7902978

FALSAFI-HAGHIGHI, RAHIM
INTERACTION BETWEEN STRUCTURES AND FLUIDS.

NEW JERSEY INSTITUTE OF TECHNOLOGY,
D.ENG.SC., 1978

University Microfilms International 300 N. ZEEB ROAD, ANN ARBOR, MI 48106

PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark ✓.

1. Glossy photographs ______
2. Colored illustrations ______
3. Photographs with dark background ______
4. Illustrations are poor copy ______
5. Print shows through as there is text on both sides of page ______
6. Indistinct, broken or small print on several pages ______ throughout ______
7. Tightly bound copy with print lost in spine ______
8. Computer printout pages with indistinct print ✓
9. Page(s) ______ lacking when material received, and not available from school or author ______
10. Page(s) ______ seem to be missing in numbering only as text follows ______
11. Poor carbon copy ______
12. Not original copy, several pages with blurred type ______
13. Appendix pages are poor copy ______
14. Original copy with light type ______
15. Curling and wrinkled pages ______
16. Other ______

INTERACTION BETWEEN STRUCTURES AND FLUIDS

BY

RAHIM FALSAFI-HAGHIGHI

A DISSERTATION

PRESENTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE

OF

DOCTOR OF ENGINEERING SCIENCE

IN MECHANICAL ENGINEERING

AT

NEW JERSEY INSTITUTE OF TECHNOLOGY

Newark, New Jesey
1978

ABSTRACT

A finite element model for the study of solid-fluid interaction, with applications in flow-induced vibration analysis of reactor vessel and heat exchanger internals, is presented. The model is based on the discretization of the solid equation of motion and the fluid continuity and momentum equations and employs the solid displacements together with the fluid pressure and velocity components as the nodal degrees of freedom. This permits a realistic and accurate implementation of boundary conditions, in contrast with methods using solid displacements and fluid pressure as the only field variables. The numerical solution of the resulting matrix equation, involving non-symetric matrices, is achieved by a combination of matrix decomposition, iterative scheme and analytic integration which allows the application of the elemental matrices, rather than the system global matrices, at considerable economy in computer storage and running time. Plane triangular finite elements for fluid, solid and solid-fluid continua and equivalent mass, damping and stiffness matrices and interaction load array have been developed for the study of wave propagation phenomena in a two-dimensional flow field. This is verified by solving a wave propagation flow problem consisting of water, between two elastic parallel plates, initially at rest and accelerated suddenly by applying a step pressure at one end. The results obtained are in good agreement with a previous study based on the finite element discretization of the two-dimensional wave equation. Futhermore, the results are also compared with those obtained for flow between two rigid parallel plates. These results indicate the development of water hammer phenomenon and pressure surge in the transverse direction, for the elastic wall case, which may have important implications in the design of fluid systems.

APPROVAL OF DISSERTATION

INTERACTION BETWEEN STRUCTURES AND FLUIDS

BY

RAHIM FALSAFI-HAGHIGHI

FOR

DEPARTMENT OF MECHANICAL ENGINEERING

NEW JERSEY INSTITUTE OF TECHNOLOGY

BY

FACULTY COMMITTEE

APPROVED: ________________________________,Advisor

____

____

____

____

September, 1978

ACKNOWLEDGEMENT

The author wishes to express his deep and sincere appreciation to his advisor, Dr. Amir N. Nahavandi, whose continued guidance, insight, patience and persistence made this study possible. These traits, and his willingness to assist the student has encouraged the author in the successful completion of this dissertation.

The author also wishes to express his special appreciation to Dr. Benedict C. Sun, the Doctoral Committee Chairman, who helped in bringing this dissertation to completion.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

During the recent years, dynamicists have been confronted with comlex, new problems involving the motion of structures subjected to an external and/or internal flowing fluid. These problems belong to the branch of engineering known as "flow-induced vibration." Flow-induced vibration problems have become significant as structures have become lighter and more slender due to the use of high-strength materials, and as advanced nuclear power reactors have been developed [1,2,13,14].[1] Flow-induced vibration is frequently encountered in the operation of many reactor systems and components Vibratory stresses and dynamic instabilities of nuclear fuel bundle and heat-exchanger tubes are a few examples of flow-induced vibration phenomena often observed.

The most important potential excitations are: vortex shedding, fluidelastic mechanisms, turbulent excitation and acoustic noises. Depending on conditions, any of several excitation sources can be the dominant excitation mechanism. When fluidelastic mechanism is dominant, fluid flowing through the system is a source of energy that can induce structural vibration and instability characterized by transversal motions. At small flow velocities, the structure vibrates at small amplitude which is called subcritical vibration. As the flow velocity increases, a certain value is reached at which the structure loses stability by buckling; this behavior is called instability. The forces causing this instability are affected by the deflection of the structure from its undeformed state. The essential parameters associated with fluid

---

[1] Number in brackets designate references given at the end of part two

elastic instability are system damping and fluidelastic forces. When flow velocity increases to certain value, the work done on the structure by fluidelastic forces exceeds the energy dissipated by damping. As a result, large amplitude oscillations occur. A study of fluidelastic excitation requires the consideration of both solid and fluid motions including their interactive forces and constraints and, mathematically, it poses a more difficult problem than the other flow-induced vibration sources stated earlier(i.e. vortex shedding, turbulent excitation and acoustic noises). The importance of fluidelastic excitation in such systems as nuclear reactor fuel bundles, thermal shields, and core barrels has created a need for detailed investigations of the interaction between structures and fluids. Experimental and analytical studies have been performed that confirm the importance of the interaction between structures and fluids[5,8,19,11].

The solution of interaction problems between structures and fluids for components involving realistic geometries is extremely complicated. This is due to the fact that when a structural element vibrates, the fluid surrounding and/or contained within the structure must be displaced to accomodate the motions. Rigorously speaking, motions should be studied by coupling between a structure represented by matrix equations of motion and a flow field simulated by continuity and Navier-Stokes equations including the interactive forces and constraints, as shown schematically in Fig. 1a. This approach has not yet been attempted.

A simplified approach to the solution of interaction between struc-

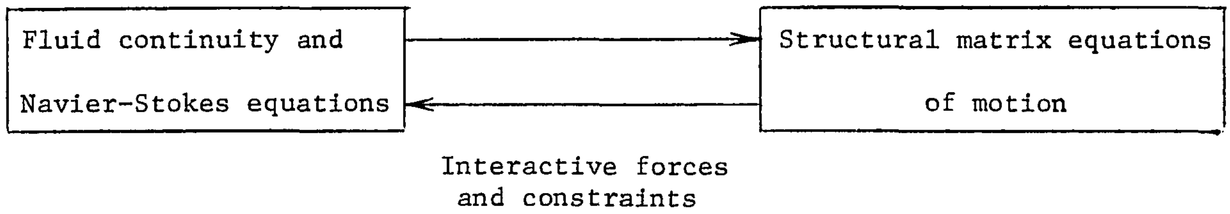


FIG. 1a RIGOROUS APPROACH TO THE SOLUTION OF INTERACTION BETWEEN STRUCRURES AND FLUIDS

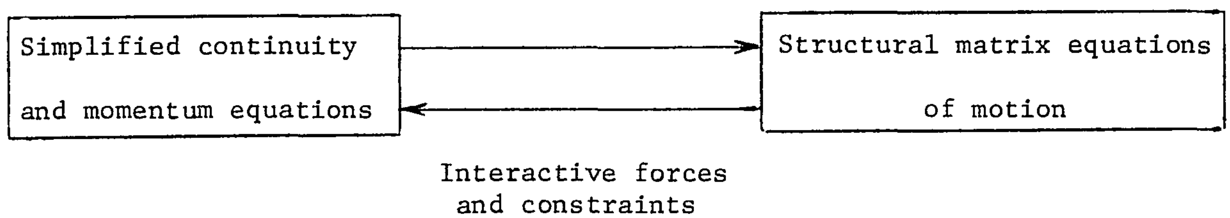


FIG. 1b THE APPROACH USED IN THIS STUDY FOR THE SOLUTION OF INTERACTION BETWEEN STRUCTURES AND FLUIDS

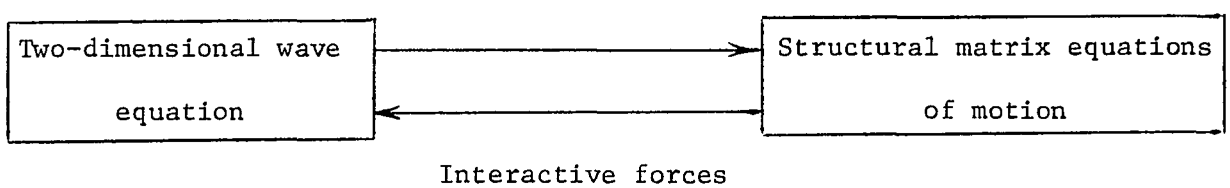


FIG. 1c SIMPLIFIED APPROACH TO THE SOLUTION OF INTERACTION BETWEEN STRUCTURES AND FLUIDS

tures and fluids shown schematically in Fig. 1c has been developed in Ref. [11]. In this approach a generalized solid-fluid interaction package has been presented which develops a fluid finite element compatible with existing structural elements and at the same time establishes the interaction between the solid and fluid elements. This study has employed the solid displacements and the fluid pressure as the only nodal degrees of freedom. The finite element models have been based on the discretization of solid matrix equations of motion and the two-dimensional wave equation including the interactive forces. The main objective of the above acousto-structural interaction is the determination of acoustic fields produced by the interaction between solid structures and the fluid contained within the structure. The mathematical formulation of acousto-structural analysis, although useful in the study of acoustic fields, cannot be readily employed in the analysis of practical solid-fluid interaction involving specified boundary conditions on the fluid velocity components. This type of problems arises in the hydraulic analysis of nuclear reactor systems in which the boundary conditions are often specified in terms of the fluid velocity components as well as pressure. Furthermore, the extension of the approach used in Fig. 1c to that of Fig. 1a is not straightforward. Therefore, to bridge the gap between the two approaches, an intermediate technique as shown schematically in Fig. 1b should be developed which uses the simplified continuity and momentum equations as a compromise between the simplified wave equation and the Navier-Stokes equations. This approach constitutes the basis for this study and uses the pressure and velocity as field variables thus avoiding the difficulty with the implementation of the boundary conditions. Expanding the number of main dependent variables in the fluid to include the fluid velocity components among the main dependent variables, boundary conditions on the velocity components as well as pressure may be easily accommodated while this is not the case for the acousto-structural analysis which can treat boundary conditions expressed in terms of pressure only.

Specifically, the purpose of the present study is to examine the problem of fluid elastic excitation by developing a mathematical model for the interaction between an elastic solid and a fluid medium using finite element approach. The objective is to employ the solid displacements and the fluid pressure and velocity components as the nodal degrees of freedom. The finite element models is based on the discretization of solid matrix equations of motion and two-dimensional continuity and simplified momentum equations including the interactive forces and constraints as shown in Fig. 1b. These simplified momentum equations permit the verification of the results with the previously mentioned solid-fluid studies based on the solid matrix equations of motion and the two-dimensional wave equation including the interactive forces shown in Fig. 1c [11], since the elimination of velocity components among the continuity and the simplified momentum equations used results in the two-dimensional wave equation.

The contributions of this study are as follows:

1) The mathematical formulation used in this study employs the solid displacements together with the fluid pressure and velocity components as the nodal degrees of freedom. This facilitates the implimentation of realistic boundary conditions in terms of pressure and velocity components, in contrast with formulations using stream function, vorticity and pressure as the only fluid field variables.

2) The mathematical formulation used in this study lays the foundation for the ultimate coupling between a structure represented by the matrix equations of motion and a flow field simulated by the continuity and Navier-Stokes equations including the interactive forces and constraints as shown in Fig. 1a.

3) Nodal velocity components time histories are the distinctive features of the present formulation since the formulation using pressure as the only fluid field variable can provide only an elemental velocity calculated from the nodal pressures.

4) The elemental matrices are used in the numerical solution. This feature results in a considerable saving in computer storage and running time, in contrast with most schemes that the assembly and storage of system global matrices are essential to numerical solution.

5) In contrast with the approach used in Fig. 1c, the approaches described in Figs.1a and 1b involve non-symmetric fluid matrices which precludes modal superposition method. Efficient numerical solution technique for the integration of these non-symmetric matrices are developed.

The approach used in this study is tested for a flow configuration consisting of water, between two elastic parallel plates, subjected to a step pressure at one end. Verification of the model is achieved by comparing the results with previous fluid elastic studies [11] based on the finite element discretization of solid matrix equation of motion and the two-dimensional wave equation including the interactive forces.

This dissertation is conveniently divided into three parts:

Part (1): This part presents the finite element model for the rigid wall case.

Part (2): This part presents the finite element model for the elastic wall case.

Part (3): Part three, user's manual of this study is provided to aid the user in understanding the computer program and to use it effectively.

PART ONE

DEVELOPMENT AND RESULTS OF FINITE ELEMENT MODEL FOR THE RIGID WALL CASE

1. INTRODUCTION

During the recent years, the finite element technique has been recognized as an effective analysis tool for the solution of a wide range of incompressible and compressible, inviscid and viscous flow problems including wave propogation phenomena [6]. Wave propogation problems are of particular interest in the loss-of-coolant accident analysis of pressurized water reactors.

Most investigators employ a stream function formulation for two-dimensional problems and a vorticity approach for three-dimensional flow fields which offer the advantage that one governing equation need be considered in the finite element discretization in a manner similar to structural analysis. The major disadvantage of stream function and vorticity approaches is the difficulty associated with the implementation of pressure, velocity, velocity gradient and stress boundary conditions. Several researchers have favored a formulation using the pressure and velocity as field variables thus avoiding the difficulty with the implementation of the boundary conditions [10]. However, this pressure-velocity formulation leads to non-symmetric matrices which precludes the application of the modal superposition to the finite element analysis [11]. The numerical solution would then require a direct numerical integration which is often hampered by excessive computer memory requirement and running time for the storage and algebraic manipulation of global matrices.

The main objective of this part is to develop an efficient technique for the finite element analysis of wave propogation problems in fluids employing the pressure-velocity formulation. This formulation reduces both the computer storage and running time requirements significantly by essentially working with the elemental matrices rather than the global matrices. Furthermore, each elemental matrix is broken into two submatrices-one diagonal matrix consisting of the terms along the diagonal and one matrix consisting of the off-diagonal coupling terms. The elemental diagonal matrices are assembled for the entire flow field, conveniently stored in a global array and kept in the left hand side of the matrix differential equations. The elemental off-diagonal matrices are stored individually and carried to the right hand side, multiplied by the respective freedom arrays and treated as time-dependent forcing functions. These elemental forcing functions are also stored in a global array. In this manner, boundary conditions on pressure and velocity as well as external forcing functions are easily introduced and the matrix equation reduces to a set of uncoupled ordinary differential equations which is readily solved by analytic integration.

The distinctive features of this efficient technique are small memory requirement, simple logic and reduced running time and computational cost. Problems requiring large high speed computers can be solved on minicomputers with limited storage and computational speed.

Specifically, this part presents a plane triangular finite element fluid model for wave propogation phenomena in a two-dimensional flow field employing the pressure-velocity formulation. This fluid element is verified numerically by solving a wave propogation flow problem consisting of water,between two flat plates,initially at rest and accelerated suddenly by applying a step pressure at one end. The results obtained are compared with a previous study[11,12] based on the finite element discretization of the two-dimensional wave equation with a single-variable formulation in which pressure is considered as the only dependent variable (referred to as the single-variable pressure formulation).

2. MATHEMATICAL FORMULATION

The simplifying assumptions employed in the development of this fluid finite element model with pressure-velocity formulation are:

1) The fluid flow is assumed to be compressible, two-dimensional and isothermal.
2) The main dependent variables for the fluid are the pressure and the two components of velocity in x and y directions.
3) The conservation laws for the fluid flow are simplified by the weak wave approximation by assuming that the density oscillations are of small magnitude.
4) The components of fluid shear stress are assumed to be proportional to their respective velocity components.
5) The momentum flux terms in the equation of motion are neglected.

The above assumptions permit the verification of the results with a finite element model based on the two-dimensional wave equation.

The differential equations governing the motion of the fluid are given by:

Continuity equation:

$$\frac{\partial \rho}{\partial t} + \rho_f \left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} \right) = 0 \tag{1}$$

Momentum equations:

$$\frac{\partial v_x}{\partial t} = - \frac{1}{\rho_f} \frac{\partial p}{\partial x} - k_f v_x \tag{2}$$

$$\frac{\partial v_y}{\partial t} = - \frac{1}{\rho_f} \frac{\partial p}{\partial y} - k_f v_y \tag{3}$$

Equation of state:

$$\left(\frac{\partial p}{\partial \rho}\right)_s = c^2 = \text{Constant} \tag{4}$$

in which ρ, p, v_x, v_y are fluid density, pressure and velocity components along x and y axis at time t, c is velocity of the acoustic waves in the fluid, ρ_f is the mean fluid density, and k_f is the viscous damping coefficient. Eliminating ρ_f between equations (1) and (4), the governing equations of motion reduces to

$$\frac{1}{c^2}\frac{\partial p}{\partial t} + \rho_f\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right) = 0 \tag{5}$$

$$\frac{\partial v_x}{\partial t} + \frac{1}{\rho_f}\frac{\partial p}{\partial x} + k_f v_x = 0 \tag{6}$$

$$\frac{\partial v_y}{\partial t} + \frac{1}{\rho_f}\frac{\partial p}{\partial y} + k_f v_y = 0 \tag{7}$$

As shown in Appendix 1, the discretization of the above equations on finite elemet subdivisions of the fluid region, shown in Fig. 2 , leads to

$$[D_e]\{\dot{Z}_e\} + [E_e]\{Z_e\} = 0 \tag{8}$$

where $[D_e]$ and $[E_e]$ are the unsymmetric fluid inertia and fluidity matrices for the element, $\{Z_e\}$ is the elemental array involving nodal degrees of freedom--pressure, x-component of velocity, and y-component of velocity as defined in Appendix 1, and $\{\dot{Z}_e\}$ is the time derivative of $\{Z_e\}$.

The pressure-velocity formulation permits the introduction of specified nodal pressure and/or velocity components into $\{Z_e\}$ and $\{\dot{Z}_e\}$

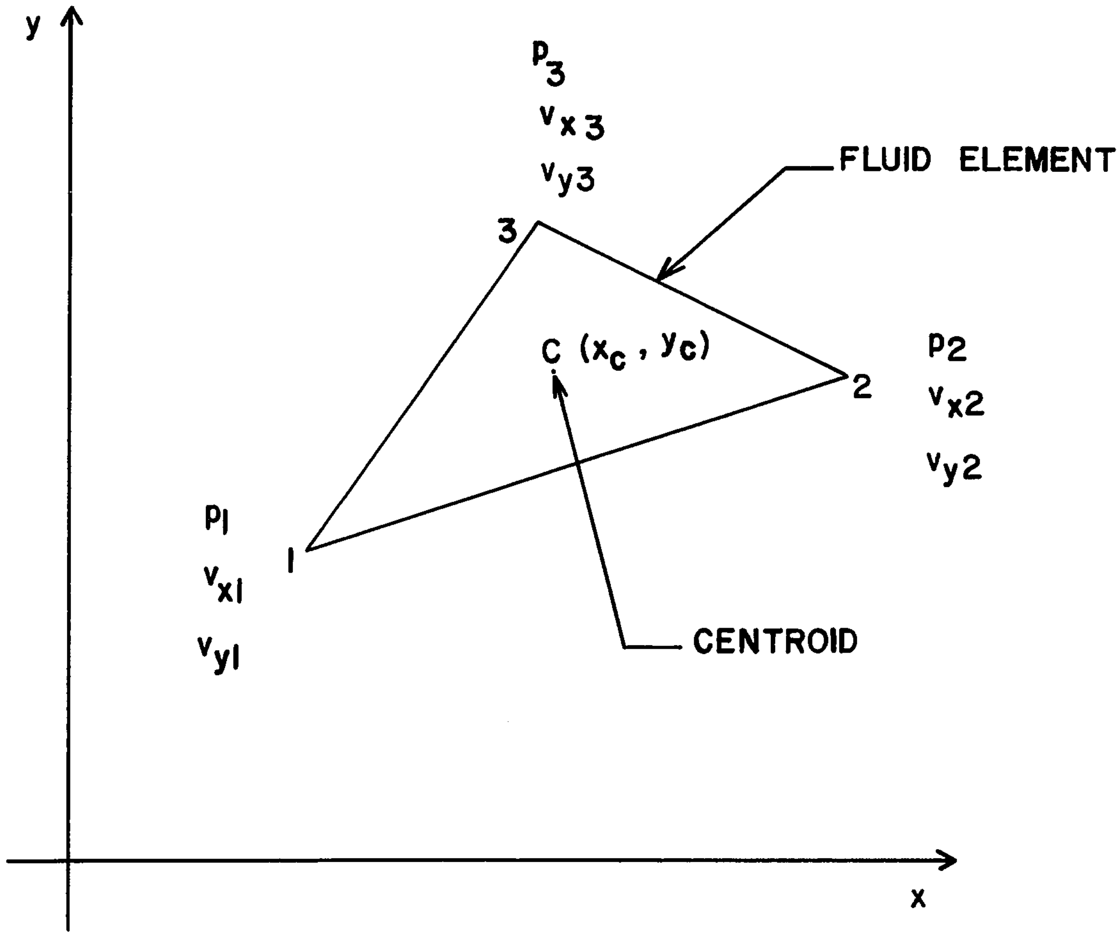


FIG. 2 PLANE TRIANGULAR FLUID FINITE ELEMENT FOR PRESSURE-VELOCITY FORMULATION

arrays. This is in contrast with the pressure formulation which is derived by the elimination of the velocity components v_x and v_y in equations (5), (6) and (7) leading to the two dimensional wave equation

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = \frac{1}{c^2} \left(\frac{\partial^2 p}{\partial t^2} + k_f \frac{\partial p}{\partial t} \right) \qquad (9)$$

The discretization of the above equation on finite element subdivisions of the fluid region will then give the matrix differential equation based on the pressure formulation as follows [12]:

$$[G_e] \{\ddot{p}_e\} + [L_e] \{\dot{p}_e\} + [H_e] \{p_e\} = \{F_e\} \qquad (10)$$

where $[G_e]$, $[L_e]$ and $[H_e]$ are the elemental inertia, viscous damping and fluidity matrices , $\{p_e\}$, $\{\dot{p}_e\}$ and $\{\ddot{p}_e\}$ are the elemental nodal p ssure arrays, its first and second time derivatives, and $\{F_e\}$ is the contribution due to the boundary integrals corresponding to the prescribed motion. The pressure formulation permits the introduction of specified pressures into $\{p_e\}$, $\{\dot{p}_e\}$ and $\{\ddot{p}_e\}$ arrays. The specified boundary conditions on velocity components can be only implemented indirectly if these conditions can be expressed in terms of nodal pressures. The derivation of the elemental matrices and the boundary integral array for equation (10), the assembly of the global matrix equation and the numerical method of solution for the pressure formulation are presented elsewhere [12].

3. NUMERICAL SOLUTION

The numerical solution for the pressure-velocity formulation is achieved by first breaking each of the fluid matrices $[D_e]$ and $[E_e]$ in equation (8) into two submatrices -- one diagonal and one off-diagonal matrix. Upon substitution, equation (8) becomes

$$\lceil D'_e \rfloor \{\dot{Z}_e\} + \lceil E'_e \rfloor \{Z_e\} = - [D''_e] \{\dot{Z}_e\} - [E''_e] \{Z_e\} \tag{11}$$

where

$$[D_e] = \lceil D'_e \rfloor + [D''_e] \tag{12}$$

$$[E_e] = \lceil E'_e \rfloor + [E''_e] \tag{13}$$

in which $\lceil D'_e \rfloor$ and $\lceil E'_e \rfloor$ are two diagonal matrices whose terms are the diagonal terms of $[D_e]$ and $[E_e]$ respectively while $[D''_e]$ and $[E''_e]$ are off-diagonal matrices with zeros along their diagonals and off-diagonal terms equal to those of $[D_e]$ and $[E_e]$ respectively. Employing temporarily the past values of $\{\dot{Z}_e\}$ and $\{Z_e\}$ for the right hand side of equations (11), these equations become completely uncoupled and the elemental matrix equation becomes

$$\lceil D'_e \rfloor \{\dot{Z}_e\} + \lceil E'_e \rfloor \{Z_e\} = \{F'_e\} \tag{14}$$

where

$$\{F'_e\} = - [D''_e] \{\dot{Z}_e\}_o - [E''_e] \{Z_e\}_o \tag{15}$$

in which the diagonal matrices $\lceil D'_e \rfloor$ and $\lceil E'_e \rfloor$ as well as the forcing function $\{F'_e\}$ can be conveniently stored in three separate arrays at significant savings in computational storage and time requirements. Assembling the above elemental equations for all n system elements, the final differential equations for the flow field are

$$\lceil D' \rfloor \{\dot{Z}\} + \lceil E' \rfloor \{Z\} = \{F'\} \tag{16}$$

where

$$\lceil D' \rfloor = \sum_{e=1}^{e=n} \lceil D'_e \rfloor \qquad \lceil E' \rfloor = \sum_{e=1}^{e=n} \lceil D'_e \rfloor \qquad \{F'\} = \sum_{e=1}^{e=n} \{F'_e\} \tag{17}$$

Equation (16) constitutes a set of 3n uncoupled first order differential equations with time-varying forcing function each having the following form

$$\sigma\dot{z} + \mu z = f(t) \tag{18}$$

in which σ and μ are constants representing the diagonal terms of $\lceil D' \rfloor$ and $\lceil E' \rfloor$ respectively. Equation (18) can be readily integrated analytically within each small time increment Δt as follows:

1) If σ and μ are nonzero,

$$z_j = z_{j-1} e^{-k\Delta t} + e^{-k\Delta t} \int_t^{t+\Delta t} e^{k\tau} f(\tau)\, d\tau \tag{19}$$

where

$$k = \frac{\mu}{\sigma}$$

and z_{j-1} and z_j are the present and the updated values of the integrated variables respectively. Since Δt is small, it would be possible to interpolate $f(\tau)$ linearly in the interval $t \leq \tau \leq t+\Delta t$ from present and updated values of "f" by

$$f(\tau) = f_{j-1} + (f_j - f_{j-1}) \frac{\tau}{\Delta t} \tag{20}$$

and perform the integration indicated by equation (19), assuming an an average value for $f = \frac{1}{2}(f_j + f_{j-1})$ within the time interval. Performing this operation one obtains

$$z_j = z_{j-1}e^{-k\Delta t} + \frac{1}{2\mu} (f_{j-1} + f_j) (1 - e^{-k\Delta t}) \tag{21}$$

and

$$\dot{z}_j = - k [z_{j-1} - \frac{1}{2\mu} (f_{j-1} + f_j) e^{-k\Delta t}] \tag{22}$$

2) If σ is nonzero but μ is equal to zero

$$z_j = z_{j-1} + \frac{\Delta t}{2\sigma} (f_{j-1} + f_j) \tag{23}$$

and

$$\dot{z}_j = \frac{1}{2\sigma} (f_{j-1} + f_j) \tag{24}$$

It should be noted that since the updated values of f_j in the above equations are not yet known, an iterative procedure would be necessary. First, the value of f_j are set equal to f_{j-1} and the first trial values z_j^1 and $\dot{z}_j^1$ are calculated from equation (21) through (24). Substituting these values into equation (15), the first updated values are computed and used in equations (21) through (24) to calculate the second trial values z_j^2 and $\dot{z}_j^2$. This iterative procedure is continued until the values of z_j converge within a prescribed error.

