MAN301C
C       COL  6 TO 10    NODE-NUMBER-INCREMENT WHICH WILL BE SUCESSIVELY         MAN302C
C                       ADDED TO NODE-NUMBER UNTIL SUM IS GREATER THAN          MAN303C
C                       NODE-NUMBER ON FOLLOWING CARD (ALGEBRAIC).              MAN304C
C       COL 15 TO 20    BOUNDARY CODE. SAME AS INPUT FOR NODAL.                 MAN305C
C                       IF SUCCEEDING CARDS HAVE IDENTICAL BOUNDARY             MAN306C
C                       CODES, THIS BOUNDARY CODE WILL BE ASSIGNED TO           MAN307C
C                       THE INTERVENING NODES. IN ALL OTHER CASES THE           MAN308C
C                       BOUNDARY CODE IS SET TO ZERO.                           MAN309C
C       COL 21 TO 30    1 COORDINATE VALUE  *                                   MAN310C
C       COL 31 TO 40    2 COORDINATE VALUE  * AS REQUIRED *                     MAN311C
C       COL 41 TO 50    3 COORDINATE VALUE  *                                   MAN312C
C                                                                               MAN313C
C       *TERMINATE WITH BLANK CARD *                                            MAN314C
C                                                                               MAN315C
C     4.2) BOUNDARY CODE PATCH-UP OPTION. (6X,A6)                               MAN316C
C                                                                               MAN317C
```

```
C        COL  7 TO 12    MUST CONTAIN BOUNDA                                     MAN318C
C                                                                                MAN319C
C        SUBSEQUENT CARDS. (8I5)                                                 MAN320C
C                                                                                MAN321C
C        COL  1 TO  5    N, NODE NUMBER TO HAVE REDEFINED BOUNDARY CODE.         MAN322C
C        COL  6 TO 10    NX, GENERATOR INCREMENT TO BE ADDED ALGEBRAI-           MAN323C
C                        CALLY TO N, UNTIL SUM EXCEEDS (MAX OR MIN) THE          MAN324C
C                        N OF THE FOLLOWING CARD.                                MAN325C
C        COL 11 TO 15    IBC(I), (I=1,2...NDF) CODE FOR SPECIFYING FORCE         MAN326C
C        COL 16 TO 20    OR DISPLACEMENT BOUNDARY CONDITIONS,                    MAN327C
C        COL  ...                                                                MAN328C
C                             IBC(I) .EQ. 0. FORCE SPECIFIED.                    MAN329C
C                             IBC(I) .GT. 0. DISPLACEMENT SPECIFIED, NO          MAN330C
C                             INTERVENING GENERATION.                            MAN331C
C                             IBC(I) .LT. 0. DISPLACEMENT SPECIFIED,             MAN332C
C                             GENERATE BETWEEN MISSING NODES IN ALGEBRAIC        MAN333C
C                             INCREMENTS OF NX.                                  MAN334C
C                                                                                MAN335C
C        * TERMINATE WITH A BLANK CARD. *                                        MAN336C
C                                                                                MAN337C
C      4.3) POLAR OR CYLINDRICAL COORDINATE CONVERSION TO CARTESIAN              MAN338C
C           COORDINATES (6X,A6)                                                  MAN339C
C                                                                                MAN340C
C        COL  7 TO 12    MUST CONTAIN POLAR     (LEFT JUSTIFIED)                 MAN341C
C                                                                                MAN342C
C        CARD 1.  (3I5,5X,2F10.0)                                                MAN343C
C                                                                                MAN344C
C        COL  1 TO  5    N1, FIRST NODE TO BE CONVERTED                          MAN345C
C        COL  6 TO 10    N2, LAST  NODE TO BE CONVERTED                          MAN346C
C        COL 11 TO 15    N3, INCREMENT ADDED (ALGEBRAICALLY), N1 TO N2           MAN347C
C        COL 21 TO 30    X0, ORIGIN OF POLAR X-COORDINATE                        MAN348C
C        COL 31 TO 40    Y0, ORIGIN OF POLAR Y-COORDINATE                        MAN349C
C                                                                                MAN350C
C     5.) ELEMENT CARDS (I5,1X,A6)                                               MAN351C
C                                                                                MAN352C
C        COL  1 TO  5    NUMEL - NUMBER OF ELEMENTS                              MAN353C
C        COL  7 TO 12    MUST CONTAIN ELEMEN                                     MAN354C
C                                                                                MAN355C
C        SUBSEQUENT CARDS (4I5,20I3/20I4)                                        MAN356C
C                                                                                MAN357C
C        CARD 1.                                                                 MAN358C
C                                                                                MAN359C
C        COL  1 TO  5    ELEMENT NUMBER                                          MAN360C
C        COL  6 TO 10    MATERIAL NUMBER                                         MAN361C
C        COL 11 TO 15    NUMBER OF SUBSEQUENT ELEMENTS USING SAME                MAN362C
C                        STIFFNESS MATRIX * SAVES RECOMPUTATION OF               MAN363C
C                        SIMILAR MATRICES.  ELEMENT MUST ALSO HAVE               MAN364C
C                        SAME ELEMENT FORCE VECTOR * IF THESE ARE                MAN365C
C                        IN THE STIFFNESS SUBROUTINE *                           MAN366C
C        COL 16 TO 20    PRINT ELEMENT MATRIX IF NONZERO.                        MAN367C
C        COL 21 TO 23    IXD(1) ELEMENT INCREMENT ARRAY ON NODE 1.               MAN368C
C        COL 24 TO 26    IXD(2)  * IF NOT INPUT IS SET AUTOMATICALLY             MAN369C
C                        UP TO                                                   MAN370C
C        COL 79 TO 80    IXD(20)    FOR SERENDIPITY ELEMENTS * SEE REPORT        MAN371C
```

```
C                                                                      MAN372C
C       CARD 2.                                                        MAN373C
C                                                                      MAN374C
C       COL  1 TO  4    NODE 1                                         MAN375C
C       COL  5 TO  8    NODE 2                                         MAN376C
C       COL  9 TO 12    NODE 3          IX(NEL1,1) ARRAY               MAN377C
C                           CONTINUE IN I4 FORMAT TO A MAXIMUM         MAN378C
C       COL 77 TO 80    NODE 20                                        MAN379C
C                                                                      MAN380C
C       ELEMENT CARDS MUST BE IN ORDER. MISSING ELEMENTS ARE GENERATED MAN382C
C       BY INCREMENTING NODES.                                         MAN383C
C       LAST ELEMENT CARD MUST NOT BE GENERATED.                       MAN384C
C       * TERMINATE ON ELEMENT NUMEL OR A BLANK CARD *                 MAN385C
C                                                                      MAN386C
C     5.1) BLOCK GENERATOR, GENERATES ALL MESH DATA.(6X,A6)            MAN387C
C                                                                      MAN388C
C       COL  1 TO  5    NUMBER OF NODAL POINTS TO BE GENERATED.        MAN389C
C       COL  7 TO 12    MUST CONTAIN BLOCK                             MAN390C
C                                                                      MAN391C
C       SUBSEQUENT CARDS (10I5/6(4X,I6)/(10X,3F10.0))                  MAN392C
C                                                                      MAN393C
C       CARD 1.                                                        MAN394C
C                                                                      MAN395C
C       COL  1 TO  5    NN, NUMBER OF POINTS REQUIRED TO DEFINE        MAN396C
C                       BOUNDARY OF REGION. FOR 2-DIM., NN=4 OR 8.     MAN397C
C                       FOR 3-DIM., NN= 8 OR 20.                       MAN398C
C       COL  6 TO 10    NUMBER OF ELEMENTS IN R-DIRECTION.             MAN399C
C       COL 11 TO 15    NUMBER OF ELEMENTS IN S-DIRECTION.             MAN400C
C       COL 16 TO 20    NUMBER OF ELEMENTS IN T-DIRECTION.             MAN401C
C       COL 21 TO 25    INITIAL NODE NUMBER, DEFAULT = 1.              MAN402C
C       COL 26 TO 30    INITIAL ELEMENT NUMBER, DEFAULT = 1.           MAN403C
C       COL 31 TO 35    MATERIAL NUMBER OVER REGION, DEFAULT = 1       MAN404C
C       COL 36 TO 40    BOUNDARY CODE SKIP. A NON-ZERO ENTRY WILL OMIT MAN405C
C                       SETTING ALL INTERIOR BOUNDARY CODES TO ZERO.   MAN406C
C       COL 41 TO 45    IREUSE - REUSE ELEMENT STIFFNESS OPTION - USES MAN407C
C                       EACH ELEMENT STIFFNESS IREUSE TIMES BEFORE     MAN408C
C                       GENERATING A NEW ELEMENT STIFFNESS MATRIX.     MAN409C
C       COL 46 TO 50    ELEMENT STIFFNESS-PRINT. A NON-ZERO ENTRY  WILL MAN410C
C                       CAUSE PRINT-OUT OF FIRST ELEMENT.              MAN411C
C       COL 51 TO 55    IMSH - IF NONZERO PRINT COMPUTED NODES.        MAN412C
C       COL 56 TO 60    IELM - IF NONZERO PRINT COMPUTED ELEMENTS.     MAN413C
C                                                                      MAN414C
C       CARD 2.  (BOUNDARY CODE AS DEFINED IN NODAL CARD.)             MAN415C
C                                                                      MAN416C
C       COL  5 TO 10    BOUNDARY CODE OVER FACE -R.                    MAN417C
C       COL 15 TO 20    BOUNDARY CODE OVER FACE +R.                    MAN418C
C       COL 25 TO 30    BOUNDARY CODE OVER FACE -S.                    MAN419C
C       COL 35 TO 40    BOUNDARY CODE OVER FACE +S.                    MAN420C
C       COL 45 TO 50    BOUNDARY CODE OVER FACE -T.                    MAN421C
C       COL 55 TO 60    BOUNDARY CODE OVER FACE +T.                    MAN422C
C                                                                      MAN423C
C       CARD 3.  (REPEAT NN TIMES.)                                    MAN424C
C                                                                      MAN425C
C       COL 11 TO 20    1-COORDINATE OF BOUNDARY-DEFINING-POINT.       MAN426C
```