The combined analytical and iterative solution is repeated for successive time increments (with the origin of time at the beginning of each interval conveniently set equal to zero) until the entire problem time is covered.

4. PRESENTATION OF RESULTS

The finite element model with pressure-velocity formulation, developed in this study, is verified by comparing the results with a previous study based on a finite element model with pressure considered as the only dependent variable, obtained from the two-dimensional wave equation [11,12]. A two-dimensional channel flow, 29.4" long and 24" wide with rigid wall, shown in Figure 3, is analyzed. Water initially at rest is accelerated suddenly by applying a step pressure p_0 at the channel entrance while maintaining a zero pressure at the channel exit.

The flow region is divided into 48 triangular element as shown in Figure 4. Three values of viscous damping are studied: (1) inviscid case, $k_f = 0$; (2) sligthly viscous case, $k_f = 1406.666\ sec^{-1}$; (3) highly viscous case, $k_f = 3062.426\ sec^{-1}$.

The FASINT digital program,described in part 3,is employed to solve the problem. Response time histories are obtained numerically for all three cases indicated above. Typical nodal pressure and axial velocity component for one position upstream and one position downstream are plotted for the present study employing the multi-variable pressure-velocity formulation, a previous study using single-variable pressure formulation based on two-dimensional wave equation [11,12], and a simplified one-dimensional analytical solution. The positions and variables selected for plotting are shown in Table 1 together with the corresponding figure numbers.

To establish the convergence of the numerical solution,the minimum

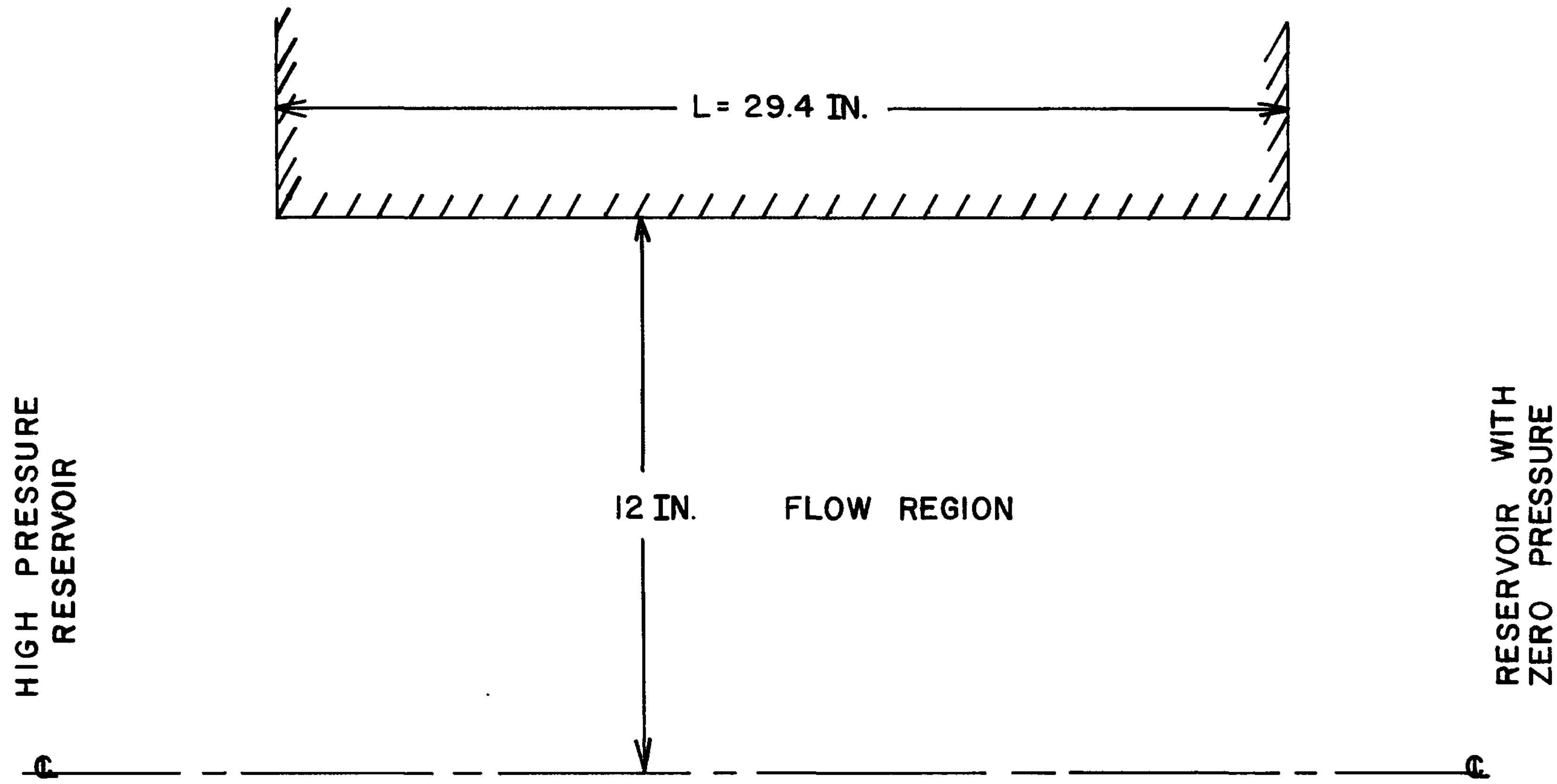


FIG. 3 TWO-DIMENSIONAL FLOW CONFIGURATION WITH RIGID WALL

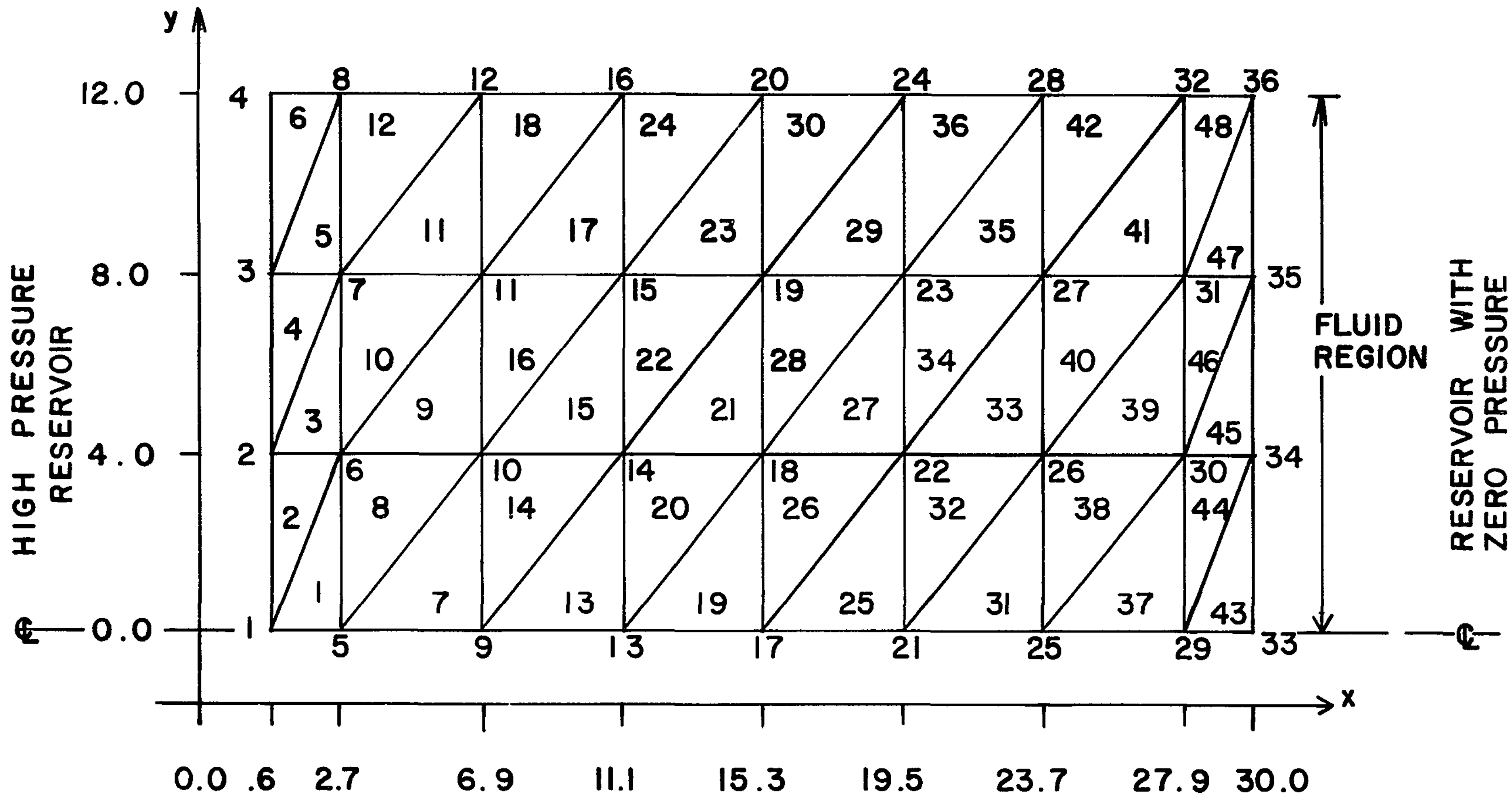


FIG. 4 48 TRIANGULAR FINITE ELEMENT FLUID MODEL FOR PRESSURE-VELOCITY FORMULATION

Table 1.

Upstream and Downstream Positions

for which Time Histories are Plotted

Upstream Node i	Position $(x_i - x_1)/L$	Variable	Figs. No.	Downstream Node i	Position $(x_i - x_1)/L$	Variable	Figs. No.
9	0.214	p	4,8,10	25	0.786	p	5,8,10
10	0.214	p	4	26	0.786	p	5
11	0.214	p	4	27	0.786	p	5
12	0.214	p	4,8,10	28	0.786	p	5,8,10
9	0.214	v_x	6,9,11	25	0.786	v_x	7
13	0.357	v_x	6,9,11	29	0.929	v_x	7
14	0.357	v_x	6,9,11	30	0.929	v_x	7

period associated with the smallest length increment is calculated from the following relation

$$T_{min} = 2\pi \frac{\ell_{min}}{c}$$

Since ℓ_{min} = 2.1 inches and c = 60,000 in/sec, T_{min} become equal to 0.00022. Examination of computer results shows that this period is indeed present in the nodal time histories on the element close to the element with the smallest length. To ascertain the convergence of the solution, the integration time step was set less than $\frac{1}{50}$ th of this minimum period at Δt = 0.0000025 sec. This time increment led to a convergent solution for all cases studied. The convergence of the solution was established when doubling the time interval produced no significant change in the pressure and velocity time histories.

The inviscid flow pressure and velocity time histories are typically shown in Figs. 5 to 8. The pressure time responses at upstream and downstream nodes, shown in Figs. 5 and6 , for both the multi-variable pressure-velocity formulation and the single-variable pressure formulation (based on the two-dimensional wave equation) oscillate about the simplified one-dimensional analytically-calculated rectangular wave form and are all in reasonable agreement. However, the simplified one-dimensional analytical solution allows only axial motion or pressure variation and, therefore, cannot be expected to be accurate for a two-dimensional flow field under consideration. The single-variable approach allows a two-dimensional pressure variation and for this reason the pressure response is oscillatory. This oscillatory behavior has also been observed by many investigators such as: 1) Conway and Jakubowski [3] who investigated wave propogation in axially impacted bars of short length

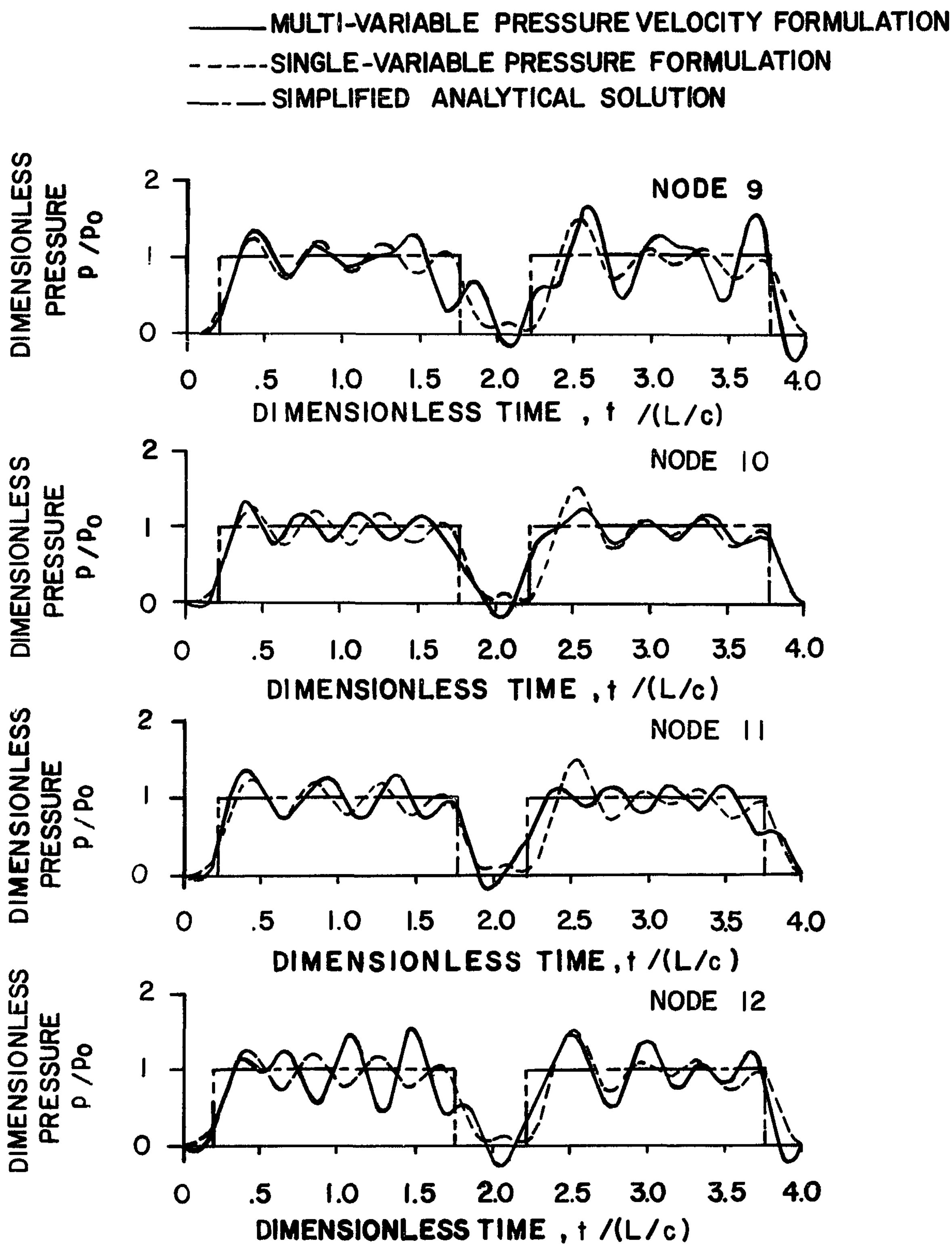


FIG. 5 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE PRESSURE TIME HISTORIES FOR INVISCID CASE AT x = 6.9 INCHES

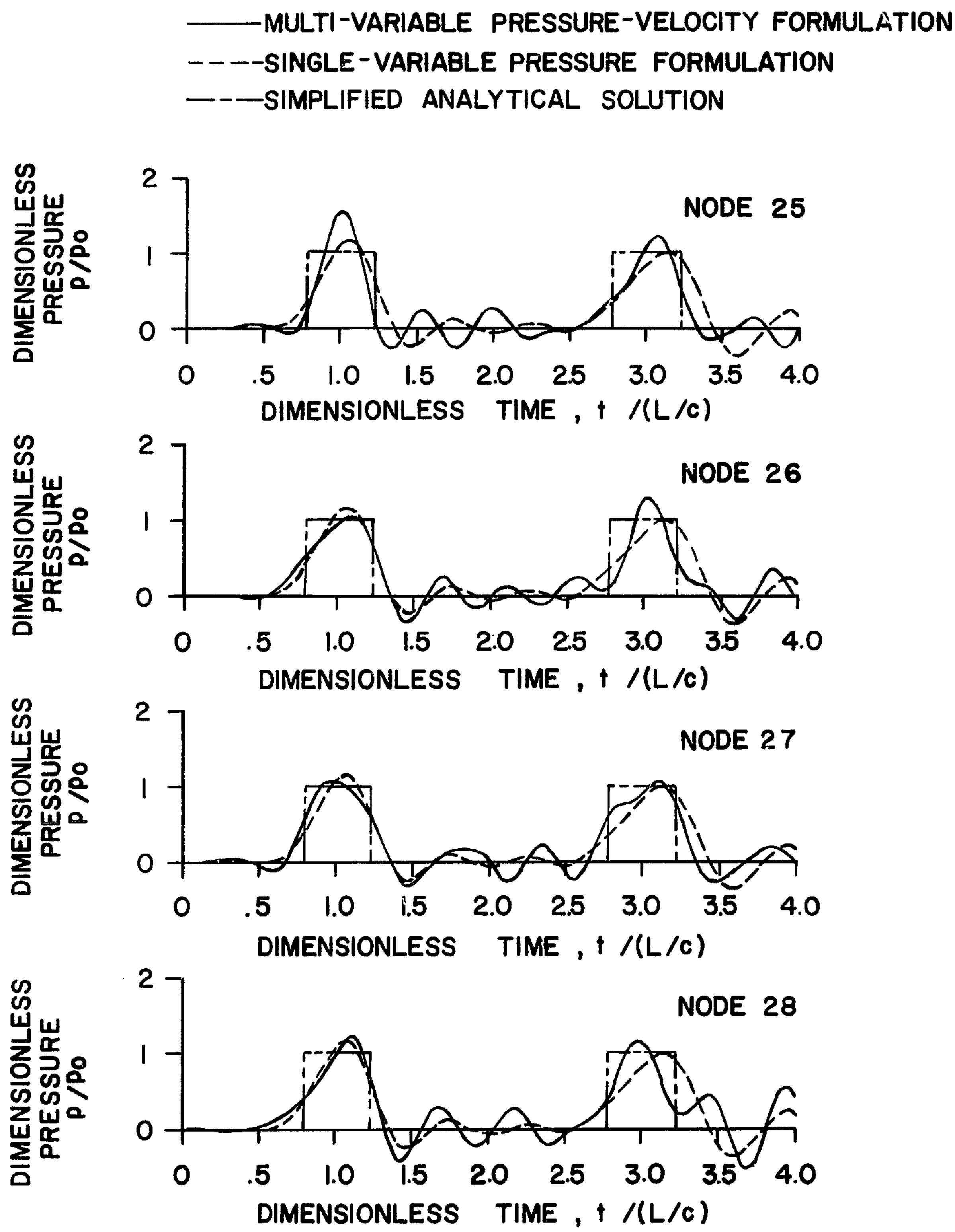


FIG. 6 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE PRESSURE TIME HISTORIES FOR INVISCID CASE AT x = 23.7 INCHES

experimentally and analytically; and 2) Zielke [22] , Holmboe and Rouleau [9], and Tarantine and Rouleau [20] who conducted experiments on the propogation of pressure pulse and flow surge in liquid transmission lines and compared the results with theory.

However, since the velocity components are not among the system degrees of freedom in the single-variable pressure formulation, the time response is not fully realistic. The multi-variable approach allows for a complete two-dimensional pressure and velocity variation and for this reason the pressure response is somewhat more oscillatory and, therefore, more realistic as compared with the single-variable pressure formulation. To demonstrate the feedback effect between the nodal pressure and the velocity components, the y velocity component for node 10 is typically plotted in Fig. 7. The y-component of velocity at node 10 oscillates about zero because of the fluid compressibilty and transversal boundary constraints. These oscillations affect the x-component of velocity at node 10 and make it more oscillatory which in turn increases the pressure-velocity vibratory response amplitudes in the entire flow field.

The velocity time response at upstream and downstream nodes, shown in Figs. 7 and 8, are the distinctive features of the pressure-velocity formulation presented herein since the single-variable pressure formulation can provide only an elemental velocity calculated from the nodal pressures. A comparison between the multi-variable and single-variable velocity time histories has been made possible by averaging the nodal velocities from the multi-variable study and comparing the result with the elemental velocity from the single-variable approach which show a good agreement.

The pressure and velocity time histories for slightly viscous case

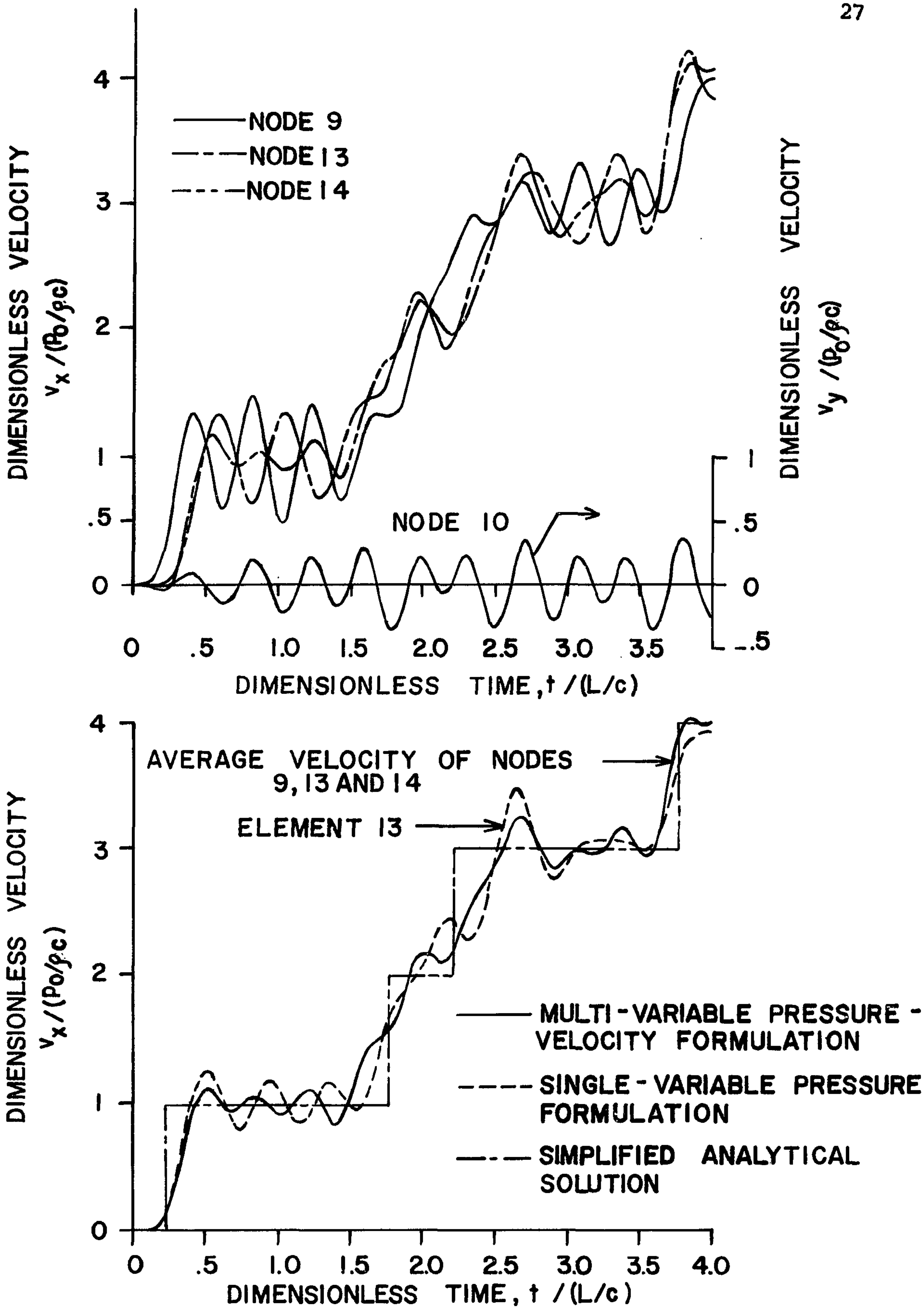


FIG. 7 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE VELOCITY TIME HISTORIES FOR INVISCID CASE FOR ELEMENT 13

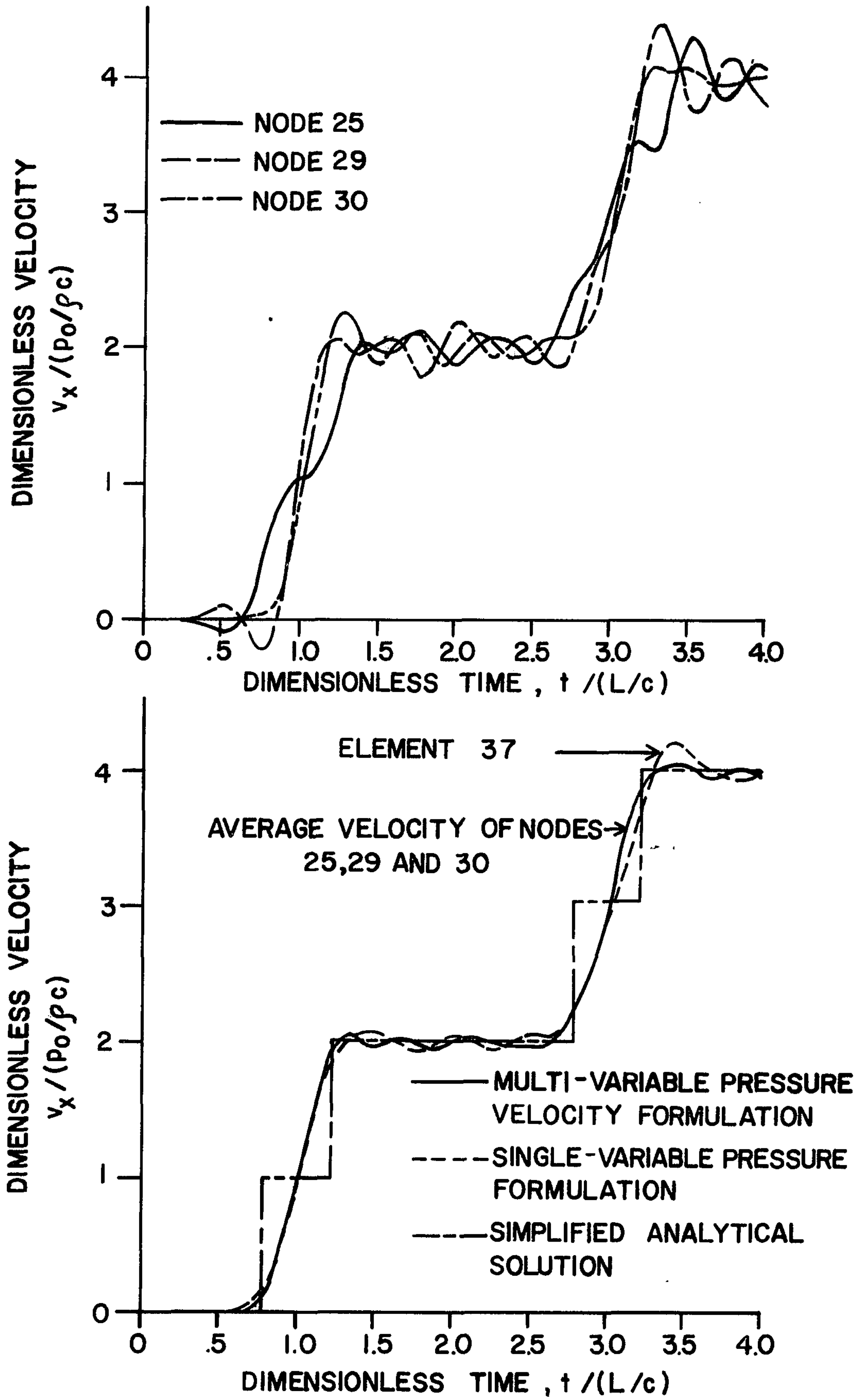


FIG. 8 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE VELOCITY TIME HISTORIES FOR INVISCID CASE FOR ELEMENT 37

typically shown in Figs. 9 and 10, exhibit a pattern similar to inviscid flow case except for the fact that all responses clearly show the viscous damping effect and indicate that ultimately a steady state pressure and velocity distribution will be reached.

The pressure and velocity time histories for highly viscous case typically shown in Figs. 11 and 12, exhibit a highly damped behavior and closely follow the single-variable and simplified analytical solutions. This is to be expected since the highly viscous condition tends to damp out the transverse pressure gradients and velocities and reduce their feedback on the longitudinal pressure and velocity distribution. All pressure and velocity responses approach their final steady state values. All physical data for the cases studied are presented in Table 2.

Table 2. Physical Data for the Rigid Wall Case

Parameter	SI Unit	British Unit
Density	999.78 N sec^2/m^4	9.3552×10^{-5} lbf-sec^2/in^4
Speed of sound	1524. m/sec	6×10^4 in/sec
Pressure at channel inlet	68948. Pa	10 psia
Channel length	0.747 m	29.4 in
Channel width	0.61 m	24.0 in
Viscous damping:		
Inviscid case	0. sec^{-1}	0. sec^{-1}
Slightly viscous case	1046.666 sec^{-1}	1046.666 sec^{-1}
Highly viscous case	3062.467 sec^{-1}	3062.467 sec^{-1}

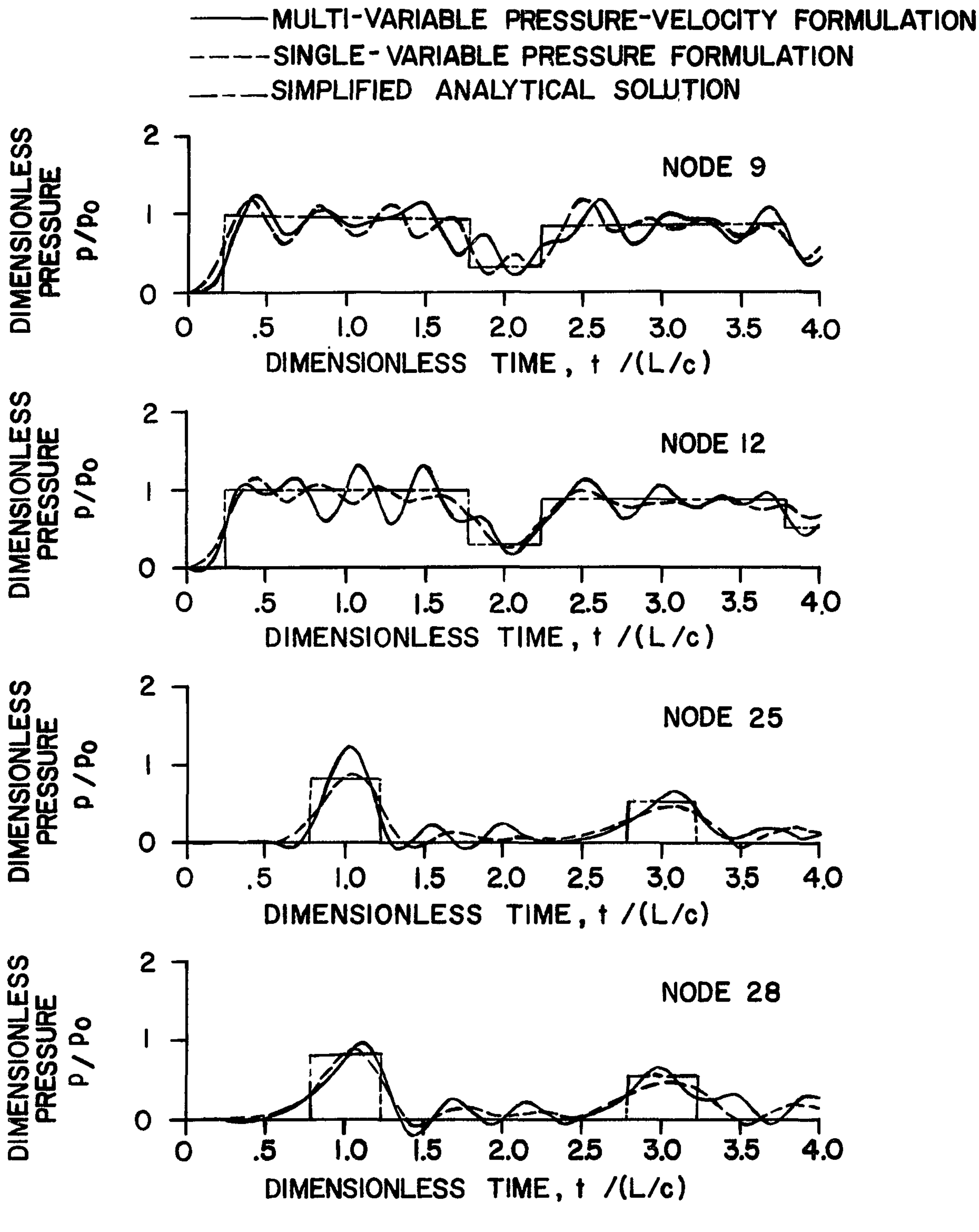


FIG. 9 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE PRESSURE TIME HISTORIES FOR SLIGHTLY VISCOUS CASE AT x = 6.9 AND x = 23.7 INCHES

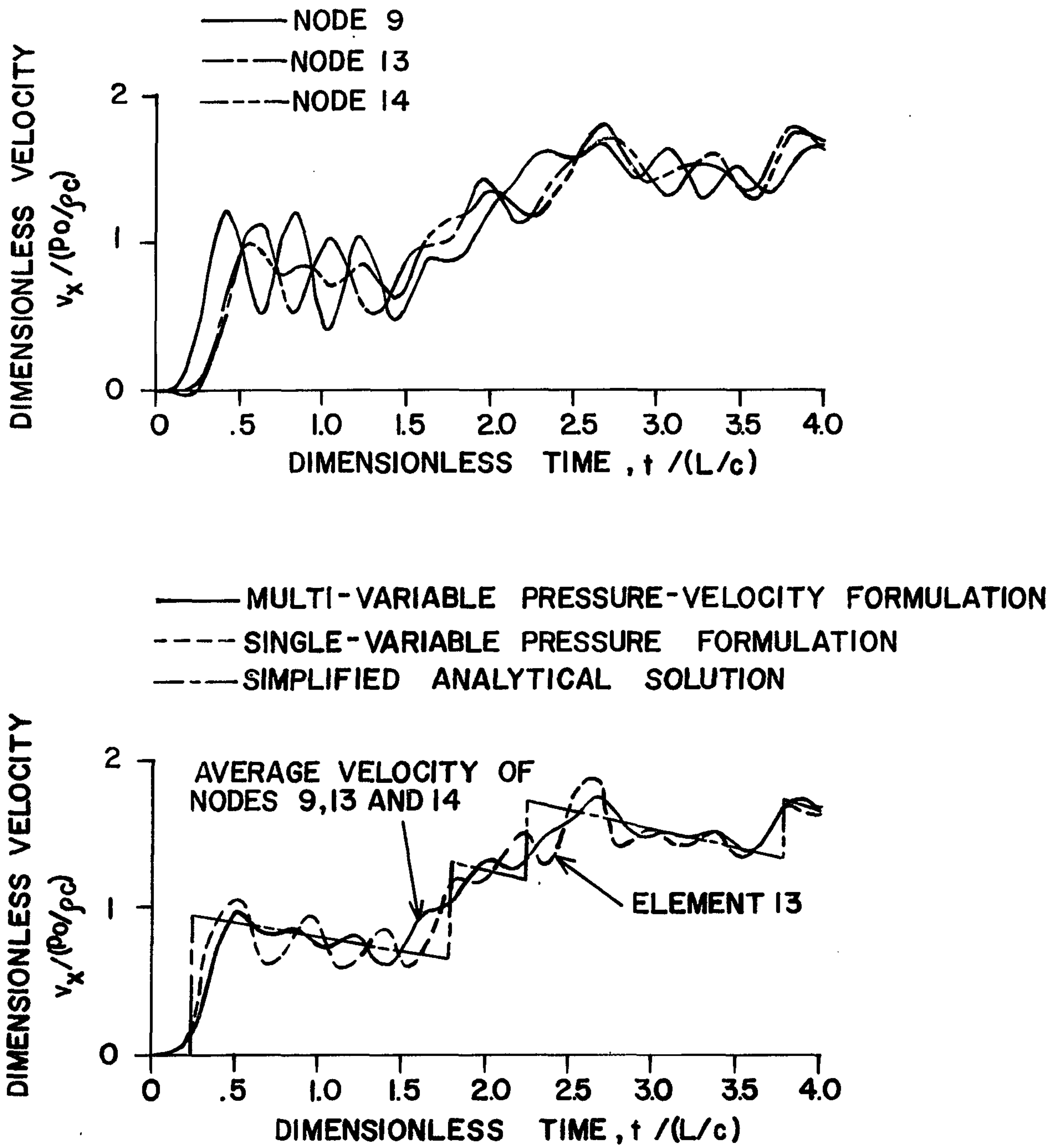


FIG. 10 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE VELOCITY TIME HISTORIES FOR SLIGHTLY VISCOUS CASE FOR ELEMENT 13

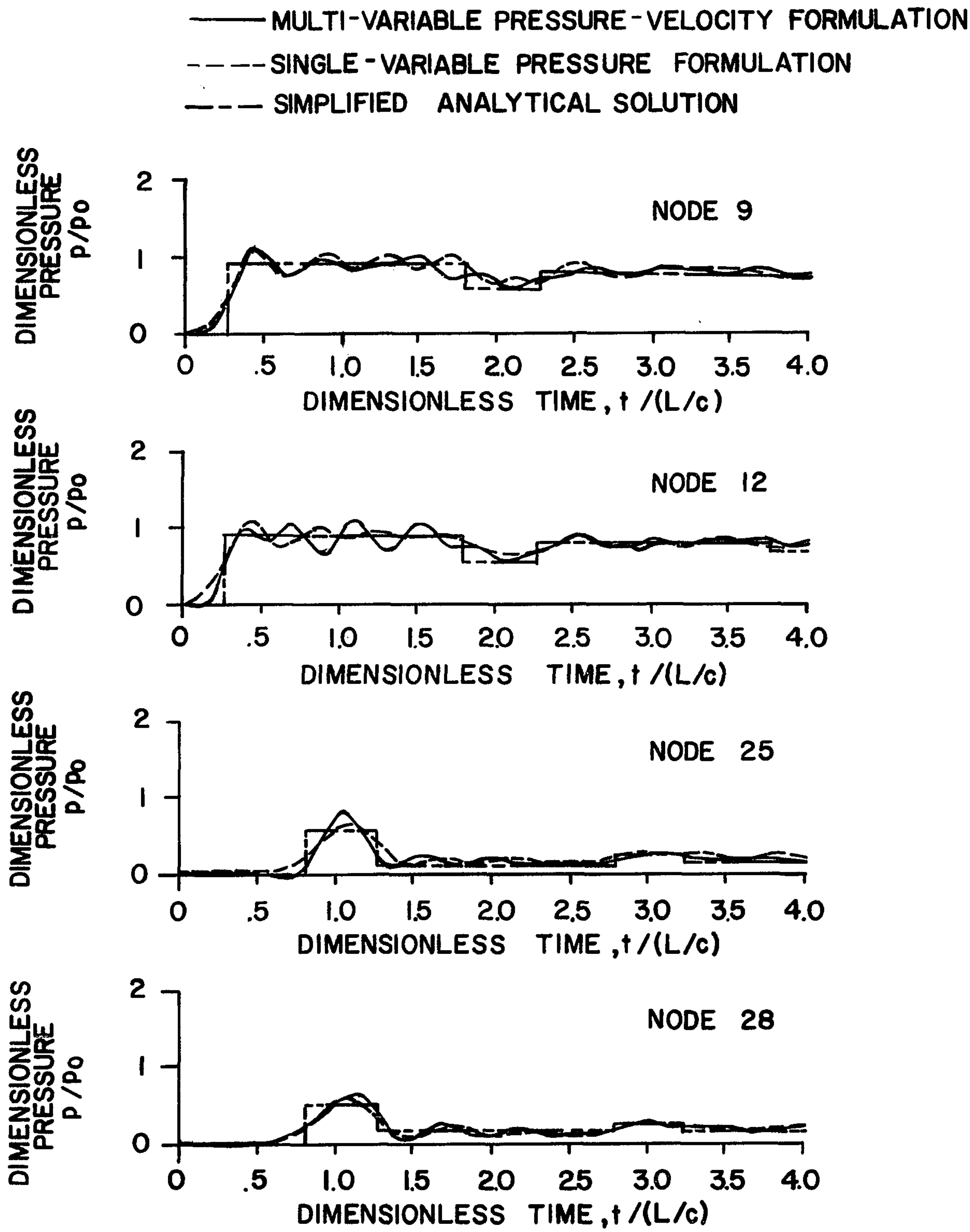


FIG. 11 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE PRESSURE TIME HISTORIES FOR HIGHLY VISCOUS CASE AT $x = 6.9$ AND $x = 23.7$ INCHES

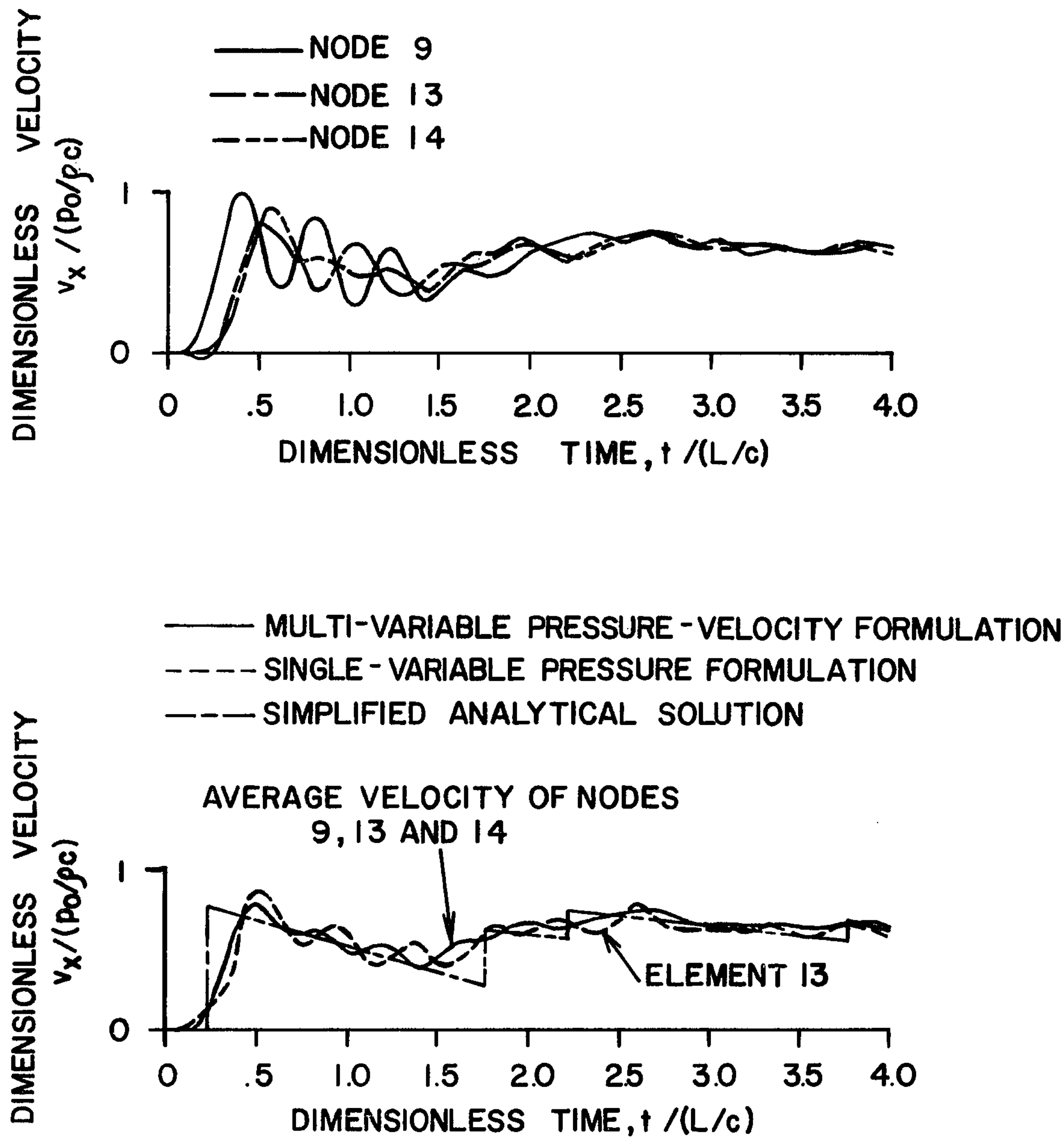


FIG. 12 A COMPARISON OF MULTI-VARIABLE AND SINGLE-VARIABLE VELOCITY TIME HISTORIES FOR HIGHLY VISCOUS CASE FOR ELEMENT 13

5. CONCLUSIONS

A multi-variable technique for the finite element analysis of wave propogation problems in fluids is presented. The main distinctive features of this technique are:

1) The mathematical formulation is based on the pressure-velocity formulation which facilitates the implementation of boundary conditions in terms of pressure and velocity components. This is in contrast with stream function, vorticity and pressure formulation for which realistic boundary conditions cannot be readily implemented.

2) The elemental matrices are used directly in the numerical solution. This feature results in a considerable saving in computer storage and running time -- in contrast with most schemes that the assembly and storage of system global matrices are essential to the numerical solution.

Based on this multi-variable pressure-velocity formulation, a plane triangular finite element fluid model for wave propogation phenomena in a two-dimensional flow field is developed. This model is verified by solving a wave propogation flow problem and comparing the results with a previous study based on a single-variable pressure formulation considering pressure as the only dependent variable. The agreement between the two models are confirmed with a parametric study using inviscid, slightly viscous and highly viscous flows.

PART TWO

DEVELOPMENT AND RESULTS OF FINITE ELEMENT MODEL FOR THE ELASTIC WALL CASE

1. INTRODUCTION

Flow induced vibration is frequently encountered in the operation of many reactor systems and components. Vibratory stresses and dynamic instabilities of reactor fuel bundle and heat exchanger tubes are a few examples of flow induced vibration phenomena which have been studied extensively both experimentally and analytically. State of the art reviews of flow induced problems have also been undertaken [17,1,2]. Four types of vibratory patterns are generally involved:

1) Turbulent excitation which is independent of the structural motion and has a broad frequency spectrum.
2) Discrete frequency excitation or vortex shedding which is also independent of structural motion and has a definite period. This phenomenon is observed in cross-flow past heat exchanger tube banks. Vortices are periodically shed from alternate sides of the tubes, causing a wavy flow pattern in the wake.
3) Acoustic oscillations normal to both flow direction and tube or rod axis which in a rectangular duct has a wave length equal to twice the duct width for its lowest frequency. This oscillation is also unrelated to the tube motion.
4) Structural natural frequency oscillations which is mechanically related to the tube or rod motion and is a function of the structural rigidity, density, geometry and support boundary conditions.

Flow induced vibration in reactor system components occurs when two or more of the above vibratory patterns coincide with one another as described in the following situations.

1) A natural frequency of the structure falls within the frequency spectrum of turbulent pressure fluctuations giving rise to resonance referred to as "buffeting".
2) The frequency of vortex shedding at a particular flow rate coincides with the natural acoustic frequency of the component. The two systems thus couple, the kinetic energy in the flow stream is converted into acoustic pressure waves.
3) A structural natural frequency may be near the vortex shedding frequency. The tube or rod will then start to vibrate and reinforce the vortex shedding causing severe noise and structural fatigue failures. This is referred to as forced vibration.
4) The position of the structure with respect to the incident stream may cause a transverse fluid force on the structure producing instability and self-excited vibration characterized by continual lateral displacements. The forces causing this instability are affected by the deflection of the structure from its undeformed state. This is referred to as fluid elastic excitation, flutter, or galloping. A study of fluid elastic excitations requires the consideration of both solid and fluid motions including their interactive forces and constraints. Mathematically, this poses a more difficult problem than the other flow-induced vibration patterns enumerated above.

The purpose of this study is to examine the problem of fluid elastic excitation by developing a mathematical model for interaction between an elastic solid and a fluid medium using a finite element approach. A previous study of these problems employed the solid displacements and the fluid pressure as the only nodal degrees of freedom[11] in which the fluid finite element was based on the discretization of the two-dimensional wave equation. This fluid element, referred to as single-variable pressure formulation does not provide a convenient means for the implementation of boundary conditions involving velocity components. The objective of this study is to employ the solid displacement and the fluid pressure and velocity components as the nodal degrees of freedom. In this approach the fluid finite element will be based on the discreization of the two-dimensional continuity and momentum equations.

The distinctive features of this study are that it provides a means for the implementation of both pressure and velocity boundary conditions and lays the foundation for the ultimate coupling between a structure represented by the matrix equations of motion and a flow field simulated by the continuity and Navier-Stokes equations.

Specifically, this part presents a plane triangular finite element solid-fluid model for wave propagation phenomena in a two-dimensional flow field employing the matrix pressure-velocity formulation for the fluid and the matrix displacement formulation for the solid. This solid-fluid model is verified by solving a wave propagation problem consisting of water, between two elastic parallel plates, initially at rest and accelerated suddenly by applying a step pressure at one end. The results obtained are compared with previous fluid elastic studies [11] based on the fluid finite element discretization of the two-dimensional wave equation with a single-variable pressure formulation as well as with previous wave propogation studies between two rigid parallel plates described in part one.

2. MATHEMATICAL FORMULATION

The simplifying assumptions employed in this study are as follows:

1) The solid continuum is assumed to be perfectly elastic and the fluid flow is compressible and isothermal.
2) The solid and fluid continua are considered to be two dimensional. The main dependent variables of the solid are two components of displacement in x and y directions and those of the fluid are the pressure and the two components of velocity in x and y directions.
3) The solid continuum undergoes small in-plane deformations while the fluid continuum experiences density changes of small magnitude. This allows the simplification of the flow momentum equations by the weak wave approximation.
4) Structural damping is considered by introducing damping matrices proportional to stiffness and/or mass matrices. Fluid friction is taken into account by assuming that the components of the fluid shear stress is proportional to their respective velocity components.
5) The momentum flux terms in the equations of motion is neglected.

The above assumptions permit the verification of the results with previous solid-fluid interaction studies based on a two-dimensional wave equation[11]. The finite element model is presented in the following sections entitled fluid element, solid element, and solid-fluid super-element and detailed further in Appendices 1 through 4.

Fluid Element

The fluid finite element, used in this study, is developed by employing the method of weighted residuals. The two-dimensional momentum and continuity equations in terms of the fluid pressure and velocity components are discretized on finite element subdivisions of the fluid region, as shown in the Appendices 1 and 2,leading to the following matrix differential equation for each fluid element shown in Fig.2.

$$[D_e] \{\dot{Z}_e\} + [E_e] \{Z_e\} = 0 \tag{25}$$

where $[D_e]$ and $[E_e]$ are the unsymmetric fluid inertia and fluidity matrices for the element, $\{Z_e\}$ is the elemental array involving nodal degrees of freedom (pressure, x-component of velocity, and y-component of velocity), and $\{\dot{Z}_e\}$ is the time derivative of $\{Z_e\}$.

It should be noted that the pressure-velocity formulation, employed above, permits the introduction of nodal pressure and velocity boundary values into $\{Z_e\}$ and $\{\dot{Z}_e\}$ arrays. This is in contrast with the pressure formulation, based on the discretization of the two-dimensional wave equation, expressed by [11]

$$[G_e] \{\ddot{p}_e\} + [L_e] \{\dot{p}_e\} + [H_e] \{p_e\} = \{F_e\} \tag{26}$$

in which the velocity boundary conditions can be implemented indirectly only if they can be expressed in terms of pressure.

Solid Element

The solid finite element is obtained by applying the principle of virtual work to the solid [23, 16, 7]. This procedure yields the matrix differential equation for the nodal displacements of the solid element shown in Fig. 13.