```
C        COL 21 TO 30    2-COORDINATE OF BOUNDARY-DEFINING-POINT.
C        COL 31 TO 40    3-COORDINATE OF BOUNDARY-DEFINING-POINT.          MAN427C
C                                                                          MAN428C
C        NOTES, 1. BLOCK GENERATES ONLY 4 PT. QUADRILATERALS OR 8 PT.      MAN429C
C                  BRICKS.                                                 MAN430C
C               2. INPUT OF CARDS 3.) FOLLOW ORDER RULES FOR ELEMENT       MAN431C
C                  INPUT (SEE SECTION 5.)).                                MAN432C
C               3. R-S-T ARE LOCAL COORDINATES,                            MAN433C
C                  I.E. (-1 .LE. R,S,T .LE. 1), WHERE R IS DIRECTED        MAN433C
C                  FROM NODE 1 TO 2, S IS IN PLANE OF FIRST THREE NODES    MAN433C
C                  AND T IS NORMAL TO R-S PLANE.                           MAN433C
C
C     6.)  VECTOR CARDS, I.E. USER DEFINED INPUT (I5,1X,A6)                MAN435C
C                                                                          MAN436C
C        COL  1 TO  5    NVEC, NUMBER OF DIFFERENT VECTORS (7 MAX)         MAN437C
C        COL  7 TO 12    MUST CONTAIN VECTOR                               MAN438C
C                                                                          MAN439C
C        SUBSEQUENT CARDS                                                  MAN440C
C                                                                          MAN441C
C        CARD 1.  (2I5)                                                    MAN442C
C                                                                          MAN443C
C        COL  1 TO  5    NSIZV, VECTOR LENGTH,COMMON TO ALL NVEC VECTORS   MAN444C
C        COL  6 TO 10    IPICK, CODED PARAMETER,                           MAN445C
C                                                                          MAN446C
C                        IPICK = 0, VECTORS ASSOCIATED WITH NODES          MAN447C
C                        IPICK = 1, VECTORS ASSOCIATED WITH DEG.FREEDOM    MAN448C
C                        IPICK = 2, VECTORS ASSOCIATED WITH ELEMENTS       MAN449C
C                                                                          MAN450C
C        CARD 2.  (6X, 2A6) REPEAT NVEC TIMES                              MAN451C
C                                                                          MAN452C
C        COL  7 TO 18     DESCRIPTIVE TITLE FOR VECTOR                     MAN453C
C                                                                          MAN454C
C        CARD 3.  (2I5,7F10.0)                                             MAN455C
C                                                                          MAN456C
C        COL  1 TO  5    POSITION NUMBER OF VECTOR ELEMENT, 1 TO NSIZV     MAN457C
C        COL  6 TO 10    GENERATOR INCREMENT                               MAN458C
C        COL 11 TO 20    VECTOR ELEMENT VALUE OF VECTOR 1                  MAN459C
C        COL 21 TO 30    VECTOR ELEMENT VALUE OF VECTOR 2                  MAN460C
C        COL ........    AS REQUIRED FOR NVEC VECTORS                      MAN461C
C                                                                          MAN462C
C        LINEAR INTERPOLATION IS PERFORMED ON ALL VECTORS BETWEEN          MAN463C
C        NON-CONSECUTIVE POSITION NUMBERS SPECIFIED IN COL 1 TO 5          MAN464C
C        IF INCREMENT IS NONZERO.                                          MAN465C
C        IF DESCRIPTIVE TITLES OF ALL VECTORS ARE BLANK CARDS,             MAN466C
C        PRINTING OF THE VECTOR VALUES IS SUPRESSED.                       MAN467C
C                                                                          MAN468C
C        * TERMINATE ON BLANK CARD *                                       MAN469C
C                                                                          MAN470C
C     6.1)  INITIAL CONDITIONS FOR TIME DEPENDENT ANALYSIS.                MAN471C
C                                                                          MAN472C
C        COL  1 TO  5    NICD, NUMBER OF INITIAL CONDITION VECTORS         MAN473C
C        COL  7 TO 12    MUST CONTAIN INITIA                               MAN474C
C                                                                          MAN475C
C        SUBSEQUENT CARDS                                                  MAN476C
C                                                                          MAN477C
```

```
C
C        CARD 1. (6X,2A6) REPEAT NICD TIMES
C
C        COL  7 TO 18     DESCRIPTIVE TITLE FOR INITIAL CONDITIONS
C
C        CARD 2.  (2I5,7F10.0)
C
C        COL  1 TO  5     POSITION NUMBER, AS IN VECTOR CARDS FOR IPICK=1
C        COL  6 TO 10     GENERATOR INCREMENT
C        COL 11 TO 20     INITIAL CONDITION 1
C        COL 21 TO 30     INITIAL CONDITION 2
C        COL ........     AS REQUIRED FOR NICD INITIAL CONDITIONS
C
C        INTERPOLATION BETWEEN INPUT VALUES AS DESCRIBED IN VECTOR INPUT
C        **** NOTE **** IF MISSING THE INITIAL CONDITIONS ARE SET ZERO
C
C     7.) FORCE CARDS (I5,1X,A6)
C
C        COL  1 TO  5     LAST NODE TO WHICH A FORCE IS TO BE SPECIFIED
C        COL  7 TO 12     MUST CONTAIN FORCE
C
C        SUBSEQUENT CARDS (I5,5X,6F10.0)
C
C        THE FOLLOWING VALUES ARE EACH INTERPRETTED AS FORCES IF THE
C        CORRESPONDING BOUNDARY CODE IS A 0 *ZERO* AND AS A DISPLACEMENT
C        IF THE CORRESPONDING BOUNDARY CODE IS 1 *ONE*.
C
C        COL  1 TO  5     NODE TO WHICH FORCE OR DISPLACEMENT IS APPLIED
C        COL 11 TO 20     VALUE OF 1 FORCE/DISPLACEMENT
C        COL 21 TO 30     VALUE OF 2 FORCE/DISPLACEMENT    * AS *
C        COL 31 TO 40     VALUE OF 3 FORCE/DISPLACEMENT * REQUIRED *
C        COL 41 TO 50     VALUE OF 4 FORCE/DISPLACEMENT
C        COL 51 TO 60     VALUE OF 5 FORCE/DISPLACEMENT
C        COL 61 TO 70     VALUE OF 6 FORCE/DISPLACEMENT
C
C
C     7.1) SURFACE LOAD CARDS (I5,1X,A6)
C
C        COL  1 TO  5     NUMBER OF LOADED FACE CARDS
C        COL  7 TO 12     MUST CONTAIN BLOADS
C
C        CARD 1. (I5,1X,A5,I4,8I5,8I3)
C
C        COL  1 TO  5     DIMENSION OF LOADING SURFACE, (1 OR 2).
C        COL  7 TO 11     SLD(NN), ALPHA-NUMERIC NAME OF SURFACE LOADING
C                         SUBROUTINE (NN IS BETWEEN 1 AND 5)
C        COL 12 TO 15     NRT,NUMCER OF ADDITIONAL ELEMENT LOAD
C                         SURFACES TO BE GENERATED FROM CURRENT MODEL.
C        COL 16 TO 20     IPRES(N), NODE NUMBERS DEFINING LOADING SURFACE
C        COL 21 TO 25     OF CURRENT ELEMENT.
C        COL ...          (IDENTIFY FROM 2 TO 8 AS REQUIRED)
C        COL 51 TO 55
C        COL 56 TO 58     INC(N), INCREMENT VALUE ADDED TO IPRES(N) TO
C        COL 59 TO 61     IDENTIFY NODE NUMBERS OF A GENERATED SEQUENCE.
```

```
C        COL ...          (IDENTIFY FROM 2 TO 8 AS REQUIRED)                        MAN532C
C                                                                                   MAN533C
C        CARD 2. (8F10.0)                                                           MAN534C
C                                                                                   MAN535C
C        COL  1 TO 80     LOAD AT NODES GIVEN ON PREVIOUS CARD *                    MAN536C
C                         MUST CORRESPOND IN SEQUENCE TO THE NODE NUMBERS           MAN537C
C                                                                                   MAN538C
C                                                                                   MAN539C
C                                                                                   MAN540C
C      7.2) ELEMENT LOAD CARDS (I5,1X,A6 )                                          MAN541C
C                                                                                   MAN542C
C        COL  1 TO  5     NLD. NUMBER OF ELEMFNT LOAD CARDS.                        MAN543C
C        COL  7 TO 11     MUST CONTAIN ELOADS                                       MAN544C
C                                                                                   MAN545C
C        SUBSEQUENT CARDS (I5,1X,A5,I4,I5,6F10.0)                                   MAN546C
C                                                                                   MAN547C
C        COL  1 TO  5     IEL, INITIAL ELEMENT OF A GENERATED SEQUENCE.             MAN548C
C        COL  7 TO 11     ELM(NN), ALPHA-NUMERIC NAME OF ELEMENT                    MAN549C
C                         SUBROUTINE WHERE ELEMENT LOADS ARE COMPUTED.              MAN550C
C                         USED AS CHECK TO INSURE IEL, ETC. ARE PROPER              MAN551C
C                         ELEMENTS.                                                 MAN552C
C        COL 12 TO 15     INC, INCREMENT NUMBER IN A GENERATED SEQUENCE,            MAN553C
C                         (DEFAULT = 1).                                            MAN554C
C        COL 16 TO 20     JEL, TERMINAL ELEMENT NUMBER IN A GENERATED               MAN555C
C                         SEQUENCE, IF JEL = 0, ONLY IEL IS COUNTED.                MAN556C
C        COL 21 TO 80     PR-USER DEFINED VALUES FOR DETERMINING BODY               MAN557C
C                         LOADS IN THE ISW=5 PORTION OF ELM(NN).                    MAN558C
C                                                                                   MAN559C
C        NOTE, USER MUST PROVIDE COMPUTATION OF LOADS IN ELMTNN.                    MAN560C
C        PR IS TRANSFERED TO SUBROUTINE ELMTNN IN THE U VECTOR,                     MAN561C
C        WHEN ISW =5, ONLY.                                                         MAN562C
C                                                                                   MAN563C
C      7.3) PROPORTIONAL LOADS FOR TIME DEPENDENT ANALYSIS                          MAN564C
C                                                                                   MAN565C
C        TRANSFER TO THIS OPTION OCCURS ONLY FOR TIME ANALYSES.                     MAN566C
C                                                                                   MAN567C
C        ONE CARD FOR EACH PROPORTIONAL LOAD REQUIRED                               MAN568C
C                                                                                   MAN569C
C        COL  1 TO  5     PROPORTIONAL LOAD TYPE, 1,2 OR 3                          MAN570C
C        COL  6 TO 10     K, TABLE CONSTANT                                         MAN571C
C        COL 11 TO 20     TMIN, SMALLEST TIME LOADING IS VALID                      MAN572C
C        COL 21 TO 30     TMAX, LARGEST TIME LOADING IS VALID                       MAN573C
C        COL 31 TO 40     A0                                                        MAN574C
C        COL 41 TO 50     A1                                                        MAN575C
C        COL 51 TO 60     A2                                                        MAN576C
C        COL 61 TO 70     A3                                                        MAN577C
C        COL 71 TO 80     A4                                                        MAN578C
C                                                                                   MAN579C
C        LOAD TYPE 1.  T = TIME                                                     MAN580C
C                                                                                   MAN581C
C            PROP = A0 + A1*T + A2*T*T + A3*T*T*T + A4*T*T*T*T                      MAN582C
C                                                                                   MAN583C
C        LOAD TYPE 2.                                                               MAN584C
C                                                                                   MAN585C
```