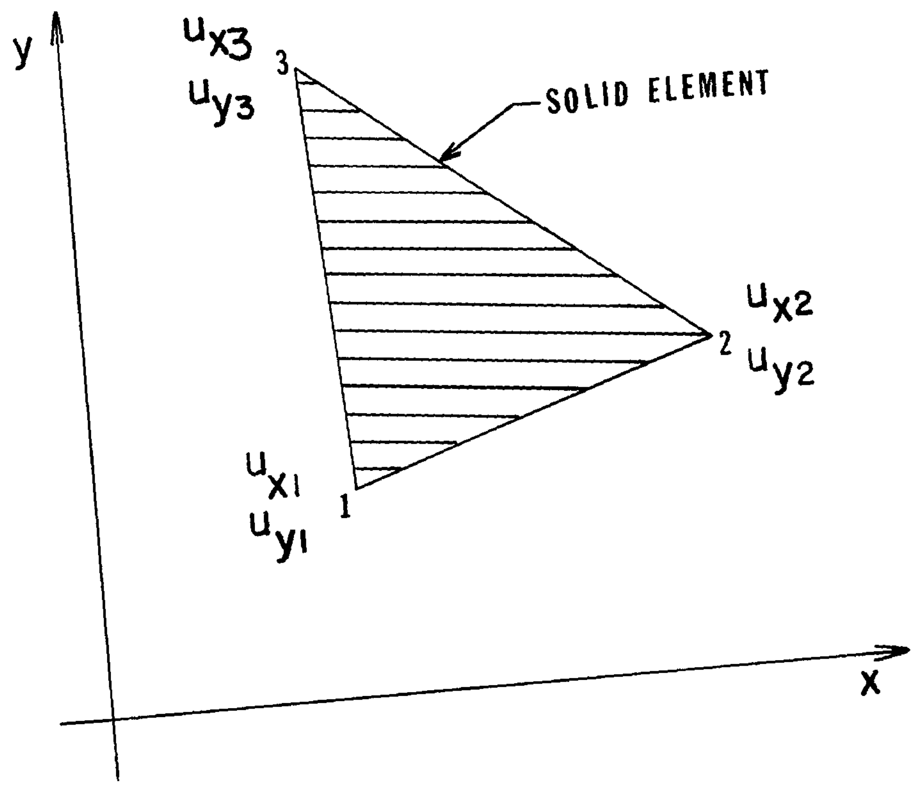


FIG. 13 PLANE TRIANGULAR SOLID FINITE ELEMENT

$$[M_e]\{\ddot{U}_e\} + [C_e]\{\dot{U}_e\} + [K_e]\{U_e\} = \{R_e\} + \{R'_e\} \tag{27}$$

This formulation permits the application of boundary conditions that can be represented by specified nodal displacements.

Solid-Fluid Superelement

The solid-fluid superelement, shown in Fig. 14, is constructed by combining the solid and fluid element while at the same time including the interaction between the two elements. For the solid part, the interactive term is the pressure force acting normal to the moving solid boundary. For the fluid part, the interaction is expressed by making the fluid nodal velocity components equal to the solid velocity components. The matrix differential equation for the solid-fluid superelement, shown in Fig. 14, becomes

$$\begin{bmatrix} M_e & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{U}_e \\ \ddot{Z}_e \end{Bmatrix} + \begin{bmatrix} C_e & 0 \\ 0 & D_e \end{bmatrix} \begin{Bmatrix} \dot{U}_e \\ \dot{Z}_e \end{Bmatrix} + \begin{bmatrix} K_e & 0 \\ 0 & E_e \end{bmatrix} \begin{Bmatrix} U_e \\ Z_e \end{Bmatrix} = \begin{Bmatrix} [S_e]\{p_e\} \\ 0 \end{Bmatrix} + \begin{Bmatrix} R'_e \\ 0 \end{Bmatrix} \tag{28}$$

together with a constraint that for nodal points on the solid-fluid boundary

$$\{v_{ye}\} = \{\dot{u}_{ye}\} \tag{29}$$

The boundary conditions applicable to equation (28) are those indicated for the solid and fluid parts.

The elemental matrices for the fluid element and the solid-fluid superelement are non-symmetric. This precludes the application of the

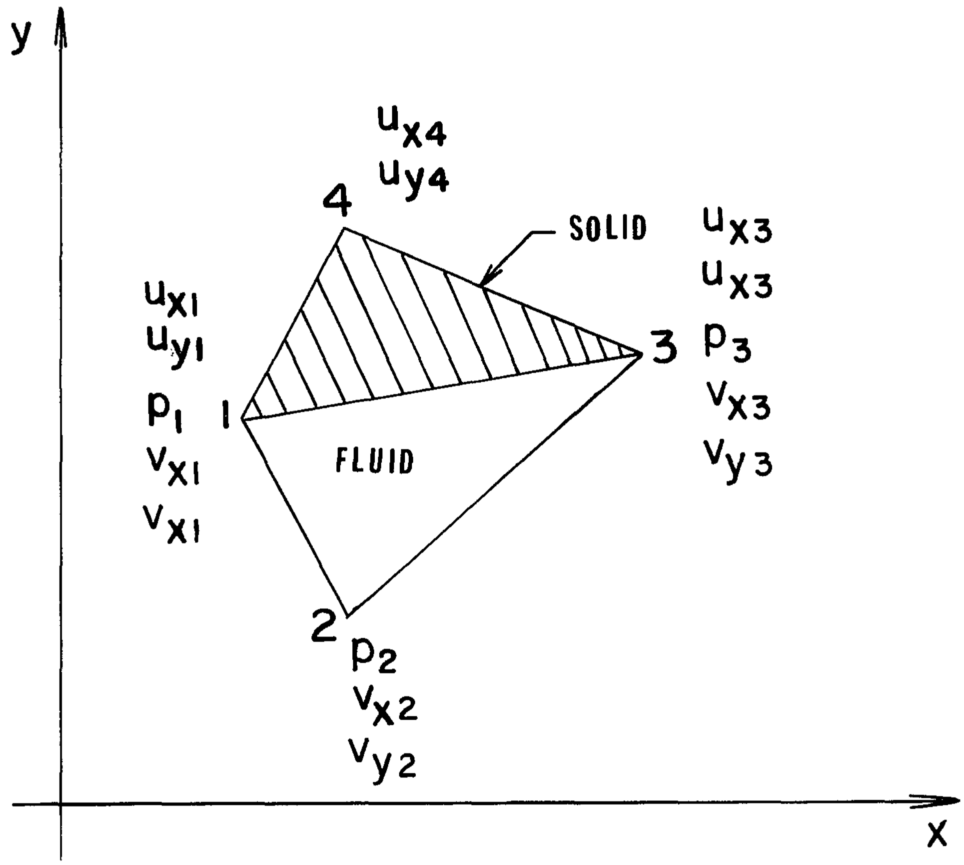


FIG. 14 PLANE QUADRILATERAL SOLID-FLUID SUPERELEMENT

modal superposition technique to the numerical solution of the problem. A direct numerical integration would require excessive computer memory and running time for the storage and algebraic manipulation of the global matrices. For these reasons, the numerical solution is obtained by a combination of matrix decomposition, analytical integration and iterative scheme discussed in the next section.

3. NUMERICAL SOLUTION

The elemental matrix equation, equations (25), (27), and (28) for the fluid, solid and solid-fluid elements are all in the following general form

$$[\bar{M}_e]\ \{\ddot{X}_e\} + [\bar{C}_e]\ \{\dot{X}_e\} + [\bar{K}_e]\ \{X_e\} = \{F_e\} \tag{30}$$

For the fluid element and the fluid part of the solid-fluid superelement, the contribution of $[\bar{M}_e]$ is zero as indicated by equations (25) and (28) respectively. The numerical solution is achieved by breaking each of the elemental matrices $[\bar{M}_e]$, $[\bar{C}_e]$ and $[\bar{K}_e]$ into two submatrices -- one diagonal matrix and one off-diagonal matrix such that equation (30) can be expressed by

$$\lceil \bar{M}'_e \rfloor\ \{\ddot{X}_e\} + \lceil \bar{C}'_e \rfloor\ \{\dot{X}_e\} + \lceil \bar{K}'_e \rfloor\ \{X_e\} = \{F_e\} - [\bar{M}''_e]\ \{\ddot{X}_e\} - [\bar{C}''_e]\ \{\dot{X}_e\} - [\bar{K}''_e]\ \{X_e\} \tag{31}$$

in which $\lceil \bar{M}'_e \rfloor$, $\lceil \bar{C}'_e \rfloor$ and $\lceil \bar{K}'_e \rfloor$ are three diagonal matrices whose terms are the diagonal terms of $[\bar{M}_e]$, $[\bar{C}_e]$ and $[\bar{K}_e]$ respectively, while $[\bar{M}''_e]$, $[\bar{C}''_e]$ and $[\bar{K}''_e]$ are off-diagonal matrices with zeros along their diagonal and off-diagonal terms equal to those of $[\bar{M}_e]$, $[\bar{C}_e]$ and $[\bar{K}_e]$ respectively. Furthermore, $\{F_e\}$ represent the array of interactive forces exerted by the fluid on the solid in addition to the external forces acting on the solid, if any. It should be noted that, in contrast with the single-variable pressure formulation [11] where the interaction of solid on the fluid appears as inertial forces, in the multi-variable pressure-velocity formulation equation (29) expresses the constraint imposed by the solid on the fluid. Employing temporarily the past values of $\{\ddot{X}_e\}$,

$\{\dot{X}_e\}$ and $\{X_e\}$ for the right hand of equation(31), this equation becomes completely uncoupled and the elemental matrix equation becomes

$$\lceil \bar{M}'_e \rfloor \{\ddot{X}_e\} + \lceil \bar{C}'_e \rfloor \{\dot{X}_e\} + \lceil \bar{K}'_e \rfloor \{X_e\} = \{F'_e\} \tag{32}$$

where

$$\{F'_e\} = \{F_e\} - [\bar{M}''_e] \{\ddot{X}_e\}_o - [\bar{C}''_e] \{\dot{X}_e\}_o - [\bar{K}''_e] \{X_e\}_o \tag{33}$$

in which the diagonal matrices $\lceil \bar{M}'_e \rfloor$, $\lceil \bar{C}'_e \rfloor$ and $\lceil \bar{K}'_e \rfloor$ as well as the forcing function $\{F'_e\}$ can be conveniantly stored in four separate arrays. This feature results in significant saving in computational storage and time requirement.

Assembling the above elemental equations for all n system elements, the global matrix differential equations for the discretized solid-fluid continua is obtained

$$\lceil \bar{M}' \rfloor \{\ddot{X}\} + \lceil \bar{C}' \rfloor \{\dot{X}\} + \lceil \bar{K}' \rfloor \{X\} = \{\bar{F}'\} \tag{34}$$

where

$$\lceil \bar{M}' \rfloor = \sum_{e=1}^{e=n} \lceil \bar{M}'_e \rfloor$$

$$\lceil \bar{C}' \rfloor = \sum_{e=1}^{e=n} \lceil \bar{C}'_e \rfloor$$

$$\lceil \bar{K}' \rfloor = \sum_{e=1}^{e=n} \lceil \bar{K}'_e \rfloor$$

$$\{\bar{F}'\} = \sum_{e=1}^{e=n} \{F'_e\} \tag{35}$$

Equation (34) constitutes a set of uncoupled second order differential equations with time varying forcing function each having the following form

$$\lambda \ddot{x} + \sigma \dot{x} + \mu x = f(t) \tag{36}$$

in which λ , σ and μ are constants representing the diagonal terms of $\lceil \bar{M}' \rfloor$, $\lceil \bar{C}' \rfloor$ and $\lceil \bar{K}' \rfloor$ respectively. Equation (36) can be readily integrated within each small time increment Δt by assuming a linear variation of $f(t)$ given by

$$\bar{f}(t) = \frac{1}{2} (f_j + f_{j-1}) \tag{37}$$

in which f_{j-1} refers to the value of $f(t)$ at time t and f_j refers to the value of $f(t)$ at time $t+\Delta t$ to be determined by an iterative procedure as will be described later.

Case (1) : Values of λ , σ and μ are nonzero. For this case which occurs for solid nodes, equation (36) may be written as follows:

$$\ddot{x} + 2 \xi \omega_n \dot{x} + \omega_n x = \frac{\bar{f}}{\lambda} \tag{38}$$

where

$$\omega_n = \sqrt{\frac{\mu}{\lambda}} \qquad \xi = \frac{1}{2\omega_n} \frac{\sigma}{\lambda} \tag{39}$$

If $\xi < 1$, the undamped solution is given by

$$x_j = e^{- \xi \omega_n \Delta t} (A_1 \cos \omega_d \Delta t + A_2 \sin \omega_d \Delta t) + \frac{\bar{f}}{\mu} \tag{40}$$

where

$$A_1 = x_{j-1} - \frac{\bar{f}}{\mu}$$

$$A_2 = \frac{1}{\omega_d} [\dot{x}_{j-1} + \xi \omega_n (x_{j-1} - \frac{\bar{f}}{\mu})]$$

$$\omega_d = \omega_n \sqrt{1 - \xi^2} \tag{41}$$

If $\xi = 1$, the critically damped solution is given by

$$x_j = (A_1 + A_2' \Delta t) e^{- \omega_n \Delta t} + \frac{\bar{f}}{\mu} \tag{42}$$

where

$$A_2' = \dot{x}_{j-1} + \omega_n (x_{j-1} - \frac{\bar{f}}{\mu}) \tag{43}$$

If $\xi > 1$, the overdamped solution is given by

$$x_j = B_1 e^{r_1 \Delta t} + B_2 e^{r_2 \Delta t} + \frac{\bar{f}}{\mu} \tag{44}$$

where

$$r_1 = \omega_n (- \xi + \sqrt{\xi^2 - 1})$$

$$r_2 = \omega_n (- \xi - \sqrt{\xi^2 - 1})$$

$$B_1 = - \frac{\dot{x}_{j-1} - r_2 [x_{j-1} - (\bar{f}/\mu)]}{r_2 - r_1}$$

$$B_2 = \frac{\dot{x}_{j-1} - r_1 [x_{j-1} - (\bar{f}/\mu)]}{r_2 - r_1} \tag{45}$$

Case (2):Values of σ and μ are nonzero but $\lambda = 0$. For this case which occurs for fluid nodes with viscous effect, equation (36) may be integrated:

$$x_j = x_{j-1} e^{-(\mu/\sigma) \Delta t} + \frac{\bar{f}}{\mu} [1 - e^{-(\mu/\sigma) \Delta t}] \tag{46}$$

Case (3): Values of σ and μ are zero but $\lambda \neq 0$. For this case which occurs for fluid nodes without viscous effects, the integration of equation (36) gives

$$x_j = x_{j-1} + \frac{\bar{f}}{\sigma} \Delta t \tag{47}$$

The first and second time derivatives of x_j needed for the calculation of $\bar{f}$, can be readily computed by analytically differentiating the above expressions as necessary. It should be further noted that since the updated value of f_j in $\bar{f}$ used in the above equation are not yet known, an iterative procedure would be necessary. First, the values of f_j are set equal to f_{j-1} and the first trial values x_j^1, $\dot{x}_j^1$ and $\ddot{x}_j^1$ are calculated from equation (40), (42), (44), (46) and (47), as the case may be. Substituting these values into equation (33), the first updated values are computed and used in equation (37) and in expressions for x_j to calculate the second trial values x_j^2, $\dot{x}_j^2$ and $\ddot{x}_j^2$. This iterative procedure is continued until the values of x_j converge within a prescribed error.

The above combined analytical and iterative solution is repeated for successive time increments (with the origin of time at the beginning of each time interval conveniantly set equal to zero) until the entire problem time is covered.

4. PRESENTATION OF RESULTS

The finite element model developed in this study (designated as multi - variable pressure-velocity formulation) is verified for a two-dimensional channel flow with elastic walls shown in Fig. 15. The results obtained are compared with a previous study based on a finite element model with pressure considered as the only dependent variable in the fluid region (referred to as the single - variable pressure formulation [11]. The results are also compared with wave propagation studies between two rigid parallel plates described in part one. Water initially at rest is accelerated suddenly by applying a step pressure, p_0 at the left end while maintaining a zero pressure at the right end. A 72 element grid model, shown in Fig. 16 , is used in this analysis. Elements 6, 15, 24, 33, 42, 51, 60 and 69 are modeled using the solid-fluid quadrilateral superelements. Elements located below and above these elements are modeled employing fluid and solid triangular finite elements respectively. Two values of fluid viscous damping are studied: 1) Inviscid case, k_f=0., and 2) Highly viscous case, k_f=3062.467 sec^{-1}.

The FASINT digital program,described in part 3,is employed to solve the problem. Response time histories are obtained numerically for the cases indicated above. Typical nodal pressure and velocity components for the fluid and the nodal displacement components for the solid for one position upstream and one position downstream are plotted for the present study employing the multi-variable pressure-velocity formulation and compared with two previous studies: 1) single-variable

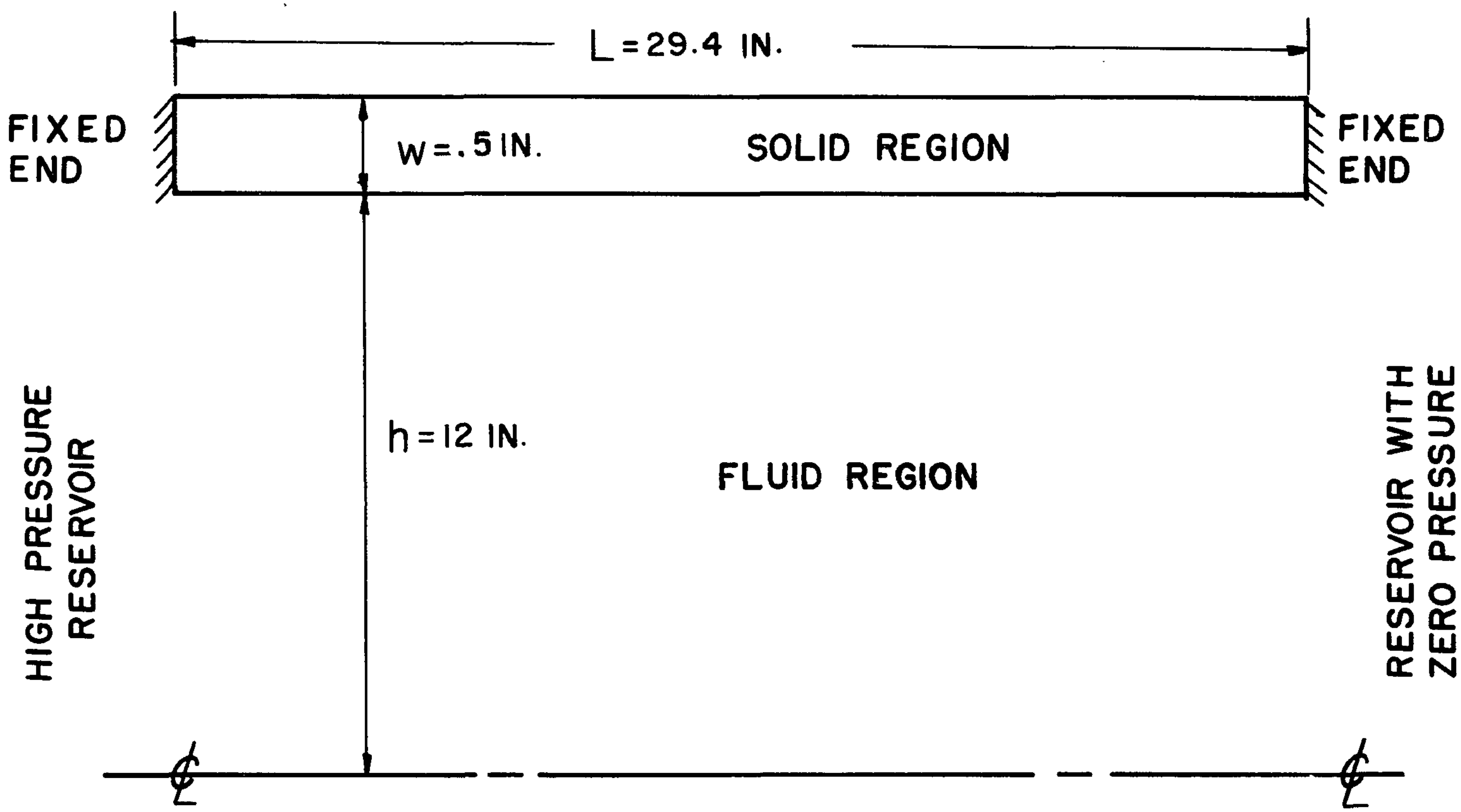


FIG. 15 FLOW CONFIGURATION

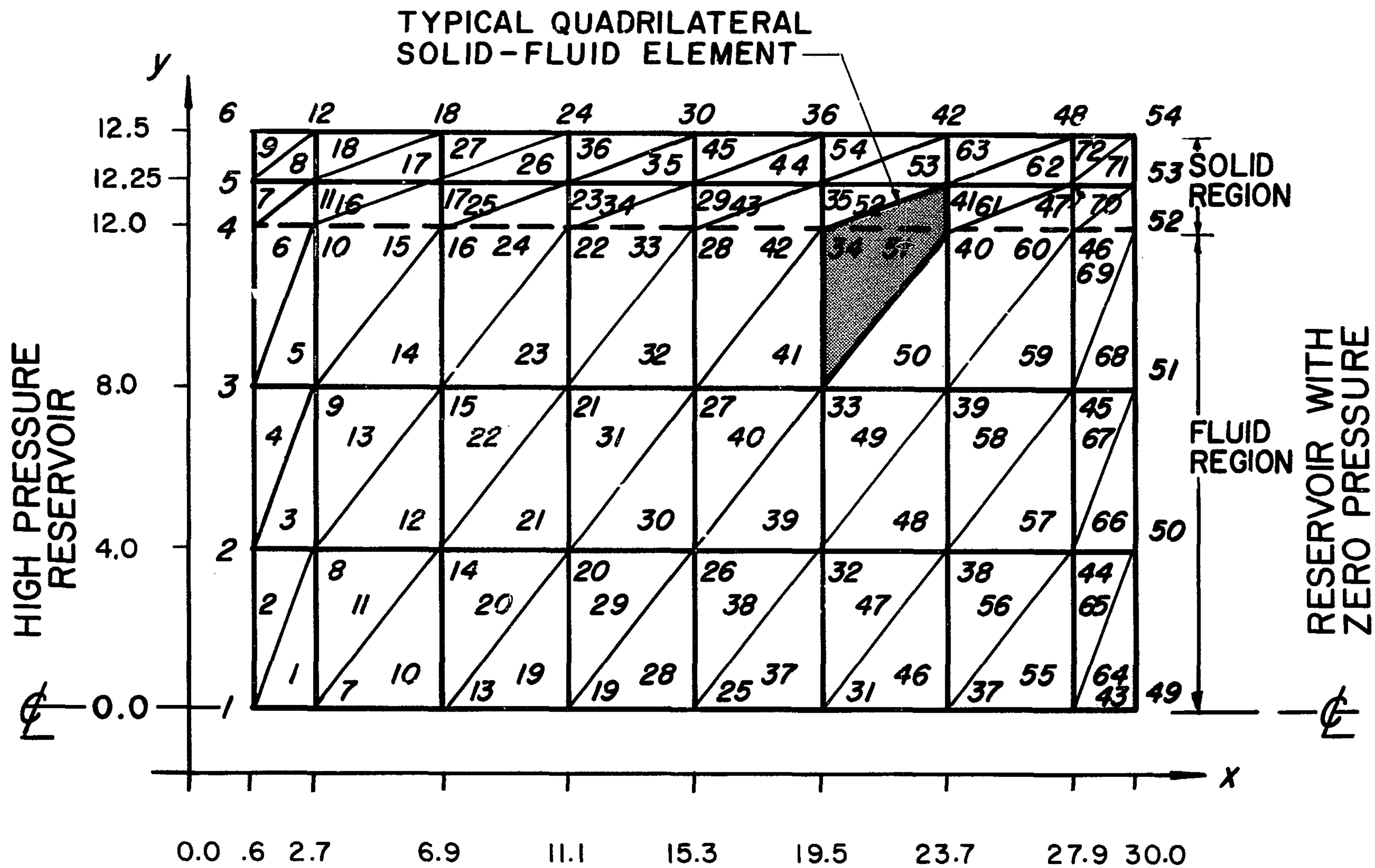


FIG. 16 72 ELEMENT SOLID-FLUID MODEL

pressure-formulation based on two-dimensional wave propagation between elastic walls (see Figs. 17 and 18) ; and 2) multi-variable pressure-velocity formulation for two-dimensional wave propagation between rigid walls and simplified one-dimensional analytical solution for channel flow with rigid walls (see Figs. 19 through 25). The positions and variables selected for plotting are marked on each figure and the physical data are summarized in Table 3 .

To establish the convergenceof the numerical solution,the minimum period associated with the smallest length increment is calculated from

$$T_{min} = 2\pi \frac{\ell_{min}}{c}$$

Since the speed of sound in the solid is much larger than that of the fluid and the solid length increments are much smaller than that of the fluid, the minimum period will correspond to the solid element with the smallest length. For ℓ_{min} = 0.25 inches and $c=2.0276 \times 10^{5}$ in/sec, T_{min} becomes equal to 7.75×10^{-6} sec.Examination of the solid displacements (not shown) indicates that this period is indeed present in the nodal time histories on the elements close to the solid element with the smallest length . To ascertain the convergence of the solution, the integration time step was set less than $\frac{1}{50}$th of this minimum period at $t=1.5 \times 10^{-7}$ seconds. This time increment led to a convergent solution for all cases studied. The convergence of the solution was established when doubling the time interval produced no significant change in the model time histories.

Figures 17 and 18 show a comparison of the multi-variable

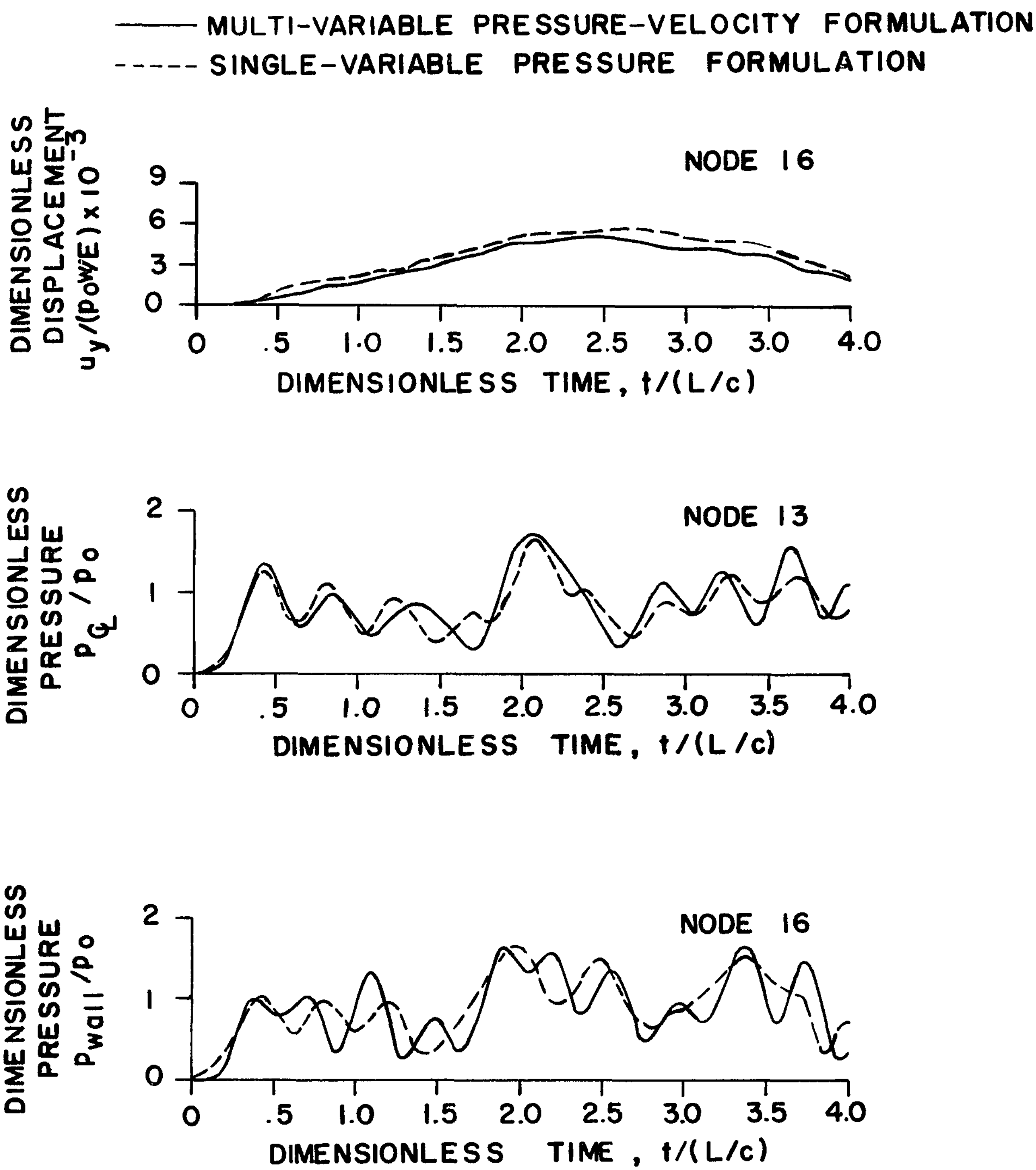


FIG. 17 A COMPARISON OF MULTI- AND SINGLE-VARIABLE PRESSURE AND DISPLACEMENT TIME HISTORIES FOR ELASTIC WALL CASE WITH INVISCID FLOW AT (x = 6.9 IN.)

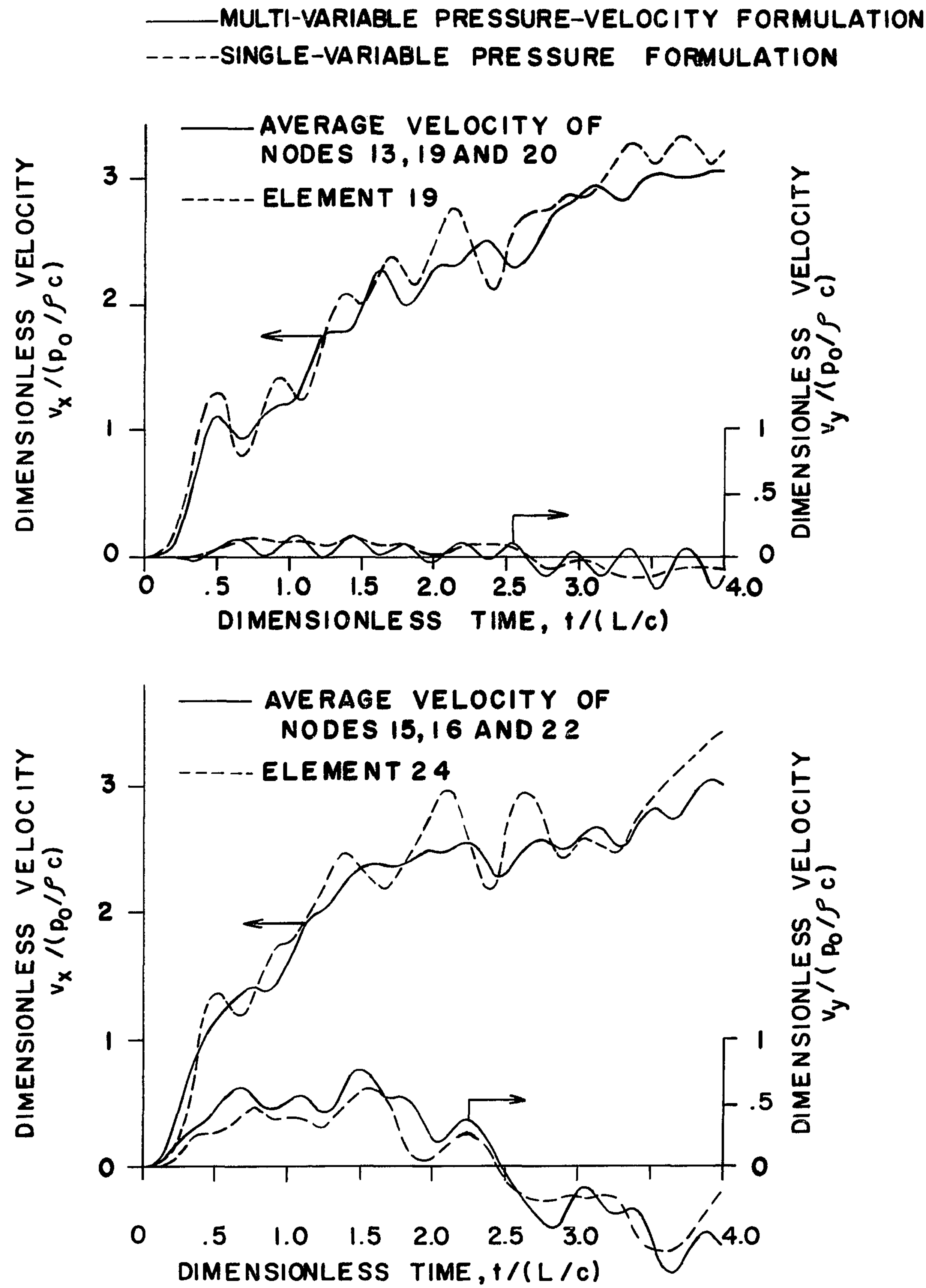


FIG. 18 A COMPARISON OF MULTI- AND SINGLE- VARIABLE FLUID VELOCITY TIME HISTORIES FOR ELASTIC WALL CASE WITH INVISCID FLOW (AT x = 6.9 IN.)

Table 3. Physical Data for the Elastic Wall Case

Parameters	British Unit	SI Unit
Fluid density	9.35521×10^{-5} lbf-sec^2/in^4	999.78 N sec^2/m^4
Solid density	7.297×10^{-4} lbf-sec^2/in^4	7798.2 N sec^2/m^4
Speed of sound in fluid	6.0×10^4 in/sec	1524.0 m/sec
Speed of sound in solid	2.0276×10^5 in/sec	5150.1 m/sec
Solid Young Modulus	30.0×10^6 psi	20.68×10^7 kPa
Solid Poisson Ratio	0.3	0.3
Pressure at channel inlet	10 psia	68948.0 Pa
Channel length, L	29.4 in	0.747 m
Channel height, h	24.0 in	0.610 m
Wall thickness, w	0.5 in	0.0127 m
Fluid damping coefficient,k:		
Inviscid	0. sec^{-1}	0. sec^{-1}
Highly viscous	3062.467 sec^{-1}	3062.467 sec^{-1}

pressure-velocity formulation, employed in this study, with single-variable pressure-formulation from reference[11] based on the two-dimensional wave equation. The two solutions are generally in reasonable agreement. However, since the velocity components are not among the system degrees of freedom in the single-variable pressure formulation, the associated time responses cannot be expected to be as accurate as the present multi-variable pressure - velocity formulation.

It should be remembered that the velocity time responses, shown in Fig.18, are the distinctive features of the multi-variable pressure-velocity formulation presented herein since the single-variable pressure formulation can provide only an elemental velocity calculated from the nodal pressures. A comparison between the multi-variable and single-variable velocity time histories has been made possible by averaging the nodal velocities from the multi-variable study and comparing the result with the elemental velocity from the single-variable approach which show a good agreement. This verification provides the confidence needed to initiate parametric studies and to present a detailed description of the system behavior as follows.

The solid vertical displacement, fluid pressure, and axial and transversal fluid velocity time histories at the centerline of the channel and at the wall are presented at one upstream and one downstream location for inviscid and highly viscous cases in Figs. 19 through 25 . These curves are representative of the response of the solid and the fluid, obtained in this study, at other locations along the channel. Superimposed on these curves are two rigid wall (no solid-fluid interaction) curves for comparison. The first is a finite element solution for the wave propagation problem with rigid walls using multi-variable pressure-velocity formulation described in part one. The second is the analytic solution to the one-dimensional wave equation for the rigid wall problem[12].

In Fig. 19 , the pressure time history for the multi-variable pressure-velocity formulation in the rigid wall case oscillate about the simplified one-dimensional analytically-calculated rectangular wave form. However, the simplified one-dimensional analytical solution allows only axial motion or pressure variation and, therefore, cannot be expected to be accurate for a two-dimensional flow field under consideration. The multi-variable pressure-velocity formulation with rigid walls allows a two-dimensional pressure and velocity variation and for this reason, the pressure response is more oscillatory. This oscillatory behavior has been observed by other investigators both experimentally and analytically [3,22,9,20]. For the elastic wall case, the pressure surge in the channel results in a gradual deflection of the wall leading to a reduction of the axial pressure surge which is initially more pronounced near the wall than at the center. This situation creates a transverse flow, as shown in Fig. 20 , until the wall deflection reaches a maximum and no longer permits a transverse flow leading to a transversal water hammer and pressure surge about $t^{L}/c = 2$ which is distinctly opposed to the zero pressure dip for the rigid wall case. The transversal pressure surge occurs first at the wall and then travels backward to the centerline. Following the transverse surge phenomenon,the elastic energy stored in the wall forces the wall to move back toward its initial position and reverses the transverse flow direction with a subsequent gradual pressure rise toward the end of the cycle. The axial flow, for the elastic

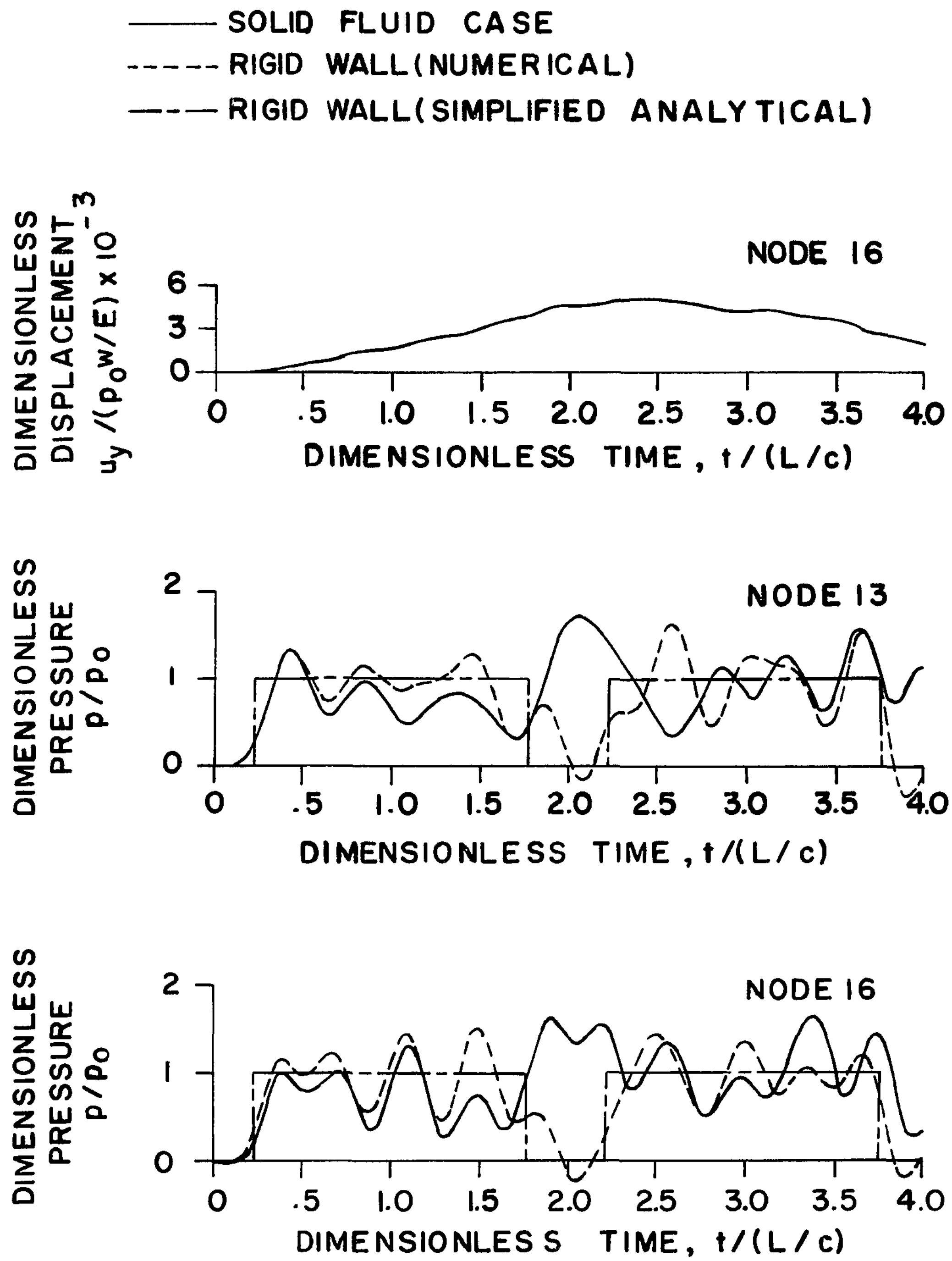


FIG. 19 A COMPARISON OF PRESSURE AND DISPLACEMENT TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH INVISCID FLOW (AT x = 6.9 IN.)

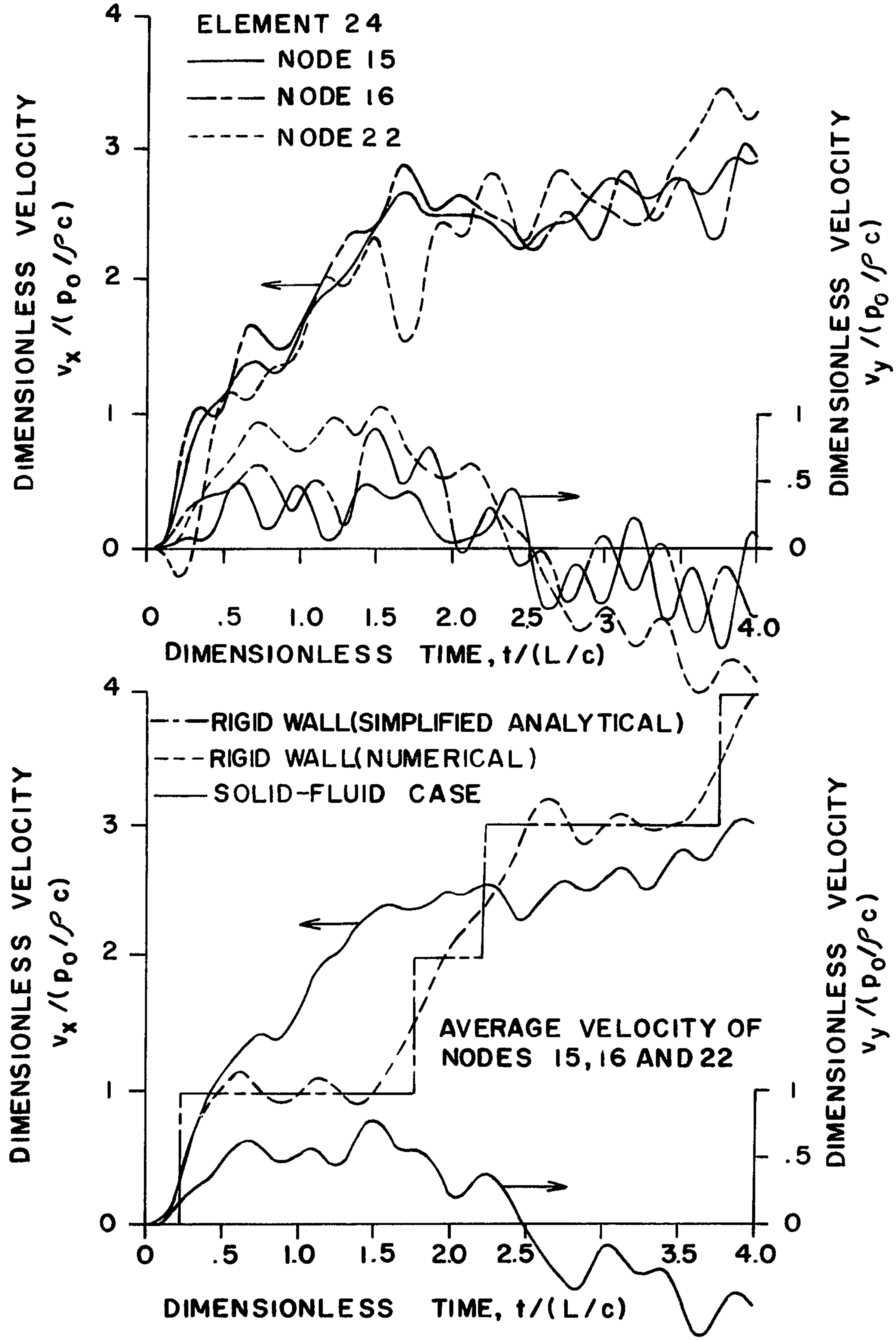


FIG. 20 A COMPARISON OF FLUID VELOCITY TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH INVISCID FLOW (AT x = 6.9 IN. NEAR THE WALL)

wall case, is larger than that of the rigid wall as the wall deflects outward and is smaller as the wall moves inward. This behavior is more pronounced near the wall (element 24 in Fig. 20) than near the channel centerline (element 19 in Fig. 21). The same characteristics are observable at the downstream position shown in Fig. 22 and 23 .

The effect of damping is shown in Figs. 24 and 25 for the highly viscous flow conditions. In comparison with the inviscid flow case, the pressure, velocity and displacement amplitudes in the highly viscous flow case are lower and the curves are smoother as expected. The transverse surge is not as large and the solution tends toward a steady-state value corresponding to the rigid wall case because of the highly viscous flow conditions.

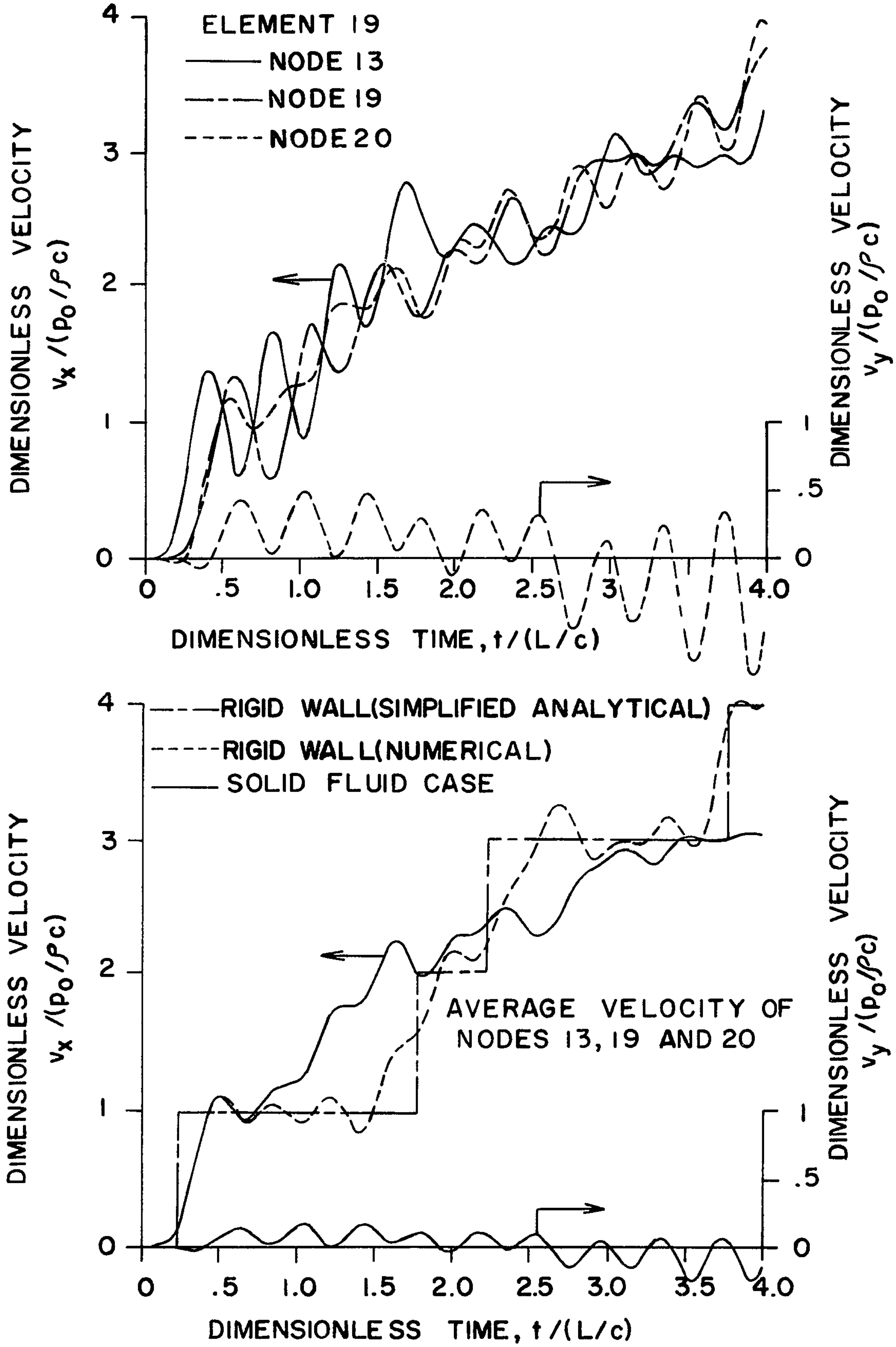


FIG. 21 A COMPARISON OF FLUID VELOCITY TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH INVISCID FLOW (AT x = 6.9 IN. NEAR THE CHANNEL CENTERLINE)

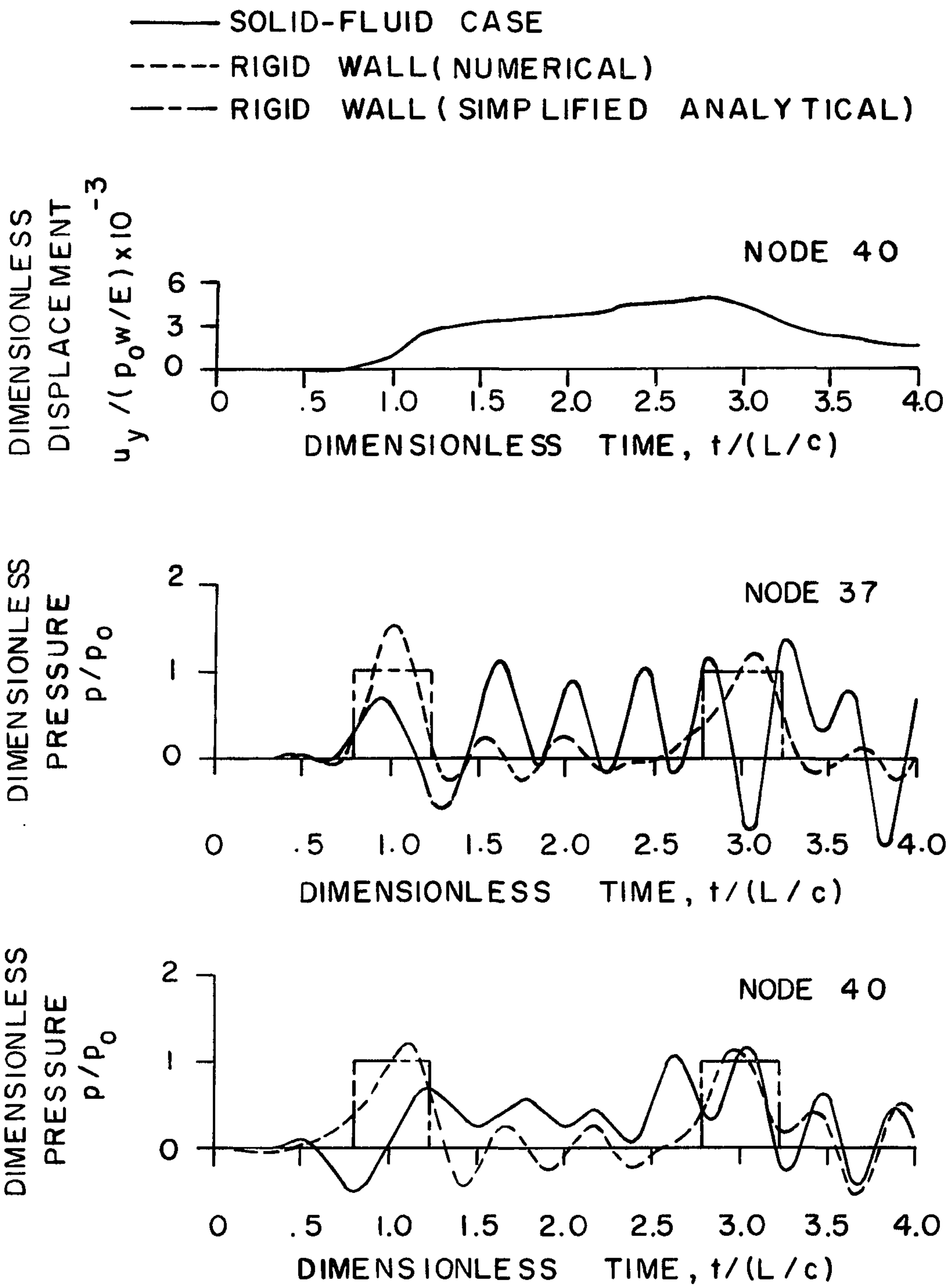


FIG. 22 A COMPARISON OF PRESSURE AND DISPLACEMENT TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH INVISCID FLOW (AT x = 23.7 IN.)