```
C           PROP = A0*(SIN(A1*T))**K + A2*(COS(A3*T))**K + A5           MAN586C
C                                                                       MAN587C
C      LOAD TYPE 3.                                                     MAN588C
C                                                                       MAN589C
C          PROP = USER DEFINED FUNCTION FROM SUBROUTINE                 MAN590C
C                 EXPRLD(PROP,T,A)  WHERE A IS AN ARRAY WITH            MAN591C
C                 INFORMATION FOR COLUMNS 6-80 OF DATA CARDS.           MAN592C
C                                                                       MAN593C
C      **NOTE** PROPORTIONAL LOADS CAN BE ACCUMULATED FROM DIFFERENT    MAN594C
C               TYPES AT THE SAME TIME.                                 MAN595C
C                                                                       MAN596C
C    8.) INITIATION OF TIME INDEPENDENT SOLUTION (I5,1X,A6)             MAN597C
C                                                                       MAN598C
C      COL  1 TO  5     IOUT, OUTPUT CONTROL CODE,                      MAN599C
C                                                                       MAN600C
C                       IOUT .EQ. 0, ALL STRESSES AND DISP. PRINTED     MAN601C
C                       IOUT .NE. 0, SELECTED PRINTOUT, MORE DATA INPUT MAN602C
C                       SEE SECTION 9 FOR DATA PREPARATION.             MAN603C
C                                                                       MAN604C
C      COL  7 TO 12     MUST CONTAIN  SOLVE  *INDICATES ALL DATA INPUT* MAN605C
C                       COMPLETE FORMULATION AND SOLUTION OF EQUATIONS. MAN606C
C      COL  7 TO 12     MUST CONTAIN  RESOLV TO OBTAIN SUBSEQUENT       MAN607C
C                       SOLUTIONS WHERE BOUNDARY CODES DO NOT CHANGE    MAN608C
C                       AND ALL PRESCRIBED DISPLACEMENTS ARE ZERO.      MAN609C
C                                                                       MAN610C
C    8.1) INITIATION OF DYNAMIC SOLUTION BY EXPLICIT INTEGRATION.       MAN611C
C                                                                       MAN612C
C      COL  1 TO  5     IPRT,  OUTPUT CONTROL FOR DISPLACEMENT AND      MAN613C
C                       STRESS PRINTOUT. SEE SECT. 9 FOR DATA INPUT.    MAN614C
C                       IOUT = 1 - MIN(1,IPRT)                          MAN614C
C                       IF IOUT .NE. 0, THE SPATIAL CONTROL DATA        MAN615C
C                       COMES AT THE END OF THE DYNAMIC SEGMENT.        MAN616C
C      COL  7 TO 12     MUST CONTAIN EXPLIC                             MAN617C
C                                                                       MAN618C
C      SUBSEQUENT CARDS (2I5,2F10.0,2I5)                                MAN619C
C                                                                       MAN620C
C      COL  1 TO  5     NUMBER OF TIME STEPS                            MAN621C
C      COL  6 TO 10     PRINT INTERVAL                                  MAN622C
C      COL 11 TO 20     TIME INCREMENT                                  MAN623C
C      COL 21 TO 30     NEWMARK DELTA-DAMPING TERM (GAMMA - .5)         MAN624C
C      COL 31 TO 35     NUMBER OF TIME EVOLUTION ELEMENT VARIABLE PLOTS MAN625C
C      COL 36 TO 40     NPROP, NUMBER OF PROPORTIONAL LOADS TO BE INPUT MAN626C
C      COL 41 TO 45     NFORC, LAST NODE ON WHICH A FORCE IS CHANGED    MAN627C
C                       DURING EACH TIME STEP.                          MAN628C
C      COL 46 TO 50     KKK, STABILITY CHECK OVERRIDE ** CAUTION USE    MAN629C
C                       ONLY WHEN A BETTER ESTIMATE OF THE STABLE TIME  MAN630C
C                       STEP IS AVAILABLE THAN CAN BE PERFORMED BY CODE MAN631C
C                       KKK ZERO, USES INTERNAL STABILITY CHECK.        MAN632C
C                       KKK NONZERO, DISREGARDS STABILITY CHECK.        MAN633C
C      COL 51 TO 55     NONL,  0 FOR LINEAR                             MAN633
C                              1 FOR NON LINEAR                         MAN633C
C      COL 56 TO 60     LPR,   0 PRINTS LOADS                           MAN633C
C                              1 SUPPRESS PRINT                         MAN633C
C                                                                       MAN634C
```

```
C        SUBSEQUENT CARDS (3I5) ONE FOR EACH STRESS PLOT.                MAN635C
C                                                                        MAN636C
C        COL  1 TO  5   ELEMENT NUMBER CONTAINING STRESS TO BE PLOTTED.  MAN637C
C        COL  6 TO 10   LOCAL COORDINATE POINT CODE, 1 TO 9, AS          MAN638C
C                       PATTERNED AFTER, COL 11 TO COL 19, IN SECT. 9.   MAN639C
C        COL 11 TO 15   PLOT COMPONENT CODE, 1 TO 6 FOR SIGMA(I,J),      MAN640C
C                       I.E., SIGMA(1,1)=1, SIGMA(1,2)=2, SIGMA(1,3)=3,  MAN641C
C                       SIGMA(2,2) = 4, SIGMA(2,3) = 5, SIGMA(3,3) = 6.  MAN642C
C                                                                        MAN643C
C        IF(NPROP.NE.0) READ PROPORTIONAL LOAD CARDS, SEE SECT. 7.3      MAN644C
C                                                                        MAN645C
C        IF(NFORC.NE.0) READ FORCE CARDS AT EACH TIME STEP. IF OUTPUT IS MAN646C
C        LIMITED BY IOUT NONZERO, THE FIRST FORCE CARD SET PRECEDES      MAN647C
C        OUTPUT CARDS AND THE REMAINDER FOLLOW THE OUTPUT CARDS NO BLANK MAN648C
C        CARDS MAY BE USED BETWEEN SETS OF CARDS OTHER THAN THE USUAL    MAN649C
C        BLANK TERMINATOR CARD FOR FORCE INPUT CARDS.                    MAN650C
C                                                                        MAN651C
C        IF(IOUT.NE.0)   DATA FOR SPATIAL PRINTOUT CONTROL, SEE SECT.9.  MAN652C
C                                                                        MAN653C
C                                                                        MAN654C
C       SPECIAL COMMENTS FOR DYNAMIC OPTION                              MAN655C
C                                                                        MAN656C
C       (1) ONLY COLUMNS 1 TO 66 ARE AVAILABLE FOR PAGE HEADING.         MAN657C
C       (2) MAXIMUM ADVANTAGE OF ELEMENT REUSE OPTION SHOULD BE TAKEN.   MAN658C
C       (3) INITIAL CONDITIONS FOR DISPLACEMENT AND VELOCITY VECTORS,    MAN659C
C           AS WELL AS STORAGE FOR ACCELERATION VECTOR, MAY BE MADE      MAN660C
C           THROUGH INPUT OF AN INITIAL CONDITION CARD SET. WITHOUT      MAN661C
C           SPECIFIED INITIAL CONDITIONS THEY ARE AUTOMATICALLY SET ZERO.MAN662C
C       (4) SPATIAL LOADING IS INPUT THROUGH FORCE OR BOUNDARY           MAN663C
C           PRESSURE CARDS.                                              MAN664C
C       (5) EXTREME CAUTION ON ORDER OF DATA CARDS MUST BE OBSERVED.  NO MAN665C
C           EXTRA CARDS ARE PERMITTED AND STRICT COUNTS ARE OBSERVED     MAN666C
C           EXCEPT FOR THE NUMBER OF FORCE CARDS USED IN EACH TIME STEP. MAN667C
C                                                                        MAN668C
C       8.2) INITIATION OF IMPLICIT TIME INTEGRATIONS (I5,1X,A6)         MAN669C
C                                                                        MAN670C
C         COL  1 TO  5   NSEQ, NUMBER OF TIME SEQUENCES                  MAN671C
C         COL  7 TO 12   MUST CONTAIN VISCOE FOR LINEAR VISCOELASTIC     MAN672C
C                        QUASI-STATIC PROBLEMS (ONE INITIAL CONDITION    MAN673C
C                        ONLY MUST BE USED)                              MAN674C
C         COL  7 TO 12   MUST CONTAIN IMPLIC FOR DYNAMIC IMPLICIT        MAN675C
C                        INTEGRATIONS (THREE INITIAL CONDITIONS ARE      MAN676C
C                        REQUIRED, MORE CAN BE SPECIFIED WITHOUT ERROR)  MAN677C
C                                                                        MAN688C
C         SUBSEQUENT CARDS, ONE SET FOR EACH TIME SEQUENCE               MAN689C
C                                                                        MAN690C
C         CARD 1. (F10.0,8I5,2F10.0,2I5)                                 MAN691C
C                                                                        MAN692C
C         COL  1 TO 10   DT, TIME INCREMENT (NONZERO FOR IMPLIC)         MAN693C
C         COL 11 TO 15   NTS, NUMBER OF TIME STEPS IN SEQUENCE           MAN694C
C         COL 16 TO 20   INT, PRINT INTERVAL (DEFAULT 1)                 MAN695C
C         COL 21 TO 25   NNI, FIRST NODE PRINTED                         MAN696C
C         COL 26 TO 30   NNE, LAST  NODE PRINTED                         MAN697C
C         COL 31 TO 35   NEI, FIRST ELEMENT STRESS TO BE PRINTED         MAN698C
```

```
C       COL 36 TO 40    NEF, LAST  ELEMENT STRESS TO BE PRINTED           MAN699C
C       COL 41 TO 45    NPROP, NUMBER OF PROPORTIONAL LOADS IN SEQUENCE   MAN700C
C       COL 46 TO 50    NFORC, LAST NODE FOR GENERALIZED FORCES TO BE     MAN701C
C                       INPUT FOR EACH TIME IN SEQUENCE (SEE SECT.7.      MAN702C
C                       FOR DATA PREPARATION FORMATS)                     MAN703C
C       COL 51 TO 60    BETA, NEWMARK INTEGRATION PARAMETER (IMPLIC)      MAN704C
C                       DEFAULT VALUE IS 0.25                             MAN705C
C       COL 61 TO 70    DEL = GAMMA - 0.5, NEWMARK INTEGRATION            MAN706C
C                       PARAMETER (IMPLIC)                                MAN707C
C       COL 71 TO 75    NUMPLT, NUMBER OF STRESS PLOTS REQUESTED          MAN708C
C                       (INPUT ON FIRST TIME SEQUENCE ONLY)               MAN709C
C       COL 76 TO 80    NITS, NUMBER OF ITERATIONS PER TIME STEP          MAN710C
C                       (TO ACHIEVE EQUILIBRIUM IN NON LINEAR PROBLEMS)   MAN711C
C                       ZERO OR BLANK FOR LINEAR PROBLEMS.                MAN712C
C                       POSITIVE - REFORM THE NEWTON JACOBIAN MATRIX      MAN713C
C                                  AT START OF EACH TIME STEP ONLY.       MAN714C
C                       NEGATIVE - REFORM THE NEWTON JACOBIAN MATRIX      MAN715C
C                                  AT EACH ITERATION IN EACH TIME STEP.   MAN716C
C                       IF CONVERGENCE OCCURS BEFORE ABS(NITS)            MAN717C
C                       ITERATIONS, THE PROGRAM WILL PROCEED TO THE       MAN718C
C                       NEXT TIME STEP.                                   MAN719C
C                       NOTE * * BECAUSE OF THE CONVERGENCE TEST USED,    MAN720C
C                                A MINIMUM OF TWO ITERATIONS MUST BE      MAN721C
C                                PERFORMED BEFORE THE CHECK CAN BE        MAN722C
C                                IMPOSED.  THUS, USE OF NITS CAN BE       MAN723C
C                                INEFFICIENT IF LINEAR PROBLEMS ARE       MAN724C
C                                BEING SOLVED.                            MAN725C
C                                                                         MAN726C
C       CARD 2. (2I5,7F10.0)                                              MAN727C
C                                                                         MAN728C
C       ONE FOR EACH NPROP. SEE SECT.7.3 FOR DATA PREPARATION             MAN729C
C                                                                         MAN730C
C                                                                         MAN731C
C       SUBSEQUENT CARDS FOR EACH TIME STEP IN THE SEQUENCE               MAN732C
C                                                                         MAN733C
C       FORCE CARDS.  SEE SECTION 7. FOR DATA PREPARATION FORMATS.        MAN734C
C                                                                         MAN735C
C     8.3) MESH CHECKS AND PLOTS                                          MAN736C
C                                                                         MAN736C
C       COL  7 TO 10    MUST CONTAIN MESH.                                MAN736C
C                       PROGRAM WILL CHECK THE PREVIOUSLY INPUTTED MESH   MAN736C
C                       FOR ERRORS, BUT DOES NOT PROCEED TO SOLVE.        MAN736C
C       COL  7 TO 10    MUST CONTAIN PLOT.                                MAN736C
C                       PERFORMS SAME CHECKS AS FOR MESH AND PLOTS THE    MAN736C
C                       MESH. (GDS OR NOVA PLOT ROUTINES AVAILABLE).      MAN736C
C                                                                         MAN736C
C     8.4) INPUT OF FOURIER SERIES HARMONICS                              MAN736C
C                                                                         MAN736C
C       COL  1 TO  5    IOUT. (SEE 8.)                                    MAN736C
C       COL  7 TO 12    FOURIE                                            MAN736C
C                                                                         MAN736C
C       CARD 1.  (I5,5X,6F10.0)                                           MAN736C
C                                                                         MAN736C
C       COL  1 TO 10    HARMONIC NUMBER                                   MAN736C
```

```
C        COL 11 TO 20     1 - FORCE/DISPL. MULTIPLIER                   MAN736C
C        COL 21 TO 30     2 - FORCE/DISPL. MULTIPLIER                   MAN736C
C        COL 31 TO 40     3 - FORCE/DISPL. MULTIPLIER                   MAN736C
C        COL 41 TO 50     4 - FORCE USER MULTIPLIER                     MAN736C
C        COL 51 TO 60     5 - FORCE USER MULTIPLIER                     MAN736C
C        COL 61 TO 70     6 - FORCE USER MULTIPLIER                     MAN736C
C        COL 71 TO 80     USER CONSTANT.                                MAN736C
C                                                                       MAN736C
C                                                                       MAN736C
C     9.) OUTPUT CONTROL FOR LIMITED PRINTS                             MAN737C
C                                                                       MAN738C
C        DISPLACEMENT OUTPUT CONTROL, IF IOUT .NE. 0.                   MAN739C
C                                                                       MAN740C
C        CARD 1.  (I5)                                                  MAN741C
C                                                                       MAN742C
C        COL  1 TO  5     NUMDIS - NUMBER OF DISPLACEMENT PRINT CARDS   MAN743C
C                                                                       MAN744C
C        SUBSEQUENT CARDS (2I5) SKIP IF NUMDIS = 0                      MAN745C
C                                                                       MAN746C
C        COL  1 TO  5     NODAL NUMBER TO BE OUTPUT.                    MAN747C
C        COL  6 TO 10     HIGHER NODE NUMBER OF A GENERATED SEQUENCE,   MAN748C
C                         IF ZERO JUST FIRST NODE IS COUNTED.           MAN749C
C        COL 11 TO 15     INCREMENT TO GENERATOR, DEFAULT = 1           MAN750C
C                         *** REPEAT UNTIL,NUMDIS CARDS HAVE BEEN READ  MAN751C
C                                                                       MAN752C
C        STRESS OUTPUT CONTROL, IF IOUT .NE. 0.                         MAN453C
C                                                                       MAN754C
C        CARD 1.  (I5,5X,9I1)                                           MAN755C
C                                                                       MAN756C
C        COL  1 TO  5     NUMSTR - NUMBER OF STRESS OUTPUT CARDS        MAN757C
C        COL 11 TO 19     NSIG(9) - PRINT PATTERN WITHIN AN ELEMENT.    MAN758C
C                         LOCAL POINTS OF EACH ELEMENT CAN BE           MAN759C
C                         SUPRESSED BY NON-ZERO ENTRIES AS FOLLOWS.     MAN760C
C                                                         E.G.          MAN761C
C             COL  11     SUPRESS PRINT AT LOCAL POINT 1, ( 0, 0, 0)    MAN762C
C             COL  12     SUPRESS PRINT AT LOCAL POINT 2, (-1, 0, 0)    MAN763C
C             COL  13     SUPRESS PRINT AT LOCAL POINT 3, ( 1, 0, 0)    MAN764C
C             COL  14     SUPRESS PRINT AT LOCAL POINT 4, ( 0,-1, 0)    MAN765C
C             COL  15     SUPRESS PRINT AT LOCAL POINT 5, ( 0, 1, 0)    MAN766C
C             COL  16     SUPRESS PRINT AT LOCAL POINT 6, ( 0, 0,-1)    MAN767C
C             COL  17     SUPRESS PRINT AT LOCAL POINT 7, ( 0, 0, 1)    MAN768C
C             COL  18     SUPRESS PRINT AT LOCAL POINT 8                MAN769C
C             COL  19     SUPRESS PRINT AT LOCAL POINT 9                MAN770C
C                                                                       MAN771C
C        SUBSEQUENT CARDS (2I5) SKIP IF NUMSTR = 0                      MAN772C
C                                                                       MAN773C
C        COL  1 TO  5     ELEMENT NUMBER TO BE PRINTED.                 MAN774C
C        COL  6 TO 10     HIGHER ELEMENT NUMBER OF A GENERATED SEQUENCE, MAN775C
C                         IF ZERO ONLY FIRST ELEMENT IS COUNTED.        MAN776C
C        COL 11 TO 15     INCREMENT TO GENERATOR, DEFAULT = 1           MAN777C
C                         *** REPEAT UNTIL,NUMSTR CARDS HAVE BEEN READ  MAN778C
C                                                                       MAN779C
C                                                                       MAN780C
C**********************************************************************MAN781C
```

C. Listing of the Contact/Impact Subroutines Added to FEAP 74

The listings for the subroutines which are added to FEAP 74 for the contact/impact theory described herein are given below.

```
              SUBROUTINE CONTAC(ISW,IX,NEL1,NDF,NUMNP,DU,U,UD,UDD)
       C****************** CONTAC ***** 04/29/74 ******************************
000014        LOGICAL CFLAG,FLAGC
000014        DIMENSION IX(NEL1,NUMNP),DU(NDF,NUMNP),U(NDF,NUMNP),UD(NDF,NUMNP),
             C          UDD(NDF,NUMNP)
000014        DIMENSION RMP(10)
000014        COMMON/CNTACT/FLAGC,CFLAG,LIST,ICLIST(10),ICDEG(10),RU1,RU2,
             C    RM(2,10)
000014        COMMON/LABELS/HEAD(12),O,IPG,XHED(3),UHED(6),XH,FH,UH,NSTR,FLAG(7)
000014        COMMON/TAPES/ ITP5,ITP6
000014        DATA CFLAG,FLAGC/.FALSE.,.FALSE./
000014        GO TO (1,2,3),ISW
000022   1    READ(ITP5,1000) LIST,RU1,RU2
000034        RP = RU1+RU2
000036        IF(RP.EQ.0.0) RP = 1.
000044        READ(ITP5,1001) (ICLIST(L),ICDEG(L),L=1,LIST)
000070        WRITE(ITP6,2000) O,HEAD,IPG,RU1,RU2,(ICLIST(L),ICDEG(L),L=1,LIST)
000131        IPG = IPG + 1
000133        FLAGC = .TRUE.
000134        RETURN
000134   2     DO 200 L = 1,LIST
000141        N = ICLIST(L)
000143        IDEG = ICDEG(L)
000144        I = IX(1,N)
000150        RM(1,L) = DU(IDEG,I)
000155        I = IX(3,N)
000161   200  RM(2,L) = DU(IDEG,I)
000171        WRITE(ITP6,2002)
000175        DO 220 L = 1,LIST
000202        N = ICLIST(L)
000204        RMP(L) = RM(1,L) + RM(2,L)
000210        IF(RM(1,L).EQ.0.0.OR.RM(2,L).EQ.0.0) RMP(L) = RMP(L)*1.E+30
000222   220  WRITE(ITP6,2001) N,(RM(I,L),I=1,2),RMP(L)
000256        DO 210 N = 1,NUMNP
000260        DO 210 I = 1,NDF
000261   210  DU(I,N) = 0.0
000272        RETURN
000272   3    DO 300 L = 1,LIST
000274        N=ICLIST(L)
000276        IDEG = ICDEG(L)
000277        K = IX(2,N)
000303        UD(IDEG,K) = 0.0
000310        UDD(IDEG,K) = 0.0
000313        IF(U(IDEG,K)) 300,300,310
000317   310  I = IX(1,N)
000323        J = IX(3,N)
000326        CU =(RU1*UD(IDEG,I)+RU2*UD(IDEG,J))/RP
000342        UD(IDEG,I) = CU
000347        UD(IDEG,J) = CU
000352        CU=(RM(1,L)*UDD(IDEG,I)+RM(2,L)*UDD(IDEG,J))/RMP(L)
000375        IF(RM(1,L).GT.RM(2,L))U(IDEG,K)=U(IDEG,K)-RM(2,L)*(UDD(IDEG,J)-CU)
000420        IF(RM(1,L).LE.RM(2,L))U(IDEG,K)=U(IDEG,K)+RM(1,L)*(UDD(IDEG,J)-CU)
000443        UDD(IDEG,I) = CU
```

```
000451          UDD(IDEG,J) = CU
000454    300   CONTINUE

000457          RETURN

000457   1000   FORMAT(I5,5X,2F10.0)
000457   1001   FORMAT(2I5)
000457   2000   FORMAT(A1,12A6,30X,4HPAGE,I4//5X,25HCONTACT SHEET DESCRIPTION//
               C      10X,13HBODY 1 RO*U =,E13.5/
               C      10X,13HBODY 2 RO*U =,E13.5//
               X   11X,7HELEMENT,1X,9HDIRECTION/(I15,I10))
000457   2001   FORMAT(I10,6E13.4)
000457   2002   FORMAT(/10H    ELEMENT,13H  BODY 1 MASS,13H  BODY 2 MASS)
000457          END
```