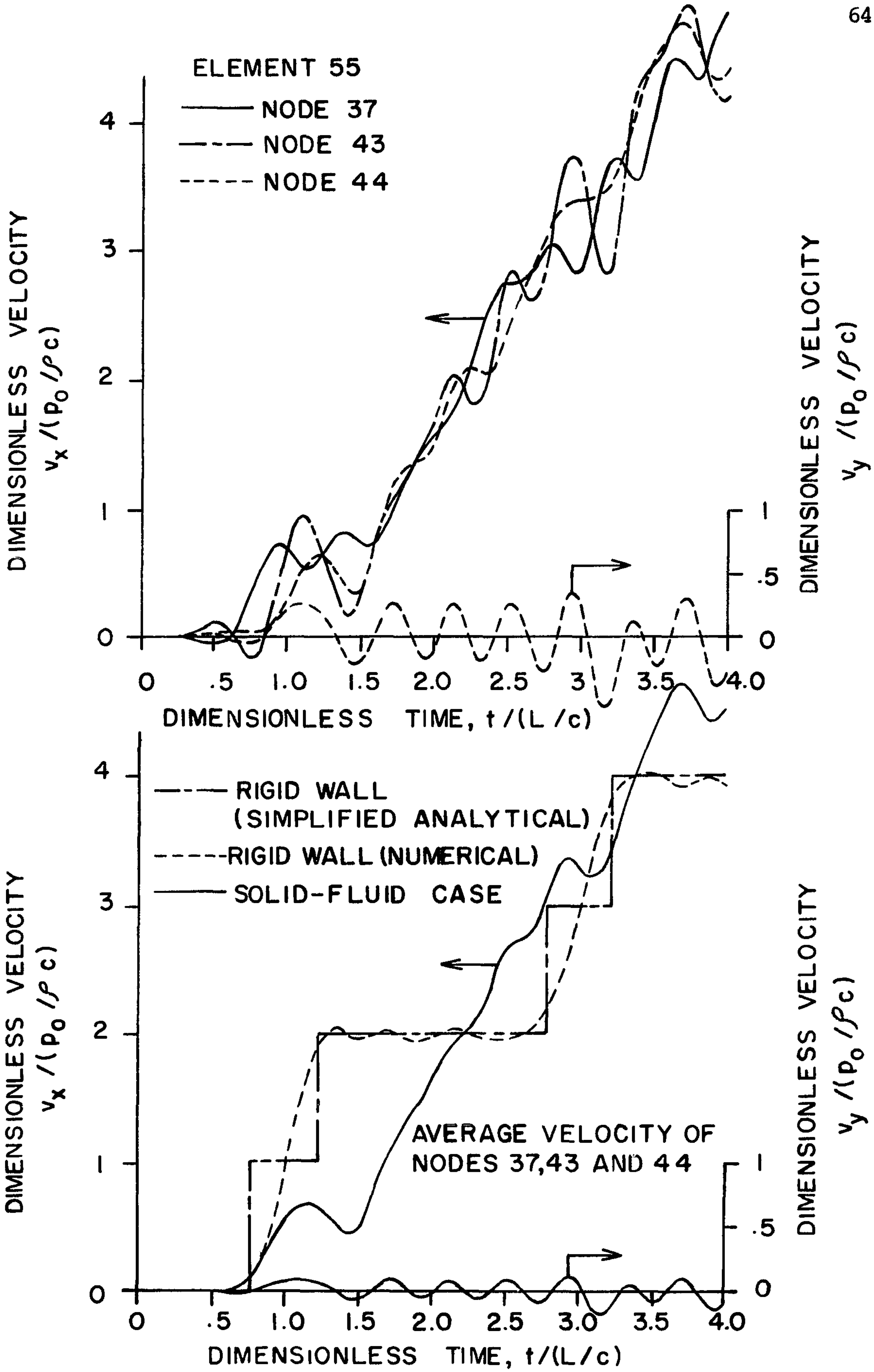


FIG. 23 A COMPARISON OF FLUID VELOCITY TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH INVISCID FLOW (AT x = 23.7 IN. NEAR THE CHANNEL CENTERLINE)

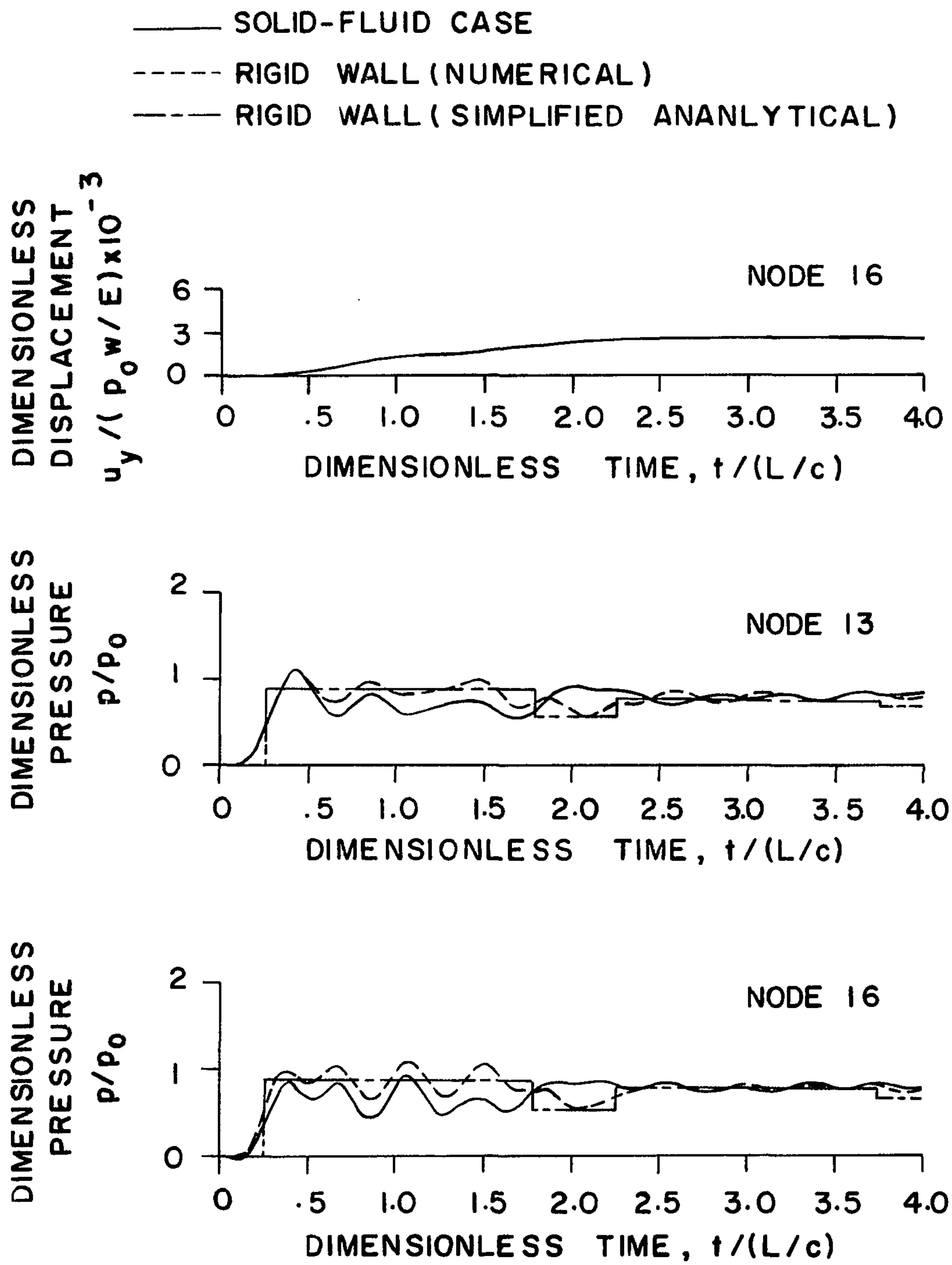
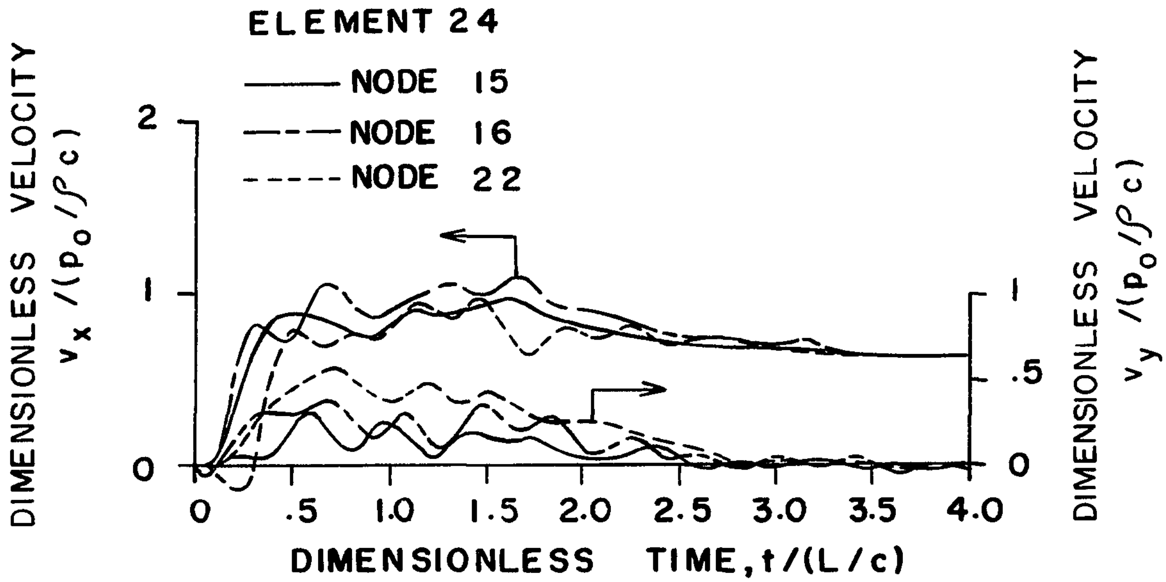


FIG. 24 A COMPARISON OF PRESSURE AND DISPLACEMENT TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH HIGHLY VISCOUS FLOW (AT x = 6.9 IN.)



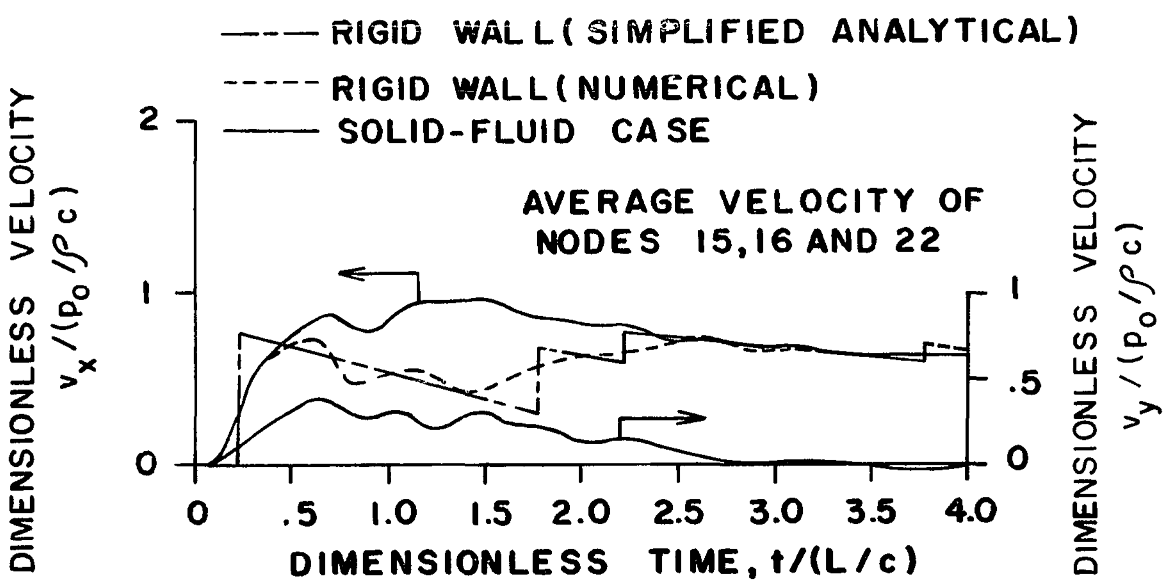


FIG. 25 A COMPARISON OF FLUID VELOCITY TIME HISTORIES FOR ELASTIC AND RIGID WALL CASES WITH HIGHLY VISCOUS FLOW (AT x = 6.9 IN.)

5. CONCLUSIONS

A finite element model for the study of solid-fluid interaction is presented. This finite element model may be used to analyze two-dimensional solid-fluid interaction problems involving complex geometries and loadings. The model is based on the discretization of the solid equation of motion and the fluid continuity and momentum equations. It employs the solid displacements together with the fluid pressure and velocity components as the nodal degrees of freedom. This facilitates the implementation of realistic boundary conditions, in contrast with formulations using stream function, vorticity and pressure as the only fluid field variables.

The elemental matrices are used directly in the numerical solution. This feature results in a considerable saving in computer storage and running time, in contrast with most schemes that the assembly and storage of system global matrices are essential to the numerical solution.

The model was tested for a flow configuration consisting of water, between two elastic parallel plates, subjected to a step pressure at one end. Verification of the model was achieved by comparing the results with a previous study based on the finite element discretization of the two-dimensional wave equation. Further comparison with flow between two rigid parallel plates demonstrated the development of a transversal water hammer phenomenon and pressure surge caused by the wall deformation and the fluid transversal flow. This transversal water hammer has a substantial effect on the response characteristics and may have important implications in the design of fluid systems. Parametric studies , conducted to observe the effect of fluid damping, demonstrated that damping can effectively reduce the amplitude of the pressure surge caused by the transversal water hammer.

NOMENCLATURE

A	Finite element area
A_1, A_2, A_2', B_1, B_2	Integration constants defined in the text
a_i, b_i, c_i	Quantities defined by equations (A-5)
$[B]$	Strain-Displacement matrix
c	Speed of sound in continuum under consideration
$[C]$	Damping or equivalent damping matrix depending upon superscript
$\lceil C' \rfloor$	Diagonal matrix with diagonal terms equal to diagonal terms of $[C]$
$[C'']$	Off-diagonal matrix with off-diagonal terms equal to off-diagonal terms of $[C]$ and zeros along the diagonal
$[D]$	Inertia matrix in fluid multi-variable pressure-velocity formulation
$\lceil D' \rfloor$	Diagonal matrix with diagonal terms equal to diagonal terms of $[D]$
$[D'']$	Off-diagonal matrix with off-diagonal terms equal to off-diagonal terms of $[D]$ and zeros along the diagonal
e	Element index number
$[E]$	Fluidity matrix in fluid multi-variable pressure-velocity formulation
$\lceil E' \rfloor$	Diagonal matrix with diagonal terms equal to diagonal terms of $[E]$
$[E'']$	Off-diagonal matrix with off-diagonal terms equal to off-diagonal terms of $[E]$ and zeros along the diagonal
E	Young's modulus
$\{F\}$	Boundary integral array in fluid single-variable pressure formulation

$\{F'\}$	Forcing function array defined by equations (15),(17), (33) or (35) depending upon subscript and superscript
f	Individual terms of $\{F'\}$
$[G]$	Inertia matrix in fluid single-variable pressure formulation
h	Channel height
$\tilde{h}$	Constant value of $\tilde{h}$ used in eqs.(D-13) and (D-15)
$[H]$	Fluidity matrix in fluid single-variable pressure formulation
i,j	Node number
$[I]$	Identity matrix
I_1, I_6	Integrals defined by eqs.(B-8) and (B-11)
k_f, k_s	Viscous or damping parameter
k	$=\mu/\sigma$ in equation (19)
k_{ij}	Terms of stiffness matrix $[K]$
$[K]$	Stiffness or equivalent stiffness matrix depending upon superscript
$\lceil K' \rfloor$	Diagonal matrix with diagonal terms equal to diagonal terms of $[K]$
$[K'']$	Off-diagonal matrix with off-diagonal terms equal to off-diagonal terms of $[K]$ and zeros along the diagonal
L	Channel length
$[L]$	Viscous damping matrix in fluid single-variable pressure formulation
ℓ_{min}	Smallest length increment
m	Slope of solid-fluid boundary
$[M]$	Mass or equivalent mass matrix depending upon superscript
$\lceil M' \rfloor$	Diagonal matrix with diagonal terms equal to diagonal terms of $[M]$
$[M'']$	Off-diagonal matrix with off-diagonal terms equal to off-diagonal terms of $[M]$ and zeros along the diagonal
$[N]$	Element shape function defined by equations (A-4),(A-5), (C- 2) and (C-3)

n	Maximum number of elements
p	Fluid pressure
p_0	Fluid pressure at channel entrance
$\{p\}$	Elemental array of nodal pressures
$\{R\}$	Elemental array of loads on the solid due to fluid pressure
$\{R'\}$	Elemental or global array of applied nodal loads
r_1, r_2	Characteristic roots defined by equations (45)
S	Solid-fluid boundary
$S_{i,j}$	Terms of coupling matrix $[S_e]$
$[S_e]$	Solid-fluid coupling matrix defined by eq.(D-11)
$[R],[S],[T]$	Matrices defined by equations(A-13),(A-14) and (A-15) with terms R_{ij}, S_{ij} and T_{ij}, respectively
T_{min}	Minimum period associated with smallest length increment
t	Time
U	Array of elemental freedom for the solid (x and y components of displacements)
u_x, u_y	Solid displacements in x and y directions respectively
v_x, v_y	Fluid velocity components in x and y directions respectively
w	Channel wall thickness
W	Work
x,y	Cartesian coordinate system
x',y'	Cartesian coordinate system passing through the centroid
x_o, x_f	Abscissa of first and last node of solid-fluid boundary, respectively
$\{X\}$	Array of elemental or global freedoms (x-component of solid displacement, y-component of solid displacement, fluid pressure, x-component of fluid velocity and y-component of fluid velocity)
x_1, x_2, x_3	Abscissa of nodes 1,2,3 respectively
x	Individual terms of X
y_o, y_f	Ordinate of first and last nodes of solid-fluid boundary
y_1, y_2, y_3	Ordinates of nodes 1,2,3 respectively
$\{Z\}$	Array of elemental or global freedoms for the fluid (fluid pressure, x-component of fluid velcity and y-component of fluid velocity)

z	Nodal freedoms(pressure,x-component of velocity or y-component of velocity)

Greek Symbols

$\alpha_i, \alpha_j, \alpha_k$	Coefficients used in equations (A-1),(A-2) and (C-14) for finite element discretization
α	Angle of solid-fluid boundary with x-axis
δ	Virtual displacement
Δt	Time increment
$[\Delta]$	Stress-strain matrix
λ	Individual terms of $\lceil \bar{M}' \rfloor$
μ	Individual terms of $\lceil \bar{K}' \rfloor$ or $\lceil \bar{E}' \rfloor$,whichever applies
ξ	Pseudo damping ratio defined by equation (39)
ν	Poisson's ratio for solid
ρ	Continuum density depending upon subscript
σ	Individual terms of $\lceil \bar{C}' \rfloor$ or $\lceil \bar{D}' \rfloor$,whichever applies
τ	Time within the interval t and t+Δt
ω_n	Pseudo natural frequency defined by equation (39)
ω_d	Pseudo damped natural frequency defined by equation (41)

Superscripts

$\cdot$, $\cdot\cdot$	First and second time derivatives
1,2	First and second trial values
T	Matrix transpose
$-$	Global matrices and arrays
$=$	Solid-fluid boundary

Subscripts

c	Refers to the centroid of the triangular element
e	Elemental matrices and arrays (without e refers to global matrices and arrays)
f	Fluid
i	Node number
j	Updated time
j-1	Present time

N Normal

s Entropy constant or solid
T Tangential

o Past values

x x-direction

y y-direction

Operators

d Virtual

APPENDIX 1 - DERIVATION OF THE FLUID FINITE ELEMENT EQUATIONS

A plane triangular fluid finite element, shown in Figure 2, is used as the basis for the fluid finite element formulation. It is assumed that the fluid pressure and velocity components at any point in the triangular fluid element may be expressed as linear polynomials in x and y:

$$p = \alpha_1 + \alpha_2 x + \alpha_3 y$$

$$v_x = \alpha_4 + \alpha_5 x + \alpha_6 y$$

$$v_y = \alpha_7 + \alpha_8 x + \alpha_9 y \qquad \text{(A-1)}$$

or generally

$$z = \alpha_i + \alpha_j x + \alpha_k y \qquad \text{(A-2)}$$

in which the coefficients α are functions of time. Employing the above equations, it is shown in Appendix 2 that

$$z = \lfloor N_f(x,y) \rfloor \{z_e(t)\} \qquad \text{(A-3)}$$

in which the shape function $\lfloor N_f \rfloor$ is given by

$$\lfloor N_f \rfloor = \lfloor N_{f1} \quad N_{f2} \quad N_{f3} \rfloor \qquad \text{(A-4)}$$

where

$$N_{fi} = \frac{1}{2A_f} (a_i + b_i x + c_i y) \qquad i = 1, 2, 3$$

$$a_1 = x_2y_3 - x_3y_2 \qquad b_1 = y_2 - y_3 \qquad c_1 = x_3 - x_2$$

$$a_2 = x_3y_1 - x_1y_3 \qquad b_2 = y_3 - y_1 \qquad c_2 = x_1 - x_3$$

$$a_3 = x_1y_2 - x_2y_1 \qquad b_3 = y_1 - y_2 \qquad c_3 = x_2 - x_1 \tag{A-5}$$

and A_f is the area of the triangular fluid element

$$A_f = (a_1 + a_2 + a_3)/2 \tag{A-6}$$

Employing the Galerkin method of weighted residuals, equations (5), (6), and (7) become

$$\iint_A [N_f]^T\{\frac{1}{c^2}\frac{\partial p}{\partial t} + \rho_f(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y})\}\ dx\ dy = 0 \tag{A-7}$$

$$\iint_A [N_f]^T\{\frac{\partial v_x}{\partial t} + \frac{1}{\rho_f}\frac{\partial p}{\partial x} + k_f v_x\}\ dx\ dy = 0 \tag{A-8}$$

$$\iint_A [N_f]^T\{\frac{\partial v_y}{\partial t} + \frac{1}{\rho_f}\frac{\partial p}{\partial y} + k_f v_y\}\ dx\ dy = 0 \tag{A-9}$$

Substituting eq. (A-3) into eqs. (A-7), (A-8) and (A-9), one obtains

$$\iint_A \frac{1}{c^2}[N_f]^T [N_f]\{\dot{p}_e\}\ dx\ dy + \iint_A \rho_f [N_f]^T \frac{\partial [N_f]}{\partial x} \{v_{xe}\}\ dx\ dy$$

$$+ \iint_A \rho_f [N_f]^T \frac{\partial [N_f]}{\partial y}\{v_{ye}\}\ dx\ dy = 0 \tag{A-10}$$

$$\iint_A [N_f]^T [N_f] \{\dot{v}_{xe}\} \;\; dx\, dy + \iint_A \frac{1}{\rho_f} [N_f]^T \frac{\partial [N_f]}{\partial x} \{p_e\} \;\; dx\, dy$$

$$+ \iint_A k_f [N_f]^T \; [N_f] \; \{v_{xe}\} \; dx\, dy = 0 \qquad \text{(A-11)}$$

$$\iint_A [N_f]^T [N_f] \{\dot{v}_{ye}\} \;\; dx\, dy + \iint_A \frac{1}{\rho_f} [N_f]^T \frac{\partial [N_f]}{\partial y} \{p_e\} \;\; dx\, dy$$

$$+ \iint_A k_f [N_f]^T \; [N_f] \; \{v_{ye}\} \; dx\, dy = 0 \qquad \text{(A-12)}$$

The above equations involve calculation of the following three types of matrices

$$[R] = \iint_A [N_f]^T \; [N_f] \;\; dx\, dy \qquad \text{(A-13)}$$

$$[S] = \iint_A [N_f]^T \frac{\partial [N_f]}{\partial x} \, dx\, dy \qquad \text{(A-14)}$$

$$[T] = \iint_A [N_f]^T \frac{\partial [N_f]}{\partial y} \, dx\, dy \qquad \text{(A-15)}$$

Performing the integration, as shown in Appendix 2, one obtains

$$[R] = \frac{A_f}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \quad \text{(A-16)}$$

$$[S] = \frac{1}{6} \begin{bmatrix} b_1 & b_2 & b_3 \\ b_1 & b_2 & b_3 \\ b_1 & b_2 & b_3 \end{bmatrix} \quad \text{(A-17)}$$

$$[T] = \frac{1}{6} \begin{bmatrix} c_1 & c_2 & c_3 \\ c_1 & c_2 & c_3 \\ c_1 & c_2 & c_3 \end{bmatrix} \quad \text{(A-18)}$$

Substituting equations (A-16) through (A-18) into equations (A-10) through (A-12), combining the resulting equations into one matrix equation after dividing the pressure equation by ρ_f and multiplying the velocity equations by ρ_f , one obtains

$$[D_e]\{\dot{Z}_e\} + [E_e]\{Z_e\} = 0 \quad \text{(A-19)}$$

where

$$\{\dot{Z}_e\} = \begin{Bmatrix} \dot{p}_1 \\ \dot{v}_{x1} \\ \dot{v}_{y1} \\ \dot{p}_2 \\ \dot{v}_{x2} \\ \dot{v}_{y2} \\ \dot{p}_3 \\ \dot{v}_{x3} \\ \dot{v}_{y3} \end{Bmatrix} \qquad \{Z_e\} = \begin{Bmatrix} p_1 \\ v_{x1} \\ v_{y1} \\ p_2 \\ v_{x2} \\ v_{y2} \\ p_3 \\ v_{x3} \\ v_{y3} \end{Bmatrix} \tag{A-20}$$

$$[D_e] = A_f \begin{bmatrix} \frac{1}{6c^2\rho_f} & 0 & 0 & \frac{1}{12c^2\rho_f} & 0 & 0 & \frac{1}{12c^2\rho_f} & 0 & 0 \\ 0 & \frac{\rho_f}{6} & 0 & 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{12} & 0 \\ 0 & 0 & \frac{\rho_f}{6} & 0 & 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{12} \\ \frac{1}{12c^2\rho_f} & 0 & 0 & \frac{1}{6c^2\rho_f} & 0 & 0 & \frac{1}{12c^2\rho_f} & 0 & 0 \\ 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{6} & 0 & 0 & \frac{\rho_f}{12} & 0 \\ 0 & 0 & \frac{\rho_f}{6} & 0 & 0 & \frac{\rho_f}{6} & 0 & 0 & \frac{\rho_f}{12} \\ \frac{1}{12c^2\rho_f} & 0 & 0 & \frac{1}{12c^2\rho_f} & 0 & 0 & \frac{1}{6c^2\rho_f} & 0 & 0 \\ 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{6} & 0 \\ 0 & 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{12} & 0 & 0 & \frac{\rho_f}{6} \end{bmatrix} \tag{A-21}$$

$$[E_e] = \begin{bmatrix}
0 & \frac{b_1}{6} & \frac{c_1}{6} & 0 & \frac{b_2}{6} & \frac{c_2}{6} & 0 & \frac{b_3}{6} & \frac{c_3}{6} \\
\frac{b_1}{6} & \frac{\rho_f k_f A_f}{6} & 0 & \frac{b_2}{6} & \frac{\rho_f k_f A_f}{12} & 0 & \frac{b_3}{6} & \frac{\rho_f k_f A_f}{12} & 0 \\
\frac{c_1}{6} & 0 & \frac{\rho_f k_f A_f}{6} & \frac{c_2}{6} & 0 & \frac{\rho_f k_f A_f}{12} & \frac{c_3}{6} & 0 & \frac{\rho_f k_f A_f}{12} \\
0 & \frac{b_1}{6} & \frac{c_1}{6} & 0 & \frac{b_2}{6} & \frac{c_2}{6} & 0 & \frac{b_3}{6} & \frac{c_3}{6} \\
\frac{b_1}{6} & \frac{\rho_f k_f A_f}{12} & 0 & \frac{b_2}{6} & \frac{\rho_f k_f A_f}{6} & 0 & \frac{b_3}{6} & \frac{\rho_f k_f A_f}{12} & 0 \\
\frac{c_1}{6} & 0 & \frac{\rho_f k_f A_f}{12} & \frac{c_2}{6} & 0 & \frac{\rho_f k_f A_f}{6} & \frac{c_3}{6} & 0 & \frac{\rho_f k_f A_f}{12} \\
0 & \frac{b_1}{6} & \frac{c_1}{6} & 0 & \frac{b_2}{6} & \frac{c_2}{6} & 0 & \frac{b_3}{6} & \frac{c_3}{6} \\
\frac{b_1}{6} & \frac{\rho_f k_f A_f}{12} & 0 & \frac{b_2}{6} & \frac{\rho_f k_f A_f}{12} & 0 & \frac{b_3}{6} & \frac{\rho_f k_f A_f}{6} & 0 \\
\frac{c_1}{6} & 0 & \frac{\rho_f k_f A_f}{12} & \frac{c_2}{6} & 0 & \frac{\rho_f k_f A_f}{12} & \frac{c_3}{6} & 0 & \frac{\rho_f k_f A_f}{6}
\end{bmatrix} \tag{A-22}$$

APPENDIX 2 - DEVELOPMENT OF FLUID SHAPE FUNCTION AND MATRICES

As stated earlier, the fluid pressure and velocity components at any point in triangular fluid element are given by equation (A-2). Expressing this equation for points 1, 2, and 3 in Fig. 2 , and eliminating α_i, α_j, and α_k among the resulting equations yields

$$\alpha_i = \frac{1}{2A_f} \begin{vmatrix} z_1 & x_1 & y_1 \\ z_2 & x_2 & y_2 \\ z_3 & x_3 & y_3 \end{vmatrix}$$

$$\alpha_j = \frac{1}{2A_f} \begin{vmatrix} 1 & z_1 & y_1 \\ 1 & z_2 & y_2 \\ 1 & z_3 & y_3 \end{vmatrix}$$

$$\alpha_k = \frac{1}{2A_f} \begin{vmatrix} 1 & x_1 & z_1 \\ 1 & x_2 & z_2 \\ 1 & x_3 & z_3 \end{vmatrix} \qquad \text{(B-1)}$$

where

$$A_f = \frac{1}{2} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} = \text{(Area of triangle 123)} \qquad \text{(B-2)}$$

Substituting the values of α_i, α_j, and α_k into equation (A-2) and factoring nodal{z}array gives

$$\{z\} = \frac{1}{2A_f} \left[(a_1 + b_1 x + c_1 y) \quad (a_2 + b_2 x + c_2 y) \quad (a_3 + b_3 x + c_3 y) \right] \begin{Bmatrix} z_1 \\ z_2 \\ z_3 \end{Bmatrix} \tag{B-3}$$

where a_i, b_i, c_i and A_f are defined in Appendix 1. A comparison of the above equation with equation (A-3) yields the shape function given by equation (A-4). Substituting the shape function N_f into equation (A-13) through (A-15) yields

$$[R] = \frac{1}{4A_f^2} \iint_A \begin{bmatrix} (a_1+b_1x+c_1y)^2 & (a_1+b_1x+c_1y)(a_2+b_2x+c_2y) & (a_1+b_1x+c_1y)(a_3+b_3x+c_3y) \\ & (a_2+b_2x+c_2y)^2 & (a_2+b_2x+c_2y)(a_3+b_3x+c_3y) \\ \text{Symmetric} & & (a_3+b_3x+c_3y)^2 \end{bmatrix} dx\, dy \tag{B-4}$$

$$[S] = \frac{1}{4A_f^2} \iint_A \begin{bmatrix} b_1(a_1+b_1x+c_1y) & b_2(a_1+b_1x+c_1y) & b_3(a_1+b_1x+c_1y) \\ b_1(a_2+b_2x+c_2y) & b_2(a_2+b_2x+c_2y) & b_3(a_2+b_2x+c_2y) \\ b_1(a_3+b_3x+c_3y) & b_2(a_3+b_3x+c_3y) & b_3(a_3+b_3x+c_3y) \end{bmatrix} dx\, dy \tag{B-5}$$

$$[T] = \frac{1}{4A_f^2} \iint_A \begin{bmatrix} c_1(a_1+b_1x+c_1y) & c_2(a_1+b_1x+c_1y) & c_3(a_1+b_1x+c_1y) \\ c_1(a_2+b_2x+c_2y) & c_2(a_2+b_2x+c_2y) & c_3(a_2+b_2x+c_2y) \\ c_1(a_3+b_3x+c_3y) & c_2(a_3+b_3x+c_3y) & c_3(a_3+b_3x+c_3y) \end{bmatrix} dx\, dy \tag{B-6}$$

To determine the fluid matrices R, S, and T, twenty seven integrations indicated by equations (B-4), (B-5) and (B-6) should be performed. To demonstrate this lenghty process, it suffices here to show the procedure involved in the evaluation of R_{11}, S_{11}, and T_{11} given by

$$R_{11} = \frac{1}{4A_f^2} \iint_A (a_1 + b_1 x + c_1 y)^2 \, dx\, dy$$

$$S_{11} = \frac{1}{4A_f^2} \iint_A b_1(a_1 + b_1 x + c_1 y) \, dx\, dy$$

$$T_{11} = \frac{1}{4A_f^2} \iint_A c_1(a_1 + b_1 x + c_1 y) \, dx\, dy \qquad \text{(B-7)}$$

The integration of the above terms involve the calculation of the following integrals

$$I_1 = \iint_A dx\, dy \qquad I_2 = \iint_A x\, dx\, dy \qquad I_3 = \iint_A x^2\, dx\, dy$$

$$I_4 = \iint_A y\, dx\, dy \qquad I_5 = \iint_A y^2 dx\, dy \qquad I_6 = \iint_A xy\, dx\, dy$$

(B-8)

Performing transformation of coordinates, involving a pure translation, from point (0,0) to the centroid of the triangle (x_c, y_c) such that

$$x = x' + y_c$$

$$y = y' + y_c \qquad \text{(B-9)}$$

where

$$x_c = \frac{1}{3} (x_1 + x_2 + x_3) \qquad y_c = \frac{1}{3} (y_1 + y_2 + y_3) \qquad \text{(B-10)}$$

The above integrals are calculated as follows by noting that the first moment with respect to the centroid vanishes.

$$I_1 = \iint_A dx\, dy = \iint_A dx'\, dy' = A_f$$

$$I_2 = \iint_A x\, dx\, dy = \iint_A (x' + x_c)\, dx'\, dy' = x_c A_f$$

$$I_3 = \iint_A x^2 dx\, dy = \iint_A (x' + x_c)^2\, dx'\, dy' = \frac{A_f}{12} (x_1^2 + x_2^2 + x_3^2) + \frac{3}{4} x_c^2 A_f$$

$$I_4 = \iint_A y\, dx\, dy = y_c A_f$$

$$I_5 = \iint_A y^2 dx\, dy = \frac{A_f}{12} (y_1^2 + y_2^2 + y_3^2) + \frac{3}{4} y_c^2 A_f$$

$$I_6 = \iint_A xy\, dx\, dy = \frac{A_f}{12} (x_1 y_1 + x_2 y_2 + x_3 y_3) + \frac{3}{4} x_c y_c A_f \qquad \text{(B-11)}$$

Employing the above integrals, expressions for S_{11} reduces to

$$S_{11} = \frac{1}{4A_f^2} (a_1 b_1 A_f + b_1^2 x_c A_f + b_1 c_1 A_f) = \frac{b_1}{4A_f} (a_1 + b_1 x_c + c_1 y_c) \qquad \text{(B-12)}$$

Substituting for a_1, b_1, c_1, x_c and y_c from equations (A-5) and (B-10), one obtains

$$S_{11} = \frac{b_1}{4A_f} \Big[(x_2 y_3 - x_3 y_2) + (y_2 - y_3) \left(\frac{x_1 + x_2 + x_3}{3}\right) + (x_3 - y_2) \left(\frac{y_1 + y_2 + y_3}{3}\right)\Big]$$

(B-13)

Upon further simplification

$$S_{11} = \frac{b_1}{4A_f} \left(\frac{a_1 + a_2 + a_3}{3}\right) = \frac{b_1}{6} \quad \text{(B-14)}$$

Similarly, it can be shown that

$$T_{11} = \frac{c_1}{6} \quad \text{(B-15)}$$

Proceeding in a similar manner, the expression for R_{11} becomes

$$R_{11} = (a_1^2/4A_f) + (b_1^2/16A_f)\left[\frac{1}{3}(x_1^2 + x_2^2 + x_3^2) + 3x_c^2\right]$$

$$+ (c_1^2/16A_f)\left[\frac{1}{3}(y_1^2 + y_2^2 + y_3^2) + 3y_c^2\right] + a_1 b_1 x_c/2A_f$$

$$+ (b_1 c_1/8A_f)\left[\frac{1}{3}(x_1 y_1 + x_2 y_2 + x_3 y_3) + 3x_c y_c\right] + a_1 c_1 y_c/2A_f$$

(B-16)

Substituting for a_1, b_1, c_1, x_c, and y_c from equation (A-5) and (B-10) one obtains

$$R_{11} = \frac{1}{72A_f}(6a_1^2 + 6a_2^2 + 12a_1 a_2 + 12a_1 a_3 + 12a_2 a_3) \quad \text{(B-17)}$$

Upon further simplification

$$R_{11} = A_f/6 \quad \text{(B-18)}$$

APPENDIX 3 - DERIVATION OF SOLID FINITE ELEMENT EQUATIONS

The solid finite element used in this study is developed based on the method of virtual work. This method equates the work and change in strain energy in a system generated during a virtual displacement. This procedure is well known and has been documented [16,23,15,4,21].

A plane triangular solid finite element, as shown in Fig. 13, is used as the basis for the solid finite element formulation. It is assumed that the displacement components at any point in the triangular solid element, u_x and u_y, may be expressed as a polynomial in x and y. In this case

$$u_x = \alpha_{10} + \alpha_{11} x + \alpha_{12}\, y \qquad u_y = \alpha_{13} + \alpha_{14}\, x + \alpha_{15}\, y \tag{C-1}$$

Employing the above equations in the same fashion as the pressure and velocity components in the fluid finite element, it can be shown that

$$\begin{Bmatrix} u_x \\ \\ u_y \end{Bmatrix} = [N_s]\ \{u_e\} = \begin{bmatrix} N_{s1} & 0 & N_{s2} & 0 & N_{s3} & 0 \\ \\ 0 & N_{s1} & 0 & N_{s2} & 0 & N_{s3} \end{bmatrix} \{u_e\} \tag{C-2}$$

where $\{u_e\}$ is the array of time-dependent nodal displacements and $[N_s]$ is the solid shape function given by

$$N_{si} = \frac{1}{2A_s} \left(a_i + b_i\, x + c_i\, y \right) \qquad i = 1,2,3 \tag{C-3}$$

in which all the terms are defined as for the fluid element in Appendix 1. A_s is the area of the triangular solid element.

$$A_s = (a_1 + a_2 + a_3) /2 \quad \text{(C-4)}$$

The relationship in eq.(C -2) is used in conjunction with the principle of virtual work and the theory of elasticity to yield the matrix differential equation for the nodal displacements of the solid finite element. The discretization procedure is well documented [16,23,15,7,18] and yields

$$[M_e] \{\ddot{U}_e\} + [C_e] \{\dot{U}_e\} + [K_e] \{U_e\} = \{R'_e\} \quad \text{(C-5)}$$

where

Solid mass matrix $[M_e] = \int\int_A [N_s] \rho_s [N_s] \, dA$

Solid damping matrix $[C_e] = k_s [M_e]$ (C-6)

Solid stiffness matrix $[K_e] = \int\int_A [B]^T [\Delta] [B] \, dA$

$\{R'_e\}$ is the external forcing function acting on the solid, ρ_s is density of the solid, $[C_e]$ is the proportional damping matrix related to the mass matrix by the damping paramater k_s[1] , matrix $[B]$ is the strain-displacement matrix and $[\Delta]$ is the elasticity matrix. The boundary conditions which may be applied to eq. (C-5) are specified nodal displacements.

---

[1] The damping matrices used throughout this study are proportional to mass matrix only.

The mass and damping matrices may be evaluated by directly substituting in the solid shape function (eq.(C -2) and (C -3))into eqs.(C-6) and performing the required integration. This yields the consistent mass matrix[15].

$$[M_e] = \frac{\rho_s\ A_s}{3}\begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{2} & 0 & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & 0 & \frac{1}{2} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{2} \end{bmatrix} \qquad (C-7)$$

Two types of mass matrices may be employed. The first is the consistent mass matrix given above. It is called consistent because it results directly from the finite element formulation. The second is the lumped mass matrix. In earlier solutions of structural dynamics problems, the mass of the element was arbitrarily lumped or concentrated at the nodes. This results in a mass matrix of the form [15]

$$[M_e] = \frac{\rho_s\ A_s}{3}\ [I_6] \qquad (C-8)$$

in which $[I_6]$ is the 6 x 6 identity matrix. The consistent mass matrix is more appealing from the point of view that it is a direct product of the mathematical formulation. It has also been observed that the off-diagonal terms in the consistent mass matrix produce sufficient numerical noise to obscure the actual results[15].It has been demonstrated that for a given number of degrees of freedom,the lumped-mass representation is less accurate than the consistent mass; however, in many practical applications it is still preferable to use the lumped-mass matrices because of the significant computational advantage derived from the fact that such matrices are diagonal[23] . For these reasons,lumped mass mass matrices have been used throughout this study.

The stiffness matrix for the plane triangular linear finite element is a 6 x 6 symmetric matrix as follows [15].

$$[K_e] = \frac{E}{4\,A_s(1-\nu^2)} \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \\ & & k_{33} & k_{34} & k_{35} & k_{36} \\ & & & k_{44} & k_{45} & k_{46} \\ & \text{Symmetric} & & & k_{55} & k_{56} \\ & & & & & k_{66} \end{bmatrix} \qquad \text{(C-9)}$$

in which

$$k_{11} = b_1^2 + \frac{1-\nu}{2} c_1^2$$

$$k_{12} = \frac{1+\nu}{2} b_1 c_1 = k_{21}$$

$$k_{13} = b_1 b_2 + \frac{1-\nu}{2} c_1 c_2 = k_{31}$$

$$k_{14} = \nu b_1 c_2 + \frac{1-\nu}{2} b_2 c_1 = k_{41}$$

$$k_{15} = b_1 b_3 + \frac{1-\nu}{2} c_1 c_3 = k_{51}$$

$$k_{16} = \nu b_1 c_3 + \frac{1-\nu}{2} b_3 c_1 = k_{61}$$

$$k_{22} = c_1^2 + \frac{1-\nu}{2} b_1^2$$

$$k_{23} = \nu b_2 c_1 + \frac{1-\nu}{2} b_1 c_2 = k_{32}$$

$$k_{24} = c_1 c_2 + \frac{1-\nu}{2} b_1 b_2 = k_{42}$$

$$k_{25} = \nu b_3 c_1 + \frac{1-\nu}{2} b_1 c_3 = k_{52}$$

$$k_{26} = c_1 c_3 + \frac{1-\nu}{2} b_1 b_3 = k_{62}$$

$$k_{33} = b_2^2 + \frac{1-\nu}{2} c_2^2$$

$$k_{34} = \frac{1+\nu}{2} b_2 c_2 = k_{43}$$

$$k_{35} = b_2 b_3 + \frac{1-\nu}{2} c_2 c_3 = k_{53}$$

$$k_{36} = \nu b_2 c_3 + \frac{1-\nu}{2} b_3 c_2 = k_{63}$$

$$k_{44} = c_2^2 + \frac{1-\nu}{2} b_2^2$$

$$k_{45} = \nu b_3 c_2 + \frac{1-\nu}{2} b_2 c_3 = k_{54}$$

$$k_{46} = c_2 c_3 + \frac{1+\nu}{2} b_2 b_3 = k_{64}$$

$$k_{55} = b_3^2 + \frac{1-\nu}{2} c_3^2$$

$$k_{56} = \frac{1+\nu}{2} b_3 c_3 = k_{65}$$

$$k_{66} = c_3^2 + \frac{1-\nu}{2} b_3^2$$

APPENDIX 4. DERIVATION OF THE SOLID-FLUID SUPERELEMENT EQUATIONS

The finite element formulation of the solid-fluid superelement (i.e.,equation (28)) is based on the solid and fluid elements matrix equations as well as the interaction forces between the two elements.

The matrix equation for the solid-fluid superelement is obtained by combining equations (A-19) and (C-5) and including the interactive forces

$$\begin{bmatrix} M_e & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{U}_e \\ \ddot{Z}_e \end{Bmatrix} + \begin{bmatrix} C_e & 0 \\ 0 & D_e \end{bmatrix} \begin{Bmatrix} \dot{U}_e \\ \dot{Z}_e \end{Bmatrix} + \begin{bmatrix} K_e & 0 \\ 0 & E_e \end{bmatrix} \begin{Bmatrix} U_e \\ Z_e \end{Bmatrix} = \begin{Bmatrix} R_e \\ 0 \end{Bmatrix} + \begin{Bmatrix} R'_e \\ 0 \end{Bmatrix} \tag{D-1}$$

where $\{R_e\}$ and $\{R'_e\}$ are the fluid pressure force exerted on the solid representing the coupling force and the external forcing function acting on the solid respectively.

The contribution of the fluid pressure loading $\{R_e\}$ is determined by calculating the work done by the pressure force during the virtual displacement as detailed hereunder.

Computation of the Solid-Fluid Coupling Matrix

The solid-fluid finite element, as shown in Fig. 14, is a quadrilateral composed of a solid and a fluid finite element. The equations for the nodal displacement components, pressure and velocity components of the quadrilateral solid-fluid element are obtained from the seperate fluid and solid parts with the addition of the interactive forces. For the solid part, the interactive force is the pressure force acting normal to the moving boundary. For the fluid part, the interactive term is established by the solid-fluid boundary constraint given in eq.(29).

The interactive force due to pressure load on the solid is determined by calculating the work done by the pressure force during the virtual displacement as follows [11,15]:

$$dW_p = \int_S (du_N)^T \; p \; dS \qquad \text{(D-2)}$$

The virtual displacement is given by

$$d\delta = d\begin{Bmatrix} u_x \\ u_y \end{Bmatrix} = [N_s] \; d\{u_e\} \qquad \text{(D-3)}$$

Resolving it into its normal and tangential components by a coordinate transformation one obtains

$$d\begin{Bmatrix} u_T \\ u_N \end{Bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} d\begin{Bmatrix} u_x \\ u_y \end{Bmatrix} \qquad \text{(D-4)}$$

where α is the angle the solid-fluid boundary makes with the horizontal axis, measured counterclockwise. The normal component of the virtual displacement is given by

$$du_N = [\,-\sin\alpha \quad \cos\alpha\,]\,[N_s]\,d\{u_e\} = [\bar{\bar{N}}_s]\,d\{u_e\} \tag{D-5}$$

in which

$$[\bar{\bar{N}}_s] = [\,-\sin\alpha \quad \cos\alpha\,]\,[N_s] \tag{D-6}$$

$[\bar{\bar{N}}_s]$ is the solid boundary shape function. Substituting eq. (D-6) into eq. (D-2) yields

$$dW_p = d\{u_e\}^T \int_S [\bar{\bar{N}}_s]^T\, p\, dS \tag{D-7}$$

Substituting eq. (A - 3) for pressure into eq. (D-7) gives

$$dW_p = d\{u_e\}^T \int_S [\bar{\bar{N}}_s]^T\,[N_f]\,dS\,\{p_e\} \tag{D-8}$$

The fluid pressure force on the solid-fluid boundary is

$$\{R_e\} = \int_S [\bar{\bar{N}}_s]^T\,[N_f]\,dS\,\{p_e\} \tag{D-9}$$

or simply

$$\{R_e\} = [S_e]\,\{p_e\} \tag{D-10}$$

where

$$[S_e] = \int_S [\bar{\bar{N}}_s]^T \; [N_f] \; dS \tag{D-11}$$

is the solid-fluid coupling matrix in which S is the line defining the solid-fluid boundary. Substituting eq.(D-10) into eq.(D-1), the final matrix equation for the superelement is obtained(i.e.eq.28). It shows clearly that the solid nodal displacements, u_x and u_y, and fluid nodal pressure, p , have been coupled. The boundary conditions for eq.(D-1) are the same as those indicated for the solid and the fluid parts.

Evaluation of the solid-fluid coupling matrix, eq. (D-11), for the case of a horizontal boundary (i.e., $y = \tilde{h}$, a constant, and $\alpha = 0$) yields a 6 x 3 matrix

$$[S_e] = \frac{1}{4A_s A_f} \begin{bmatrix} 0 & 0 & 0 \\ s_{21} & s_{22} & s_{23} \\ 0 & 0 & 0 \\ s_{41} & s_{42} & s_{43} \\ 0 & 0 & 0 \\ s_{61} & s_{62} & s_{63} \end{bmatrix} \tag{D-12}$$

in which

$$s_{2I,J} = (a_i \, a_j + (a_i \, c_j + a_j \, c_i) \, h + c_i \, c_j \, \tilde{h}^2)(x_f - x_o) +$$

$$(a_i \, b_j + a_j \, b_i + (b_i \, c_j + b_j \, c_i) \, \tilde{h}) \, (x_f^2 - x_o^2)/2 +$$

$$b_i \, b_j \, (x_f^3 - x_o^3)/3 \tag{D-13}$$

where i refers to the i^{th} node of the solid portion of the solid-fluid element and j refers to the j^{th} node of the fluid portion of the solid-fluid element, and x_o and x_f refer to the abscissa of the nodes defining the solid-fluid boundary. The coupling matrix may also be calculated for a vertical wall and the generalized case of a sloped wall, yielding similar results. For the vertical wall (i.e., $x = \tilde{h}$, a constant, and $\alpha = \pi/2$) the coupling matrix is

$$\left[S_e\right] = \frac{1}{4A_sA_f} \begin{bmatrix} -s_{11} & -s_{12} & -s_{13} \\ 0 & 0 & 0 \\ -s_{31} & -s_{32} & -s_{33} \\ 0 & 0 & 0 \\ -s_{51} & -s_{52} & -s_{53} \\ 0 & 0 & 0 \end{bmatrix} \tag{D-14}$$

in which

$$\begin{aligned} s_{(2I-1),J} = {} & (a_i a_j + (a_i b_j + a_j b_i)\tilde{h} + b_i b_j \tilde{h}^2)(y_f - y_o) \\ & + (a_i c_j + a_j c_i + (b_i c_j + b_j c_i)\tilde{h})(y_f^2 - y_o^2)/2 \\ & + c_i c_j (y_f^3 - y_o^3)/3 \end{aligned} \tag{D-15}$$

in which all terms are defined as before and y_o and y_f are the ordinates the nodes defining the solid-fluid boundary. For the general case of a sloped wall, the relationship along the boundary given by

$$y = \frac{y_f - y_o}{x_f - x_o}(x - x_o) + y_o = m(x - x_o) + y_o \tag{D-16}$$

is used. This results in the coupling matrix

$$\left[S_e\right] = \frac{1}{4A_sA_f} \begin{bmatrix} -s_{11}\sin\alpha & -s_{12}\sin\alpha & -s_{13}\sin\alpha \\ s_{11}\cos\alpha & s_{12}\cos\alpha & s_{13}\cos\alpha \\ -s_{31}\sin\alpha & -s_{32}\sin\alpha & -s_{33}\sin\alpha \\ s_{31}\cos\alpha & s_{32}\cos\alpha & s_{33}\cos\alpha \\ -s_{51}\sin\alpha & -s_{52}\sin\alpha & -s_{53}\sin\alpha \\ s_{51}\cos\alpha & s_{52}\cos\alpha & s_{53}\cos\alpha \end{bmatrix} \tag{D-17}$$

in which

$$s_{(2I-1),J} = (a_i\, a_j - (a_i\, c_j + a_j\, c_i)(m\, x_o - y_o) + c_i\, c_j\, (m\, x_o - y_o)^2)(x_f - x_o) + (a_i\, b_j + a_j\, b_i + (a_i\, c_j + a_j\, c_i)\, m - (b_i\, c_j + b_j\, c_i)(m\, x_o - y_o) - 2\, c_i\, c_j m\, (m\, x_o - y_o))(x_f^2 - x_o^2)/2 + (b_i\, b_j + (b_i\, c_j + b_j\, c_i)\, m + c_i\, c_j\, m\,)(x_f^3 - x_o^3)/3 \qquad \text{(D-18)}$$

$$\alpha = \tan^{-1} \frac{y_f - y_o}{x_f - x_o} \qquad \text{(D-19)}$$

and all other quantities are defined as before.

REFERENCES

[1] Chen, S.S., "Flow-Induced Vibration of Circular Cylindrical Structtures,Part I: Stationary Fluids and Parallel Flow", The Shock and Vibration Digest, vol. 9,No. 10, pp. 25-38, October 1977.

[2] Chen, S.S., "Flow-Induced Vibration of Circular Cylindrical Structures,Part II: Cross-Flow Considerations", The Shock and Vibration Digest, vol. 9, No. 11, pp. 21-28, November 1977.

[3] Conway, H.D. and Jakubowski, M., "Axial Impact of Short Cylindrical Bars", ASME publication 69-WA/APM-6 for meeting of November 16-20 1969.

[4] Cook, R.D., Concepts and Applications of Finite Element Analysis, John Wiley and Sons,Inc., New York, 1974.

[5] Feng, G.C., Kiefling, L., "Fluid-Structure Finite Element Vibrational Analysis", AIAA 74-102, AIAA 12th Aerospace Sciences Meeting, Washington, D.C., 1974.

[6] Gallagher, R.H., Oden, J.T., Taylor, C. and Zienkiewicz, O.C., "Finite Elements in Fluids", volumes 1 and 2, John Wiley and Sons, 1975.

[7] Gallagher, R.H., Finite Element Analysis Fundamentals, Prentice-Hall, Inc., New Jersey, 1975.

[8] Greenspon, J.E., "Fluid-Solid Interaction", The American Society of Mechanical Engineers, New York, 1967.

[9] Holmboe, E.L. and Rouleau, W.T., "The Effect of Viscous Shear on Transients in Liquid Lines", Transactions of ASME, Journal of Basic Engineering, vol. 89, No. 1,pp. 174-180, March 1967.

[10] Huebner, K.H., "The Finite Element Methods for Engineers", John Wiley and Sons, 1975.

[11] Nahavandi, A.N. and Pedrido, R.R., "An Analysis of Solid-Fluid Interaction Using the Finite Element Method", Dynamic Analysis of Pressure Vessel and Piping Components, ASME PVP-PB-022, pp. 75-93, 1977.

[12] Nahavandi, A.N., Bohm, G.J. and Pedrido, R.R., " Structurally Compatible Fluid Finite Element for Solid-Fluid Interaction Studies", Nuclear Engineering and Design 35 (1975) 335-347.