```
                 SUBROUTINE ELMT05(N,MA,NDIM,NDF,NEL,NEL1,NSTF,NSIZV,NVEC,MCT,DM,D,        ELM  1C
                X XYZ,IX,F,FORCE,ESTIF,U,VECT,ISW)
            C****************** ELMT05 ***** 04/29/74 ******************************
000027           LOGICAL NPR
000027           DIMENSION ESTIF(NSTF,NSTF),FORCE(NSTF,2)
000027           DIMENSION U(NDF,1),IX(NEL1,1),PLOT(8),THED(8)
000027           COMMON/LOCALS/ DUL(6,20),UL(6,20),UDL(6,20),UDDL(6,20)
000027           COMMON/PRTPLT/ NSIG(9),NPLT(9,2),NT,NSTEP,NUMPLT,NEDATA(20,3),NPR
000027           COMMON/SHAP/ XJAC,SHAPE(4,20),SG(3,3),SK(3,3),X(3,20)
000027           COMMON/TAPES/ ITP5,ITP6
000027           COMMON/TIMHIS/ TIME,DT,DTP,NH,ISZH,C0,C1,C2,C3,C4,C5,C6,NCT
000027           EQUIVALENCE (PLOT,TAU),(PLOT(2),ETA)
000027           DATA THED/3HTAU,3HETA,2HU1,3HUD1,4HUDD1,2HU3,3HUD3,4HUDD3/
000027           GO TO (1,2,3,3,5,3), ISW
000041    1      READ(ITP5,1000) IDEG,NPS
000051           IDEG = MAX0(IDEG,1)
000055           WRITE(ITP6,2000) IDEG,NPS
000064           TOL = 1.E-06
000066           I = NDF  + IDEG
000073           J = I + NDF
000074           RI = 0.5
000075           RJ = 0.5
000076           C6 = 0.
000077           NV = NPS
000100           NA = NPS + NPS
000101           RETURN
000102    2      DM = X(IDEG,3) - X(IDEG,1)
000106           DM = DM + 1.E-25
000110           RETURN
000110    3      TAU = UL(IDEG,2)
000112           DD = (X(IDEG,3)-X(IDEG,1)) + (UL(IDEG,3)-UL(IDEG,1))
000120           ETA = 0.
000121           IF(DD.LT.TOL) ETA = 1.
000125           IF(TAU.LT.0.0  ) ETA = 0.0
000130           IF(ISW.GT.3) GO TO 4
000134           ESTIF(IDEG,I) = ETA
000142           ESTIF(I,IDEG) = ETA
000145           ESTIF(I,J) = -ETA
000151           ESTIF(J,I) = -ETA
000155           ESTIF(I,I) = 1.0 - ETA
000163           RETURN
000163    4       IF(ISW.EQ.4) GO TO 6
000166           FORCE(I,1) = ETA*DD
000174           IF(ETA.EQ.0.0) FORCE(I,1) = - TAU
000203           TAU = AMAX1(0.0,TAU)
000207           FORCE(IDEG,1) = -TAU
000215           FORCE(J,1) = TAU
000220    6      IF(.NOT.NPR) WRITE(ITP6,2001) N,DM,MA,TAU,ETA
000243           IF(NUMPLT.LE.0) RETURN
000246           PLOT(3) = UL(IDEG,1)
000250           PLOT(4) = UDL(IDEG,1)
000252           PLOT(5) = UDDL(IDEG,1)
000254           PLOT(6) = UL(IDEG,3)
```

```
000256          PLOT(7) = UDL(IDEG,3)
000260          PLOT(8) = UDDL(IDEG,3)
000262          DO 60 K = 1,NUMPLT
000263          KK = NPLT(K,2)
000265          IF(NPLT(K,1).GT.0) CALL PLDATA(NDIM,NPLT(K,1),THED(KK),X(1,2),
               C   PLOT(KK))
000306    60    CONTINUE
000311    5     RETURN
000312    1000  FORMAT(2I5)
000312    2000  FORMAT(5X,21HPOINT CONTACT ELEMENT/
               C         5X,27HCONTACT DEGREE OF FREEDOM = I3,5X,5HNPS = I5)
000312    2001  FORMAT(5X,7HELEMENT,I5,5X,A5,5X,8HMATERIAL,I3,5X,5HTAU =,E15.5,5X,
               X   5HETA =,F3.1)
000312          END
```

```
               SUBROUTINE ELMT09(N,MA,NDIM,NDF,NELM,NEL1,NSTF,NSIZV,NVEC,MCT,DM,       ELM  1C
              X D,XYZ,IX,F,FORCE,ESTIF,U,VECT,ISW)                                      ELM  2C
000027         LOGICAL NOPRNT,NPR,NPL
000027         DIMENSION ESTIF(NSTF,NSTF),D(3,21,1),IX(NEL1,1),U(NDF,1),                ELM  4C
              X V(3,20),DU(3),XX(3),FORCE(NSTF,2)                                       ELM  5C
000027         COMMON/GAUS/ LIM,SGAUSS(5,5),WGAUSS(5,5)                                  ELM  8C
000027         COMMON/LABELS/HEAD(12),O,IPG,XHED(3),UHED(6),XH,FH,UH,NSTR,FLAG(7)
000027         COMMON/LOCALS/ DUL(6,20),UL(6,20),UDL(6,20),UDDL(6,20)
000027         COMMON/PRTPLT/ NSIG(9),NPLT(9,2),NT,NSTEP,NUMPLT,NEDATA(20,3),NPR
000027         COMMON/SHAP/ XJAC,SHAPE(4,20),SG(3,3),SK(3-3),X(3,20)
000027         COMMON/TAPES/ ITP5,ITP6                                                  ELM 10C
000027         COMMON/TIMHIS/ TIME,DT,DTP,NH,ISZH,C0,C1,C2,C3,C4,C5,C6,NCT
000027         DATA SH,EH,BL/6HSTRESS,6HSTRAIN,6H      /                                ELM 11C
000027         GO TO (1,2,3,4,5,4) , ISW
000041    1    CONTINUE                                                                 ELM 13C
         C.... BAR STIFFNESS CHARACTERIZATION                                           ELM 14C
000041         READ(ITP5,1000) E,AA,RO
000053         WRITE(ITP6,2100) E,AA,RO
000065         D(1,1,MA) = E                                                            ELM 17C
000075         D(1,2,MA)=AA
000100         D(1,3,MA)=RO
000103         C6 = 0.
000104         RETURN                                                                   ELM 20C
000104    2    CONTINUE                                                                 ELM 21C
000104         RETURN                                                                   ELM 22C
         C.... COMPUTATION OF BAR STIFFNESS NO BENDING * ISOPARAMETRIC 1ST AND 2N      ELM 23C
000105    3    CONTINUE                                                                 ELM 24C
         C.... CONSTRUCT STIFFNESS AT INTEGRATION POINTS                                ELM 25C
000105         EA = D(1,1,MA)*D(1,2,MA)                                                 ELM 26C
000113         RA = D(1,3,MA)*D(1,2,MA)                                                 ELM 27C
000116         DO 260 II = 1,NELM                                                       ELM 28C
000120         SS = SGAUSS(II,NELM)                                                     ELM 29C
000123         CALL LINE(SS,NDIM,NELM)                                                  ELM 30C
000125         WW=WGAUSS(II,NELM)
000134         DVOL=EA*WW/(XJAC*XJAC*XJAC)
         C.... COMPUTE A LUMPED MASS MATRIX                                             ELM 32C
000137         IU = 0                                                                   ELM 33C
000140         RAA=RA*XJAC*WW
000142         DO 217 I = 1,NELM                                                        ELM 34C
000143         AA=RAA*SHAPE(2,I)
000146         DO 215 KK = 1,NDIM                                                       ELM 35C
000150  215    FORCE(IU+KK,2) = FORCE(IU+KK,2) + AA                                     ELM 37C
000162  217    IU = IU + NDF                                                            ELM 38C
000166         DO 250 KK = 1,NDIM                                                       ELM 39C
000167         DO 250 LL = KK,NDIM                                                      ELM 40C
000170         SGD=SG(KK,LL)*DVOL
000174         I1 = KK                                                                  ELM 41C
000175         DO 240 I = 1,NELM                                                        ELM 42C
000177         SHI=SHAPE(1,I)*SGD
000202         L = 1                                                                    ELM 43C
000203         IF(KK.EQ.LL) L = I                                                       ELM 44C
000206         J1 = LL + NDF*(L-1)                                                      ELM 45C
000213         DO 230 J = L,NELM                                                        ELM 46C
```

```
000214          ESTIF(I1,J1)=ESTIF(I1,J1)+SHI*SHAPE(1,J)
000225    230   J1 = J1 + NDF
000231    240   I1 = I1 + NDF                                              ELM 48C
000234    250   CONTINUE                                                   ELM 49C
000241    260   CONTINUE                                                   ELM 50C
          C.... CONSTRUCT SYMMETRIC PART OF STIFFNESS MATRIX                ELM 51C
000243          KK = NELM*NDF                                              ELM 52C
000245          LL = KK - 1                                                ELM 53C
000247          DO 360 I=1,LL                                              ELM 54C
000250          K = I + 1                                                  ELM 55C
000252          DO 360 J = K,KK                                            ELM 56C
000254          ESTIF(I,J) = ESTIF(I,J) + ESTIF(J,I)                       ELM 57C
000266    360   ESTIF(J,I) = ESTIF(I,J)                                    ELM 58C
000302          IF(C6.EQ.0.0) RETURN                                       ELM 59C
000304          DO 370 I = 1,NSTF
000306    370   ESTIF(I,I) = ESTIF(I,I) + C6*FORCE(I,2)
000324          RETURN
000325    4     CONTINUE                                                   ELM 60C
000325          E = D(1,1,MA)                                              ELM 61C
000331          NELP = NELM                                                ELM 64C
000332          DO 400 NN = 1,NELP
000334          NOPRNT=NPR
000336          IF(NSIG(NN).GT.0) NOPRNT=.TRUE.
000341          NPL=.TRUE.
000342          IF(NPLT(NN,1).GT.0) NPL = .FALSE.
000345          SS = SGAUSS(NN,NELP)
000351          CALL LINE(SS,NDIM,NELM)
          C.... COMPUT(E STRESS AND STRAIN                                  ELM 71C
000353          DO 100 KK = 1,NDIM                                         ELM 72C
000350          XX(KK) = 0.0                                               ELM 73C
000361          DU(KK)= 0.                                                 ELM 74C
000362          DO 100 LL = 1,NELM                                         ELM 75C
000364          XX(KK) = XX(KK) + X(KK,LL)*SHAPE(2,LL)                     ELM 76C
000372    100   DU(KK) = DU(KK) +UL(KK,LL)*SHAPE(1,LL)                     ELM 77C
000404          EPS = 0.
000405          DO 200 KK = 1,NDIM                                         ELM 79C
000406    200   EPS = EPS + SK(KK,1)*DU(KK)/XJAC/XJAC                      ELM 80C
000416          SIG = E*EPS                                                ELM 81C
000420          IF(NOPRNT) GO TO 350                                       ELM 82C
000421          MCT = MCT - 1
000423          IF(MCT.GT.0) GO TO 300                                     ELM 83C
000425          MCT=50                                                     ELM 84C
000426          WRITE(ITP6,2000) HEAD,TIME,IPG,( XHED(I),XH,I=1,NDIM),BL,SH,BL,EH
000465          IPG = IPG + 1
000467    300   WRITE(ITP6,2001) N,MA,DM,(XX(I),I=1,NDIM),SIG,EPS          ELM 87C
000523    350   IF(.NOT.NPL) CALL PLDATA(NDIM,NPLT(NN,1),SHAXIAL,XX,SIG)   ELM 88C
000542          IF(ISW.EQ.4) GO TO 400
000545          WT = WGAUSS(NN,NELM)*D(1,2,MA)
000553          WW = WT*D(1,3,MA)*XJAC
000557          IU = 0
000560          DO 600 I = 1,NELM
000562          AF = WW*SHAPE(2,I)
000565          BB = SHAPE(1,I)*SIG*WT/XJAC
000572          DO 610 KK = 1,NDIM
```

```
000573   615   FORCE(KK+IU,1) = FORCE(KK+IU,1) - AA*UDDL(KK,I) - BB*SK(KK,I)
000615   600   IU = IU + NDF
000621   400   CONTINUE                                                        ELM 91C
000624   5     RETURN
000625   1000  FORMAT(3F10.0)                                                  ELM 93C
000625   2000  FORMAT(1H1,12A6,E13.5,17X,4HPAGE,I3//10H   ELEMENT,4X,8HMATERIAL,
              X 3X,5(2A6))
000625   2001  FORMAT(I10,I5,5X,A5,5E12.4)                                     ELM 96C
000625   2100  FORMAT(37H0LINEAR ELASTIC MATERIAL, BAR ELEMENT//
              X        5X,3HE =,E15.5,5X,6HAREA =,E15.5,5X,9HDENSITY =E15.5/)  ELM 97C
000625   4000  FORMAT(A5,I5,1P3E12.5,24X,I10)                                  ELM 98C
000625         END                                                             ELM 99C
```

```
RUN FORTRAN COMPILER VERSION 2.3 B.3                                    06 MAY 74 14:09:30  PAGE NO. 1

              SUBROUTINE LINE(S,NDIM,NEL)                                             LIN  1C
        C***** LINE       ****** 7/02/73 ***************************************
        C.... SHAPE FUNCTION ROUTINE FOR 1ST AND 2ND ORDER ISOPARAMETRIC LINES         LIN  2C
000006        COMMON/SHAP/ XJAC,SHAPE(4,20),SG(3,3),SK(3,3),X(3,20)
        C.... FORM SHAPE FUNCTIONS                                                     LIN  4C
000006        SHAPE(1,1) = -0.5                                                        LIN  5C
000007        SHAPE(2,1) = 0.5*(1.0-S)                                                 LIN  6C
000012        SHAPE(1,NEL) = 0.5                                                       LIN  7C
000014        SHAPE(2,NEL) = 0.5*(1.0+S)                                               LIN  8C
000016        IF(NEL.EQ.2) GO TO 350                                                   LIN  9C
000020        SHAPE(1,2) = -2.0*S                                                      LIN 10C
000021        SHAPE(2,2) = 1.0-S*S                                                     LIN 11C
000024        K = NEL-1                                                                LIN 12C
000026        DO 100 I = 1,NEL,K                                                       LIN 13C
000027        SHAPE(1,I) = SHAPE(1,I) - 0.5*SHAPE(1,2)                                 LIN 14C
000033  100   SHAPE(2,I) = SHAPE(2,I) - 0.5*SHAPE(2,2)                                 LIN 15C
        C.... FORM JACOBIAN MATRIX AND DETERMINANT                                     LIN 16C
000040  350   DO 360 I = 1,3                                                           LIN 17C
000042  360   SK(I,1) = 0.0                                                            LIN 18C
000046        DO 400 I = 1,NDIM                                                        LIN 19C
000047        DO 400 J = 1,NEL                                                         LIN 20C
000050  400   SK(I,1) = SK(I,1) + X(I,J)*SHAPE(1,J)                                    LIN 21C
000063        XJAC = 0.0                                                               LIN 22C
000064        DO 500 I = 1,NDIM                                                        LIN 23C
000065        XJAC = XJAC + SK(I,1)*SK(I,1)                                            LIN 24C
000070        DO 500 J = 1,NDIM                                                        LIN 25C
000071  500   SG(I,J) = SK(I,1)*SK(J,1)                                                LIN 26C
000103        XJAC = SQRT(XJAC)                                                        LIN 27C
000105        RETURN                                                                   LIN 28C
000105        END                                                                      LIN 29C
```

```
              SUBROUTINE NLNORM(ISW,MDEG,NN,U,DU,DF,F,PROP,NCT,NTT,IDEST,IBLK)
         C***** NLNORM    ****** 12/14/73   ************************************
000017        DIMENSION U(1),DU(MDEG),DF(1),F(MDEG),IDEST(1)
000017        COMMON/NORMS/ UNORM,DNORM,ANORM,CS,CSP,DP,DNP,IFL,XFLAG,ERR
000017        GO TO (100,200,200,100), ISW
000026    100 CS=1.0
000027        CSP=1.0
000030        DP=0.75
000032        IFL=0
000032        XFLAG=.FALSE.
000033        ERR=1.0E-3
000035        RETURN
000036    200 UNORM = 0.
000037        DNORM = 0.
000037        ANORM=0.
000041        DO 500 N = 1,MDEG
000042        DUN = DU(N)
000044        UN = U(N)
000046        IF(ISW.EQ.3) UN = UN + DF(N) + DUN
000053        ANORM=ANORM+UN*DUN
000056        UNORM=UNORM+UN*UN
000060    500 DNORM=DNORM+DUN*DUN
000065        UNORM=SQRT(UNORM)
000067        DNORM=SQRT(DNORM)
000070        IF(NN.EQ.1) DNP=UNORM
000077        CS=1.0
000100        IF(UNORM*DNORM.NE.0.) CS=ANORM/(UNORM*DNORM)
000105        IF(IFL.EQ.1) DP=0.75
000111        IF(IFL.EQ.1.AND.XFLAG) CSP=CS
000120        IFL=0
000121        IF(DNORM.LE.0.5*UNORM) GO TO 550
000124        IFL=1
000125        DP=0.5*UNORM/DNORM
000127    550 IF(DP*DNORM.GT.DNP) DP=DNP/DNORM
000134        IF(DP.EQ.0.0) DP = 1.0
000136        IF(CS*CSP.LT.0.) DP=DP/2.
000142        IF(CS*CSP.GT.0.) DP=1.25*DP
000146        DNP=DP*DNORM
000150        CSP=CS
000150        IF(ISW.EQ.2) RETURN
000153        DO 600 N = 1,MDEG
000155        DF(N) = DF(N) + DU(N)
000160        IF(NCT.EQ.0) DF(N)= DU(N)
000164    600 DU(N) = DF(N)
000171        RETURN
000171        END
```

```
             SUBROUTINE TSOLVE(NUMNP,NUMEL,NUMMAT,NDIM,NDF,NEN,NEL1,NSTF,NVEC,    TSO  1C
            1   NSIZV,NICD,NSICD,IBLK,ISZA,NEQB,MAXBAN,NDEG,ID1,M8,IBUF,DS,       TSO  2C
            2  TYPE,D,ICOD,XYZ,F,IX,IDEST,VECT,FORCE,ESTIF,LD,A,DU,MAXB,H,U,DF,   TSO  3C
            3   NSEQ,TYME)                                                        TSO  4C
       C***** TSOLVE    ****** 03/06/74  ************************************     TSO  5C
000053       LOGICAL CFLAG,FLAGC
000053       LOGICAL NPR,VFLAG,DFLAG                                              TSO  6C
000053       REAL LABL(12)
000053       DIMENSION TYPE(1),ICOD(1),XYZ(NDIM,1),F(NDF,1),IX(NEL1,1),           TSO  8C
            1   IDEST(NDF,1),VECT(NSIZV,1),FORCE(NSTF,2),ESTIF(NSTF,NSTF),        TSO  9C
            2   A(NEQB,1),DU(NDF,1),H(IBUF),U(NDF,1),DF(1),DS(M8),TYME(7),LD(1)   TSO 10C
000053       COMMON/CNTACT/FLAGC,CFLAG,LIST,ICLIST(10),ICDEG(10),RU1,RU2,
            C      RM(2,10)
000053       COMMON/LABELS/HEAD(12),O,IPG,XHED(3),UHED(6),XH,FH,UH,NSTR,FLAG(7)   TSO 11C
000053       COMMON/LOCALS/ DUL(6,20),UL(6,20),UDL(6,20),UDDL(6,20)               TSO 12C
000053       COMMON/NORMS/ UNORM,DNORM,ANORM,CS,CSP,DP,DNP IFL,XFLAG,ERR          TSO 13C
000053       COMMON/PRTPLT/ NSIG(9),NPLT(9,2),NT,NSTEP,NUMPLT,NEDATA(20,3),NPR    TSO 14C
000053       COMMON/SHAP/ XJAC,SHAPE(4,20),SG(3,3),SK(3,3),X(3,20)                TSO 15C
000053       COMMON/TAPES/ ITP5,ITP6                                              TSO 16C
000053       COMMON/TIMHIS/ TIME,DT,DTP,NH,ISZH,C0,C1,C2,C3,C4,C5,C6,NCT          TSO 17C
000053       COMMON/VTAPE/ ITP13,ITP14,ITPD,ITWR                                  TSO 18C
000053       DATA AWORD1,AWORD2/8H  F13.4,,8H  E13.4,/,AWORD3/8H  F12.2,/
000053       DATA VH,AH,AWORD4,AWORD5/6H VEL. ,6H ACCEL ,6H(/12X,,8H9E13.4))/
000053       DATA LABL(1)/8H (I12,  /                                             TSO 20C
000053       DATA CFLAG,FLAGC/.FALSE.,.FALSE./
       C.... SUBROUTINE TO PERFORM IMPLICIT INTEGRATION WITH 1 OR 3 INIT. COND.   TSO 21C
000053       DO 2 N=1,NEN                                                         TSO 22C
000054       DO 2 J=1,NDF                                                         TSO 23C
000055       DUL(J,N)=0.                                                          TSO 24C
000060       UL(J,N) = 0.0                                                        TSO 25C
000063       UDL(J,N)=0.                                                          TSO 26C
000065 2     UDDL(J,N) = 0.0                                                      TSO 27C
000073       DO 3 I=2,7                                                           TSO 28C
000074 3     NSIG(I)=1                                                            TSO 29C
000077       IF(IBLK.EQ.0) GO TO 4                                                TSO 30C
000101       KU = NUMNP*NICD                                                      TSO 31C
000104       REWIND 7                                                             TSO 32C
000106       WRITE(7) DS                                                          TSO 33C
       C.... SET THE OUTPUT LABELS FOR DISPLACEMENT PRINTS                        TSO 34C
000117 4     HEAD(12) = 6H TIME=                                                  TSO 35C
000124       IFLG =-NSICD                                                         TSO 36C
000126       MDEG = NSICD                                                         TSO 37C
000127       MDEG2 = MDEG + MDEG                                                  TSO 38C
000130       I = 2                                                                TSO 39C
000131       DO 5 N = 1,NDIM                                                      TSO 40C
000132       LABL(I) = AWORD1                                                     TSO 41C
000134 5     I = I + 1                                                            TSO 42C
000137       L = NDIM + 1
000140       DO 7 N = L,9
000141       LABL(I) = AWORD2
000143 7     I = I + 1
000146       LABL(I) = AWORD4
000150       LABL(I+1) = AWORD5
```

```
000152          DTP = 0.
000153          TIME = 0.                                                TSO 44C
000154          MB = MAXBAN + 1                                          TSO 45C
000156          NE1 = NEN + 1                                            TSO 46C
       C....    INITIALIZE INCREMENTAL FORCE VECTOR                      TSO 47C
000157          DO 6 N = 1,NSICD                                         TSO 48C
000160   6      DF(N) = 0.0                                              TSO 49C
000165          DO 10 N = 1,NUMEL                                        TSO 50C
000166   10     IX(NE1,N) = 0                                            TSO 51C
       C....    PUT INITIAL DATA ON TAPE                                 TSO 52C
000177          ISZH = IBUF                                              TSO 53C
000201          ITRD = ITP13                                             TSO 54C
000202          ITWR = ITP14                                             TSO 55C
000204          REWIND ITRD                                              TSO 56C
000206          IF(IBLK.GT.0) WRITE(ITRD) ((U(I,J),I=1,NDF),J=1,KU)      TSO 57C
000241          DO 11 N = 1,ISZH                                         TSO 58C
000243   11     H(N) = 0.                                                TSO 59C
000247          NTB= (NH+ISZH-1)/ISZH                                    TSO 60C
000253          WRITE(ITRD) H                                            TSO 61C
000264          IF(NTB.LE.1) GO TO 13                                    TSO 62C
000272          NT = NTB*2                                               TSO 63C
000273          DO 12 N = 1,NT                                           TSO 64C
000275   12     WRITE(ITRD) H                                            TSO 65C
000314   13       REWIND ITRD                                            TSO 66C
000316          IF(IBLK.GT.0) READ(ITRD) DM                              TSO 67C
000331          NEP = NUMNP + NUMNP                                      TSO 68C
000332          C6 = 0.0                                                 TSO 69C
000333          NSTEP = 0                                                TSO 70C
000334          NST = 0                                                  TSO 71C
000335          NUMPLT=0                                                 TSO 72C
000336          PROP = 0.                                                TSO 73C
000337          CFLAG = .FALSE.                                          TSO 74C
000340          DO 900 M = 1,NSEQ
000341          READ(ITP5,1000) DT,NTS,INT,NNI,NNF,NEI,NEF,NPROP,NFORC,C0,DM,NT,NL   TSO 75C
000376          IF(NICD. GT.1) C6= DM                                    TSO 76C
000406          IF(NL.NE.0) NST= NL                                      TSO 77C
000410          NTT = IABS(NST)                                          TSO 78C
000412          DFLAG = .FALSE.                                          TSO 79C
000413          IF (NTT. NE. 0 )        DFLAG = .TRUE.                   TSO 80C
000415          NSTEP=NSTEP+NTS                                          TSO 81C
000417          IF(M.EQ.1) NUMPLT=NT                                     TSO 82C
000422          IF(INT.LE.0) INT = 1                                     TSO 83C
000425          WRITE(ITP6,2000) O,HEAD,TIME,IPG,DT,NTS,INT,NNI,NNF,NEI,NEF   TSO 84C
000457          IF(DFLAG) WRITE(ITP6,2032) NST                           TSO 85C
000476          IPG = IPG + 1                                            TSO 86C
000500          IF(NICD.EQ.1) GO TO 1                                    TSO 87C
000501          IF(DT.EQ.0.0) GO TO 901                                  TSO 88C
000502          CALL UPDATE(3)                                           TSO 89C
000504   1      IF(NPROP.GT.0) PROP = PROPLD(TIME,NPROP)                 TSO 90C
000517          IF(NFORC.GT.0.AND.NPROP.GT.0) WRITE(ITP6,4001)           TSO 91C
       C....    READ STRESS PLOT INFORMATION                             TSO 92C
000536          IF(NUMPLT.LE.0.OR.M.GT.1) GO TO 590                      TSO 93C
000547          REWIND 12                                                TSO 94C
000551          WRITE(ITP6,2005)                                         TSO 95C
                                                                         TSO 9[illegible]C
```

```
000555          DO 501 N=1,NUMPLT
000562          READ(ITP5,1006) (NEDATA(N ,I),I=1,3)                                    TS0 97C
000600   501    WRITE(ITP6,2006) N,(NEDATA(N ,I),I=1,3)                                 TS0 98C
000632          CALL PLZERO                                                             TS0 99C
000633   500    VFLAG = .FALSE.                                                         TS0100C
000634          DO 600 NT = 1,NTS                                                       TS0101C
000641          PPROP = PROP                                                            TS0102C
000643          PROP = 1.0                                                              TS0103C
000644          IF(DFLAG)CALL NLNORM(4,MDEG,NT,U,DF,DU,F,PPROP,NCT,NTT,IDEST,IBLK)      TS0104C
000667          IF(NPROP.GT.0) PROP = PROPLD(TIME+DT,0)                                 TS0105C
000701          IF(NFORC.GT.0) CALL RESET(-NFORC,NUMNP,NDF,F)                           TS0106C
000713          NCT = 0                                                                 TS0107C
000714   14     REWIND ITWR                                                             TS0108C
000716          IF(IBLK.EQ.0) GO TO 15                                                  TS0109C
000720          IF(.NOT.VFLAG) GO TO 20                                                 TS0110C
000721          DO 21 I = 1,NDEG                                                        TS0111C
000723   21     DF(I) = 0.                                                              TS0112C
000730          GO TO 20                                                                TS0113C
000730   15     DO 19 N = 1,NUMNP                                                       TS0114C
000732          DO 18 K = 1,NDF                                                         TS0115C
000733          J = IDEST(K,N)                                                          TS0116C
000740          IF(J.EQ.0) GO TO 18                                                     TS0117C
000741          IF(VFLAG) GO TO 17                                                      TS0118C
000743          DO 16 I = 1,MAXBAN                                                      TS0119C
000744   16     A(J,I) = 0.0                                                            TS0120C
000756   17     DF(J) = F(K,N)*PROP                                                     TS0121C
000766   18     CONTINUE                                                                TS0122C
000771   19     CONTINUE                                                                TS0123C
000773   20     NH = 1                                                                  TS0124C
000774          IF(NTB.GT.1) READ(ITRD) H                                               TS0125C
001614          IF(NCT.GT.0) GO TO 44                                                   TS0126C
        C.... OUTPUT THE SOLUTION VECTOR FOR THE CURRENT TIME                           TS0127C
001017          IF(NNI.EQ.0.OR.((NT-1)/INT)*INT.NE.NT-1) GO TO 40                       TS0128C
001032          MCT = 0                                                                 TS0129C
001033          DO 30 N = NNI,NNF                                                       TS0130C
001034          MCT = MCT - 1                                                           TS0131C
001036          IF(MCT.GT.0) GO TO 31                                                   TS0132C
001040          IF(NICD.EQ.1) WRITE(ITP6,2001) O,HEAD,TIME,IPG,PPROP,M,NT,
               X   (XHED(I),XH,I=1,NDIM),(UHED(I),UH,I=1,NDF)
001121          IF(NICD.NE.1) WRITE(ITP6,2001) O,HEAD,TIME,IPG,PPROP,M,NT,
               X   (XHED(I),XH,I=1,NDIM),(UHED(I),UH,I=1,NDF),(UHED(I),VH,I=1,NDF)
               X  ,(UHED(I),AH,I=1,NDF)
001235          IPG = IPG + 1                                                           TS0136C
001237          MCT = 50                                                                TS0137C
001240   31     IF(NICD.EQ.1)WRITE(ITP6,LABL)N,(XYZ(I,N),I=1,NDIM),(U(I,N)
               X   ,I=1,NDF)
001311          IF(NICD.NE.1)WRITE(ITP6,LABL)N,(XYZ(I,N),I=1,NDIM),(U(I,N),I=1,NDF
               X ) ,(U(I,NUMNP+N),I=1,NDF),(U(I,NEP+N),I=1,NDF)
001420   30     CONTINUE
        C.... UPDATE (1) * * FOR DYNAMIC SOLUTIONS ONLY                                 TS0139C
001423   40     IF(NICD.GT.1) CALL UPDATE(1,MDEG,NICD,U,U(1,NUMNP+1),U(1,NEP+1),        TS0140C
               X   DU,F,DF,IDEST,PROP)                                                  TS0141C
001457          IF(IBLK.GT.0) WRITE(ITWR) ((U(I,J),I=1,NDF), J=1,KU )                   TS0142C
001512   44     MP = 0                                                                  TS0143C
```

```
001513          TEMP = 0.                                          TS0144C
001514          DO 400 N = 1,NUMEL                                 TS0145C
001515          NPR = .TRUE.                                       TS0146C
001516          DO 43 I = 1,9                                      TS0147C
001520          NPLT(I,1) = 0                                      TS0148C
001521     43   NPLT(I,2) = 0                                      TS0149C
001524          IF(NCT.GT.0) GO TO 46                              TS0150C
001526          IF(N.GE.NEI.AND.N.LE.NEF) NPR = .FALSE.            TS0151C
001540          IF(NUMPLT.LE.0) GO TO 46                           TS0152C
001542          DO 45 I = 1,NUMPLT                                 TS0153C
001543          IF(NEDATA(I,1).NE.N) GO TO 45                      TS0154C
001546          J = NEDATA(I,2)                                    TS0155C
001547          NPLT(J,1) = I                                      TS0156C
001551          NPLT(J,2) = NEDATA(I,3)                            TS0157C
001553     45   CONTINUE                                           TS0158C
001556     46   MA = MOD(IX(NEL1,N),100)                           TS0159C
001567          IF(MR.LE.0) MRR= IX(NEL1,N)/1000                   TS0160C
001600          IF(MR.LE.0) MR = MRR                               TS0161C
001603          DO 60 I = 1,NSTF                                   TS0162C
001605          FORCE(I,1) = 0.                                    TS0163C
001612          FORCE(I,2)=0.                                      TS0164C
001616          LD(I) = 0                                          TS0165C
001620          IF(MR.NE.MRR.OR.VFLAG) GO TO 60                    TS0166C
001625          DO 50 J = 1,NSTF                                   TS0167C
001627     50   ESTIF(I,J) = 0.0                                   TS0168C
001640     60   CONTINUE                                           TS0169C
001643          L = 0                                              TS0170C
001644          DO 110 I = 1,NEN                                   TS0171C
001645          K = IX(I,N)                                        TS0172C
001653          DO 90 J = 1,NDIM                                   TS0173C
001654     90   X(J,I) = 0.                                        TS0174C
001661          IF(K.EQ.0)  GO TO 120                              TS0175C
001662          NEL = I                                            TS0176C
001663          DO 100 J = 1,NDIM                                  TS0177C
001665    100   X(J,I) = XYZ(J,K)                                  TS0178C
001701          DO 110 J = 1,NDF                                   TS0179C
001702          L = L + 1                                          TS0180C
001704          DUL(J,I) = DU(J,K)                                 TS0181C
001713          UL(J,I) = U(J,K)                                   TS0182C
001722          IF(NCT.GT. 0) UL(J,I) =UL(J,I) +DUL(J,I)           TS0183C
001731          IF(NICD.EQ.1) GO TO 110                            TS0184C
001734          UDL(J,I) = U(J,K+NUMNP)                            TS0185C
001744          UDDL(J,I) = U(J,K+NEP)                             TS0186C
001754          IF(NCT.GT.0) UDDL(J,I)= UDDL(J,I) +C6*DUL(J,I)     TS0187C
001765    110   LD(L) = IDEST(J,K)                                 TS0188C
002001    120   DM = TYPE(MA)                                      TS0189C
002004          IF(DM.NE.TEMP) MCT = 0                             TS0190C
002007          TEMP = DM                                          TS0191C
002010          CALL TICTOC(TYME,6)                                TS0192C
          C.... COMPUTE ELEMENT STRESSES AND UPDATE FORCES         TS0193C
002013          CALL ELMLIB(N,MA,NDIM,NDF,NEL,NEL1,NSTF,NSIZV,NVEC,MCT,DM,D,XYZ,   TS0194C
               X IX,H,FORCE,ESTIF,U,VECT,6)                        TS0195C
002047          CALL TICTOC(TYME,5)                                TS0196C
          C.... FORM STIFFNESSIF NEEDED FOR THE NEXT TIME STREP    TS0197C
```

```
002052            IF((.NOT.VFLAG.AND.MR.EQ.MRR).OR.(VFLAG.AND.IX(NE1,N).EQ.1))         TS0198C
                 XCALL ELMLIB(N,MA,NDIM,NDF,NEL,NEL1,NSTF,NSIZV,NVEC,MCT,DM,D,XYZ,    TS0199C
                 X IX,H,FORCE,ESTIF,U,VECT,3)                                         TS0200C
002131            IF(MOD(IX(NEL1,N),1000)/100.GT.0.AND..NOT.VFLAG)                    TS0201C
                 X CALL PRTMAT(HEAD,IPG,N,NSTF,ESTIF,FORCE,LD,NSTF,0)                 .S0202C
            C.... MODIFY FOR THE DISPLACEMENT B.C.                                     TS0203C
002165            CALL MODIFY(NDF,NEL,NEL1,NE1,IBLK,NSTF,PROP,IX,ICOD,F,FORCE,ESTIF,  TS0204C
                 X   N)                                                                TS0205C
002205            IF(VFLAG) GO TO 300                                                  TS0206C
002212            IF(IBLK.EQ.0) CALL COREIN(A,DF,NEQB,ESTIF,FORCE,LD,NSTF)            TS0207C
002231            IF(IBLK.GT.0) WRITE(7) ESTIF,FORCE,LD                                TS0208C
002263            IF(.NOT.FLAGC.OR.CFLAG) GO TO 400
002266            L = 1
002267            DO 130 I = 1,NEL
002270            K = IX(I,N)
002276            DO 130 J = 1,NDF
002277            DU(J,K) = DU(J,K) + FORCE(L,2)
002311     130    L = L + 1
002317            GO TO 400                                                            TS0209C
002320     300    CONTINUE                                                             TS0210C
            C.... ADD THE FORCE TO THE SOLUTION FOR A RESOLVE                          TS0211C
002320            DO 310 K = 1,NSTF                                                    TS0212C
002322            J = LD(K)                                                            TS0213C
002325            IF(J.GT.0) DF(J) = DF(J) + FORCE(K,1)                                TS0214C
002335     310    CONTINUE                                                             TS0215C
002340     400    CALL TICTOC(TYME,3)                                                  TS0216C
002351            IF(FLAGC.AND..NOT.CFLAG) CALL CONTAC(2,IX,NEL1,NDF,NUMNP,DU)
002364            CFLAG = .TRUE.
002365            IF(NTB.GT.1) WRITE(ITWR) H                                           TS0217C
002405            IF(IBLK.GT.0) WRITE(ITWR) (IX(NE1,N),N=1,NUMEL)                      TS0219C
002436            IF(.NOT.VFLAG) CALL SOLVEQ(NUMNP,NUMEL,NDF,ID1,M8,MAXBAN,9,NSTF,     TS0220C
                 1   ISZA,NEQB,IBLK,A,DU,DF,IDEST,FORCE,ESTIF,LD,MAXB,NDEG)            TS0221C
002476            IF(VFLAG) CALL RESVEQ(NUMNP,NDF,M8,MAXBAN,ISZA,NEQB,IBLK,A,DF,DF,    TS0222C
                 1   IDEST,IDEST,MAXB,IFLG)                                            TS0223C
002524            CALL TICTOC(TYME,4)                                                  TS0224C
            C.... UPDATE THE SOLUTION                                                  TS0225C
002527            IF(NT.EQ.NTS.AND.M.EQ.NSEQ) GO TO 900                                TS0226C
002542            IF(NCT.GT.0) GO TO 410                                               TS0227C
002544            DTP = DT                                                             TS0228C
002545            I = ITRD                                                             TS0229C
002546            ITRD = ITWR                                                          TS0230C
002547            ITWR = I                                                             TS0231C
002550     410    IF(IBLK.GT.0) BACKSPACE ITRD                                         TS0232C
002554            IF(IBLK.GT.0) READ(ITRD) (IX(NE1,N),N=1,NUMEL)                       TS0233C
002605            REWIND ITRD                                                          TS0234C
002607            IF(IBLK.GT.0) READ(ITRD) ((U(I,J),I=1,NDF),J=1,KU)                   TS0235C
002642            IF(DFLAG)CALL NLNORM(3,MDEG,NT,U,DF,DU,F,PPROP,NCT,NTT,IDEST,IBLK)   TS0236C
002666            VFLAG = .TRUE.                                                       TS0237C
002667            IF(NST.LT.0) VFLAG = .FALSE.                                         TS0238C
002672            NCT = NCT + 1                                                        TS0239C
002674            IF(.NOT.DFLAG) GO TO 700
002675            WRITE(6,6000) NCT,UNORM,DNORM,ANORM
002711            IF(NCT.LT.NTT.AND.DNORM.GT.UNORM*EPR) GO TO 14                       TS0243C
002730     700    IF(DFLAG) VFLAG=.FALSE.
```

```
        C.... UPDATE (2)                                                          TS0245C
002733        CALL UPDATE(2,MDEG,NICD,U,U(1,NUMNP+1),U(1,NEP+1),DU,F,DF,IDEST,    TS0246C
             X   PROP)                                                            TS0247C
002761        IF(FLAGC) CALL CONTAC(3,IX,NEL1,NDF,NUMNP,DU,U,U(1,NUMNP+1),
             X         U(1,NEP+1))
003016        MR = MR - 1                                                         TS0248C
003020  800   TIME = TIME + DT                                                    TS0249C
003024  900   CONTINUE                                                            TS0250C
003027        IF(NUMPLT.GT.0) CALL PLOTGO(NDIM,NEL,DM,U)                          TS0251C
003040        CALL TICTOC (TYME,6)                                                TS0252C
003043        WRITE(ITP6,2033) TYME                                               TS0253C
003052        RETURN                                                              TS0254C
003053  901   WRITE(ITP6,2030)                                                    TS0255C
003057        IPG = 0                                                             TS0256C
003060        RETURN                                                              TS0257C
        C.... FORMATS                                                             TS0258C
003061  1000  FORMAT(F10.0,8I5,2F10.0,2I5)                                        TS0259C
003061  1006  FORMAT(3I5)                                                         TS0260C
003061  2000  FORMAT(A1,12A6,E13.5,17X,4HPAGE,I4//5X,23HTIME DEPENDENT SOLUTION   TS0261C
             X//10X,17HTIME INCREMENT  = 1PE15.4/                                 TS0262C
             X  10X,17HNUMBER OF STEPS = I5/                                      TS0263C
             X  10X,17HPRINT INTERVAL  = I5/                                      TS0264C
             X  10X,14HPRINT NODES   ,I5,3H TO,I5/                                TS0265C
             X  10X,14HPRINT ELEMENTS,I5,3H TO,I5)                                TS0266C
003061  2001  FORMAT(A1,12A6,E13.5,17X,4HPAGE,I4//5X,12HNODAL VALUES,5X,          TS0267C
             X    27HPROPORTIONAL LOAD FACTOR = ,F8.3,5X, 8HSEQUENCE,I4,5X,
             X   9HTIME STEP,I4//12H NODAL POINT,9(1X,2A6),(12X,9(1X,2A6)) )
003061  2002  FORMAT(10X,17HITERATION LIMIT =,I5/1X)                              TS0270C
003061  2005  FORMAT(4X,50H DESCRIPTION OF STRESS EVOLUTION PLOTS TO BE MADE.  /  TS0271C
             X  5X,10H PLOT NO.  5X,8H ELEMENT ,7X, 9H XYZ-CODE,6X,9H SIG-CODE/)  TS0272C
003061  2006  FORMAT(4(7X,I5,3X))                                                 TS0273C
003061  2030  FORMAT(5X,53H**FATAL ERROR 33** TIME STEF ZERO FOR DYNAMIC PROBLEM  TS0274C
             X/1X)                                                                TS0275C
003061  2033  FORMAT (    22X,12HELAPSED TIME///10X,25HINPUT PROPERTIES AND MESH  TS0276C
             ,F10.3/10X,25HCHECK AND PLOT INPUT DATA,F10.3/10X,14HFORM STIFFNESS  TS0277C
             X,F21.3/10X,21HSOLUTION OF EQUATIONS,F14.3/10X,15HOUTPUT STRESSES,   TS0278C
             XF20.3/10X,27HIMPLICIT ALGORITHM + DISPL.,F8.3/10X,19HTOTAL TIME/IM  TS0279C
             XPLICIT,F16.3/1X)                                                    TS0280C
003061  4001  FORMAT(5X,89H**WARNING  3**  BOTH THE PROPORTIONAL LOADING AND FOR  TS0281C
             XCE ARE BEING RESET ON EACH TIME STEP )                              TS0282C
003061  6000  FORMAT(5X*NCT*I5,*  UNORM*E15.5*  DNORM*E15.5*  ANORM*E15.5)        TS0242C
003061        END
```

```
               SUBROUTINE UPDATE(ISW,NDEG,NICD,U,UD,UDD,DU,F,DF,IDEST,PROP)             UPD   1C
         C***** UPDATE     ****** 12/14/73   *********************************************  UPD   2C
000016         DIMENSION U(1),DU(1),F(1),DF(1),IDEST(1),UD(1),UDD(1)                  UPD   3C
000016         COMMON/TAPES/ ITP5,ITP6                                                UPD   4C
000016         COMMON/TIMHIS/ TIME,DT,DTP,NH,ISZH,C0,C1,C2,C3,C4,C5,C6,NCT            UPD   5C
000016         GO TO (100,300,500), ISW                                               UPD   6C
         C.... UPDATE (1) * * PREUPDATE OF ACCELERATIONS FOR DYNAMIC SOLUTIONS.       UPD   7C
000024   100   DO 200 N = 1,NDEG                                                      UPD   8C
000026   200   UDD(N) = C4*UDD(N) - C5*UD(N)                                          UPD   9C
000035         RETURN                                                                 UPD  10C
         C.... UPDATE (2) * * UPDATE THE SOLUTION FOR GENL. DISPL, VEL, ACCL.         UPD  11C
000035   300   DO 400 N = 1,NDEG                                                      UPD  12C
000037         DU(N) = DF(N)                                                          UPD  13C
000043         IF(IDEST(N).EQ.0) DU(N) = F(N)*PROP - U(N)                             UPD  14C
000053         TEMP = DU(N)                                                           UPD  15C
000056         IF(NICD.EQ.1) GO TO 400                                                UPD  16C
000050         P = UDD(N)                                                             UPD  17C
000061         UD(N) = C1*UD(N) + C2*P + C3*TEMP                                      UPD  18C
000067         UDD(N) = P + C6*TEMP                                                   UPD  19C
000073   400   U(N) = U(N) + TEMP                                                     UPD  20C
000100         RETURN                                                                 UPD  21C
         C.... UPDATE(3) * * SET INTEGRATION CONSTANTS                                UPD  22C
000100   500   C6 = C6 + 0.5                                                          UPD  23C
000102         IF(C0.EQ.0.0) C0 = 0.25                                                UPD  24C
000104         WRITE(ITP6,2002) C0,C6                                                 UPD  25C
000114         C5 = 1./C0/DT                                                          UPD  26C
000116         C4 = 1. - 0.5/C0                                                       UPD  27C
000121         C3 = C6*C5                                                             UPD  28C
000123         C2 = (1. - C6/C0/2.)*DT/C4                                             UPD  29C
000127         C1 = 1. - C6/C0 + C2*C5                                                UPD  30C
000134         C6 = C5/DT                                                             UPD  31C
000137         RETURN                                                                 UPD  32C
000137   2002  FORMAT(10X,30HNEWMARK INTEGRATION PARAMETERS/                          UPD  33C
              C10X17HBETA VALUE     = F6 3/                                           UPD  34C
              C10X17HGAMMA VALUE    = F6.3)                                           UPD  35C
000137         END
```