[13] Pettigrew, M.J. and Paidoussis, M.P., "Dynamics and Stability of Flexible Cylinders Subjected to Liquid and Two-Phase Axial Flow in Confined Annuli," 3rd Intl. Conf. Struc. Mech. Reactor Tech., London, vol. 1, Part D(sept. 1-5,1975).

[14] Paidoussis, M.P., "Vibration of Cylindrical Structures Induced by axial Flow", J. Engr. Indus., Trans., ASME, 96, pp. 547-553 (1974).

[15] Pedrido, R.R., "Dynamic Analysis of Structures with Solid-Fluid Interaction", Dr. Eng. Sc. Dissertation, New Jersey Institute of Technology, Newark, New Jersey (1977).

[16] Przemieniecki, J.S., Theory of Matrix Structural Analysis, New York: McGraw-Hill, 1968.

[17] Savkar, S.D., "A Survey of Flow Induced Vibrations of Cylindrical Arrays in Cross-Flow", ASME publication 76-WA/FE-21 for meeting of Dec. 5, 1976.

[18] Segerlind, L.J., Applied Finite Element Analysis, John Wiley and Sons,Inc., New Jersey, 1976.

[19] Szilard, R., Hydrodynamically Loaded Shells, Honolulu: The University Press of Hawaii, 1973.

[20] Tarantine, F.J. and Rouleau, W.T., "Fluid Pressure Transients in a Tapered Transmission Line", Transactions of ASME, Journal of Basic Engineering, vol. 89, No. 1,pp. 181-190, March 1967.

[21] Wang, C.K., Computer Methods in Advanced Structural Analysis, Intext Educational Publishers, New York, 1973.

[22] Zielke, W., "Frequency-Dependent Friction in Transient Pipe Flow", Transactions of ASME, Journal of Basic Engineering, March 1968, pp. 109-115.

[23] Zienkiewicz, O.C., The Finite Eleemnt Method in Engineering Science, London: McGraw-Hill, 1971.

PART THREE

USER'S MANUAL

1. DESCRIPTION OF FASINT DIGITAL COMPUTER PROGRAM

The program FASINT(Fluid And Solid Interaction),developed for the analysis of interaction between an elastic solid and a fluid medium having pressure and velocity components in the fluid as well as solid displacements as the main dependent variables, consists of a main program and several subroutines. At the user's option,the program may also be used to analyze totally solid or totally fluid continua. A tabular outline of the MAIN program and its subroutines together with their functions are given below and followed by a more detailed description. The flowchart of FASINT is shown in Fig.26.

PROGRAM OR SUBROUTINE NAME	FUNCTIONS
MAIN	Controls the calling sequence of subroutines,initialization of the problem, insertion of the specified freedoms and the normal termination of the program.
CLEAR	Initializes all labelled common blocks to zero.
GDATA	Reads in and prints out the input data and initializes time and iteration number.
LOAD	Reads in and assembles applied loads and specified freedoms(i.e.,displacements, pressure and velocities).
STIFT1(N)	Finds mass, damping and stiffness matrices for solid element number N.

PROGRAM OR SUBROUTINE NAME	FUNCTIONS
STIFT2(N)	Finds the inertia and fluidity matrices for fluid element number N.
STIFT3(N)	Finds the equivalent mass, combined damping-inertia and combined stiffness-fluidity matrices for solid-fluid superelement number N.
FORMK	Assembles the diagonal terms of matrices in global arrays and stores the off-diagonal terms.
SF(I,J)	Function subroutine used to calculate solid-fluid coupling matrix.
FORCE	Finds generalized global force array.
AINTEG	Finds analytical solution to the uncoupled global differential equations.

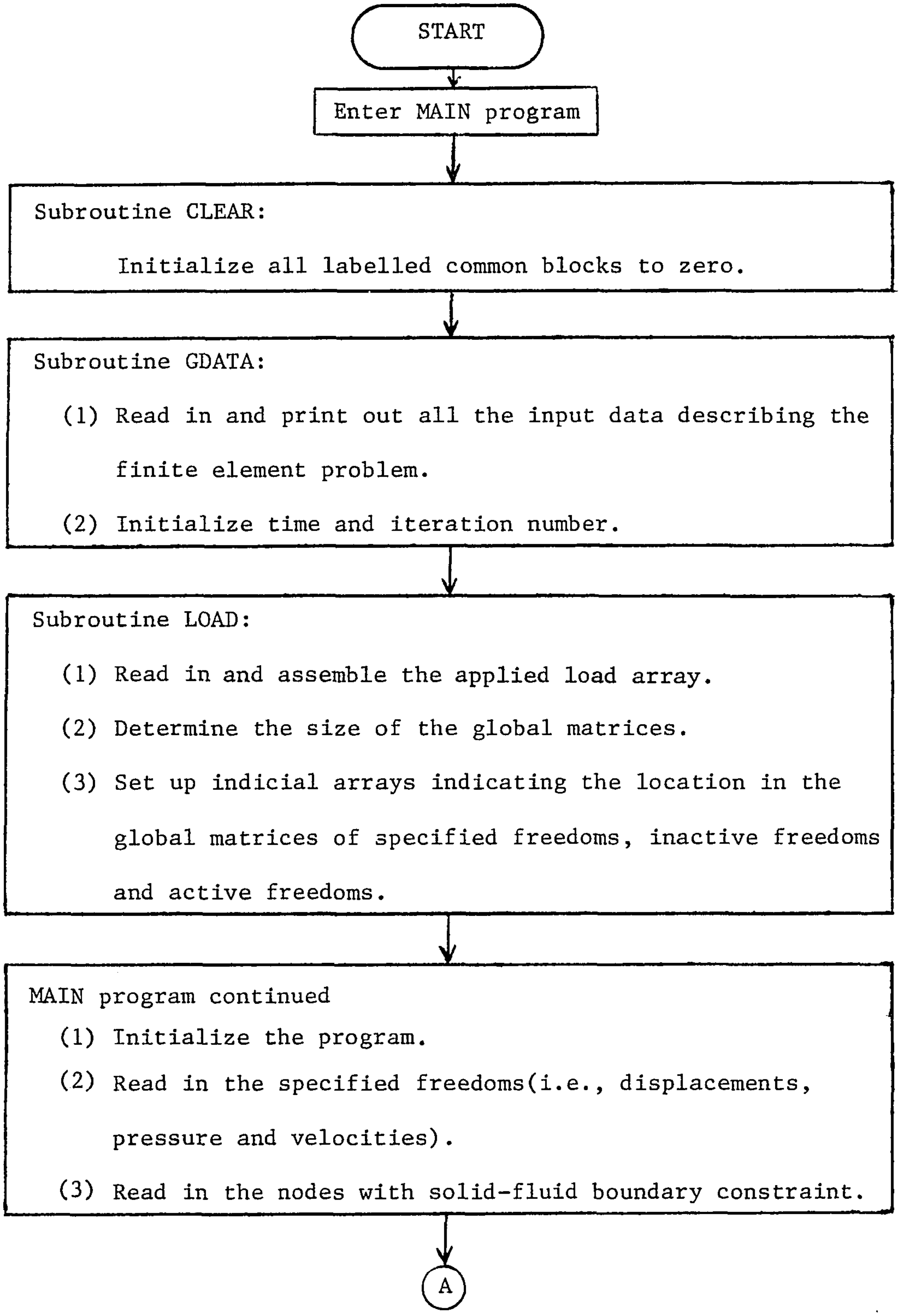


FIG. 26 SIMPLIFIED FLOWCHART OF FASINT PROGRAM

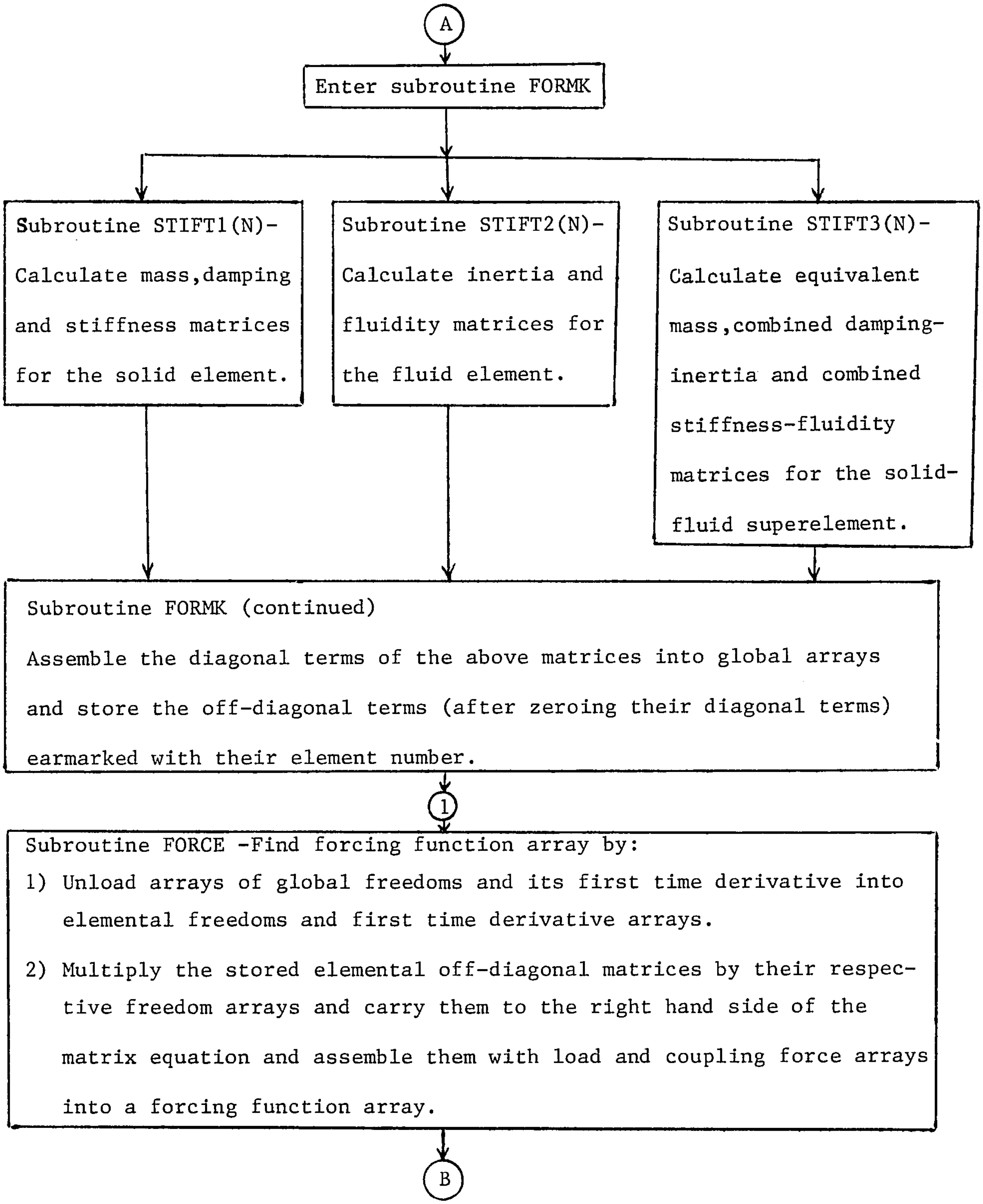


FIG. 26 SIMPLIFIED FLOWCHART OF FASINT PROGRAM (CONTINUED)

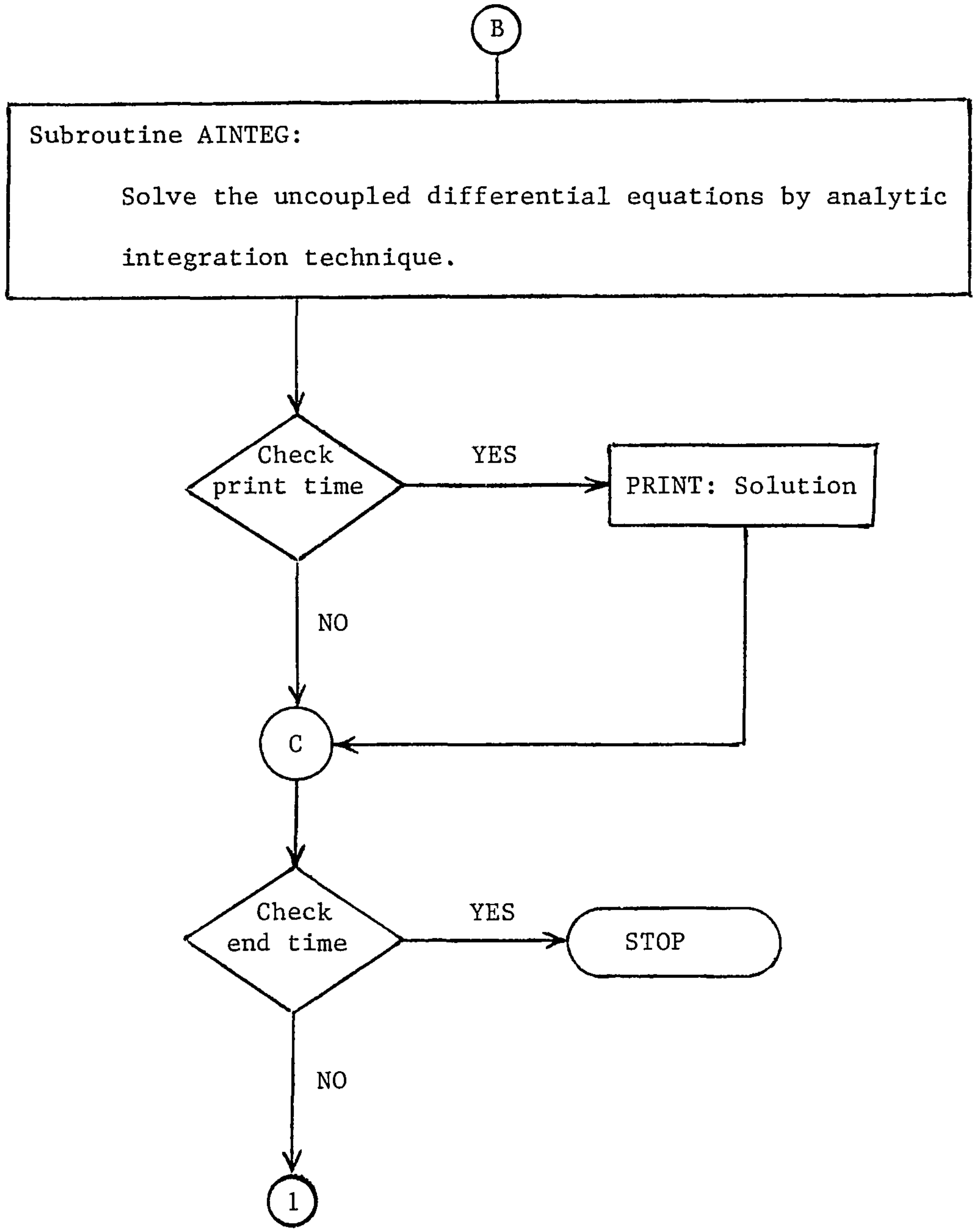


FIG. 26 SIMPLIFIED FLOWCHART OF FASINT PROGRAM (CONTINUED)

1.1 MAIN Program

The MAIN program controls the calling sequence of subroutines, initialization of the problem,insertion of the specified freedoms and the normal termination of the program. The solution is obtained in an iterative loop based on the elapsed time of the problem. The program normally terminates when the iteration number(ITIME) is equal to the offline storage controller(IITIME) or the elapsed problem time is equal or greater than the final time.

1.2 CLEAR Subroutine

CLEAR subroutine initializes all variables in a labelled COMMON block equal to zero. Its calling sequence is

CALL CLEAR (AMEMBR, LENGTH)

in which AMEMBR is the name of the first variable in the COMMON block and LENGTH is the total number of variables occupying the COMMON block[15]. For example, if a COMMON block is of the form

COMMON/ITER/DIADIT(270),DIAK1(270)

the calling sequence for CLEAR would be

CALL CLEAR(DIADIT,540)

1.3 GDATA Subroutine

GDATA reads in and prints out all the input data describing the finite element problem. The data which are read in are described in section 2, DESCRIPTION OF INPUT DATA. GDATA also initializes the elapsed time and iteration number. Its calling sequence is

CALL GDATA

1.4 LOAD Subroutine

This subroutine reads in and assembles the global applied load array and reads in the specified displacements, pressure and velocities. LOAD also functions as the system initialization routine. It determines the size of the global matrices and further sets up indicial arrays indicating the location in the global matrix of specified freedoms, inactive freedoms, and active freedoms for analytical solution. The calling statement for LOAD is

CALL LOAD

1.5 FORMK Subroutine

Subroutine FORMK is entered only in the first iteration. Its function is to find and assemble the diagonal terms of mass, damping, stiffness, inertia, volumetric fluidity, combined damping-inertia and combined stiffness-volumetric fluidity matrices into global arrays and store their off-diagonal terms (after zeroing their diagonal terms) earmarked with their element number. FORMK calls the subroutines STIFT1, STIFT2 and STIFT3 to calculate the solid, fluid and solid-fluid matrices as described later. If the coupling matrix is to be calculated, FORMK calls STIFT3 which in turn calls SF Function to obtain this matrix. The calling sequence for this subroutine is

CALL FORMK

1.6 STIFT1(N) Subroutine

Subroutine STIFT1(N) is called by FORMK to calculate the mass, damping and stiffness matrices for solid plane triangular finite element N. The calling statement for this subroutine is

CALL STIFT1(N)

in which the argument N is the solid element number.

1.7 STIFT2(N) Subroutine

Subroutine STIFT2(N) is called by FORMK to calculate the inertia and fluidity matrices for the fluid plane linear triangular finite element N. The calling statement for this subroutine is

CALL STIFT2(N)

in which the argument N is the fluid element number.

1.8 STIFT3(N) Subroutine

Subroutine STIFT3(N) is called by FORMK to calculate the combined damping-inertia and combined stiffness - fluidity matrices for the superelement plane linear quadrilaterial solid-fluid finite element N. STIFT3(N) is also entered from FORMK to calculate the solid-fluid coupling matrix. The calling statement for this subroutine is

CALL STIFT3(N)

in which the argument N is the solid-fluid superelement number.

1.9 SF Function Subroutine

SF function is called by subroutine STIFT3 to calculate the terms of the 6 x 3 solid-fluid coupling matrix according to equations (D-13),(D-15) and (D-18). The subscripts I and J in these equations are indicial node numbers which refers to the node of the element being considered,i.e., first, second, or third. I refers to the solid nodes of the solid-fluid superelement, and J refers to the fluid nodes. This function is basically similar to the one presented by Pedrido [15].

1.10 FORCE Subroutine

FORCE prepares matrices for analytic solution by unloading global displacements, pressure, and velocities and their first and second time derivatives arrays into elemental arrays and multiplying the off-diagonal terms of matrices by these elemental arrays and assembling them with load and coupling force arrays into a generalized force array. The calling statement for this subroutine is

CALL FORCE

1.11 AINTEG Subroutine

AINTEG solves the uncoupled global matrix differential equations which results from the application of FORMK and FORCE subroutines described earlier , by analytic integration technique. Nodal displacement, pressure and velocities and their first and second time derivatives are determined analytically as function of time. The solution can be obtained for inviscid, sligthly viscous and highly viscous or undamped and highly damped cases. The calling statement for this subroutine is

CALL AINTEG

2.DESCRIPTION OF INPUT DATA

In this section,the input data for FASINT is described in detail,followed by the printout of a sample data deck. The format for each card description is as follows:

Card number. Title of Card

Purpose of Card

Format

Columns Variable Comments

FASINT Input Data

1. Title Card

This card contains a 48-character descriptive title which is printed as a heading for the output data.

Format(12A4)

Columns	Variable	Comments
1-48	TITLE	If no title is desired,insert a blank card.

2. Restart Card

IRUN,Restart Switch. This quantity controls the initialization of the program. When IRUN=0,program will start from initial condition. When IRUN=1,program will skip the initialization and will use previous output already stored on tape as initial values. The latter tape storage is achieved by setting IEND=1 in the previous run as discussed next.

IEND,Restart Controller. This quantity controls the output at the completion of a dynamic run. When IEND = 1,the final program output will be stored on tape for restarting the run at a later date. When IEND = 0 ,no storage of the final output will be made.

Format(2I5)

Columns	Variable	Comments
1-5	IRUN	Restar switch
5-10	IEND	Restart controller

3. Program Control Card

This card reads two parameters. The first parameter,designated as offline storage controller(IITIME),controls the offline storage of output data for subsequent restart. At the appropriate iteration,indicated by IITIME, the output are stored on a disc, and recalled in the restart mode. The second parameter is the maximum number of iteration(LMAX) which may be needed for convergence of pressure and velocities.

Format(2I5)

Columns	Variable	Comments
1-5	IITIME	Offline storage controller
5-10	LMAX	Maximum number of iterations for convergence of pressure and velocities

4. Convergence Error Card

This card reads in the convergence error for convergence of pressure and velocities.

Format(I5)

Columns	Variable	Comments
1-5	ERRMAX	Convergence error

5. System Parameters Card

The data describing the system parameters, such as the number of nodes, elements, etc., are provided on this card.

Format(10I8)

Columns	Variable	Comments
1-8	NP	Number of nodes; $\leq$ 54
9-16	NE	Number of elements; $\leq$ 72
17-24	NB	Number of nodes with applied loads; NB $\geq$ 1
25-32	NLD	Not used
33-40	NMAT	Number of materials, $\leq$ 2
41-48	I1	Debug option, 0 $\leq$ I1 $\leq$ 3
		= 0 yields solution only = 3 yields maximum debug printout[1]
49-56	NPRINT	Frequency of printout; $\geq$ 1
57-64	ND	Number of nodes with specified displacements, pressure and velocities; 1 $\leq$ ND $\leq$ 22
65-72	NDF	Number of degrees of freedom of the system
		= 3 for fluid system
		= 2 for solid system
		= 5 for solid-fluid system[2]

[1] In view of the voluminous amount of data resulting when I1 = 3, it is suggested that the user limit the duration of his runs when using a high debug option (I1 $\geq$ 2).

[2] Care must be taken that the product NP x NDF, which represents the total global number of degrees of freedom, does not exceed 270.

Columns	Variable	Comments
73-80	NCON	Number of nodes with specified solid-fluid boundary constraints; 1< NCON < 7.

6. Time Card

This card reads in the parameters associated with the time history duration, as well as the angle between the solid-fluid boundary and the x-axis.

Format(I10,3F10.6)

Columns	Variable	Comments
1-10	NIT	Number of iterations.
11-20	TBEG	Initial time of time history.
21-30	TEND	Final time of time history.
31-40	ALPHA	Angle between the solid-fluid boundary and the x-axis,measured counter-clockwise, radians.

7. Material Properties Card

This card reads in the material properties for the number of materials specified in NMAT. One card is needed for each material.

Format(I10,3E12.5)

Columns	Variable	Comments
1-10	N	Material type number; $\leq$ 2
11-22	ORT(N,1)	= E,Young's modules for the solid.
		= c,Speed of sound for the fluid.
23-34	ORT(N,2)	= ν,Poisson's ratio for the solid.
		= k_f,Viscous damping coefficient for the fluid.
35-46	ORT(N,3)	= ρ,Density for the solid or the fluid.

8. Damping Parameter Card

This card reads in the value of proportional damping to be used in the problem.

Format(E12.5)

Columns	Variable	Comments
1-12	DAMP	= 0 for undamped system.
		= Nonzero for damped system.

9. Nodal Point Data Card

This card reads in the x and y coordinates of each nodal point. There must be one card for every node.

Format(I10,2F10.3)

Columns	Variable	Comments
1-10	N	Node number.
11-20	CORD(N,1)	x-coordinate of node.
21-30	CORD(N,2)	y-coordinate of node.

10. Nodal Point Freedom Card

This card contains the number of degrees of freedom which exist at each node. There must be one card for each node.

Format(2I5)

Columns	Variable	Comments
1-5	N	Node number.
6-10	NTYPE(N)	Active degrees of freedom at node N.

11. Element Connections Card

This card contains the node number associated with each element and the element material type number (see card 7). Element connections should be numbered in a consistant counter-clockwise fashion. Solid-fluid elements are numbered listing the solid nodes first, and then the fluid nodes.

Format(8I5)

Columns	Variable	Comments
1-5	N	Element number.

Columns	Variable	Comments
6-10	NOP(N,1)	Nodes of fluid or solid finite element or
11-15	NOP(N,2)	solid nodes of solid-fluid finite element.
16-20	NOP(N,3)	
20-25	NOP(N,4)	Fluid nodes of solid-fluid finite element;
26-30	NOP(N,5)	blank for fluid or solid elements.
31-35	NOP(N,6)	
36-40	IMAT(N)	Material type number for finite element N.
		= 1 for totally solid problem.
		= 2 for totally fluid problem.
		For interactive problem.
		= 1 for solid element.
		= 2 for fluid element.
		= 3 for solid-fluid element.

12. Applied Load Type Card

This card contains the nodes at which applied loads act and the type of load which acts there. Even if no applied loads exist, a dummy load of zero must be applied at some arbitrary node. One card is needed for each applied load.

Format(2I5)

Columns	Variable	Comments
1-5	NBC	Node at which specified load acts.
6-10	NFIX	Code indication of direction of load
		= 01 load is in y-direction.
		= 10 load is in x-direction.
		= 11 load is in x and y-directions.
		= 0 dummy load.

13. Specified Freedom Type Card

This card contains the nodes at which specified freedoms (i.e., displacements, pressure and velocities)act and the type of specified freedoms which act there. At least one specified freedom must be given in order to restrain the system and remove rigid body modes. If a specified displacement and a specified pressure and velocity exist at a node, that node must be indicated twice, once for the specified displacements and once for the pressure and velocity.

Format(2I5)

Columns	Variable	Comments
1-5	NSD	Node at which specified freedoms act.
6-10	NDFIX	Code indication of direction of specified freedom. = 01 displacement in y-direction is given = 10 displacement in x-direction is given = 11 displacements in x and y-directions are given = 200 pressure is given = 20 velocity in x-direction is given = 2 velocity in y-direction is given = 22 velocities in x and y-directions are given = 220 pressure and x-direction velocity are given = 202 pressure and y-direction velocity are given = 222 pressure, x and y-direction velocities are given

14. Solid-Fluid Boundary Constraint Nodes

This card reads in the solid-fluid boundary constraint nodes in order to inforce eq.(29). There must be one card for every node.

Format(I5)

Columns	Variable	Comments
1-5	NSCON	Solid-fluid boundary constraint node.

15. Load Card

This card reads in the load information corresponding to card No. 12.

Format(F10.4)

Columns	Variable	Comments
1-10	PMAX	Magnitude of load.

16. Specified Freedom Card

This card contains the specified freedoms (i.e., displacements. pressure and velocities). One card is needed for each set of specified displacements and pressure and velocities corresponding to card No. 13.

Format(3F10.3)

Columns	Variable	Comments
1-10	DISP(N,1)	Specified pressure or specified displacement in x-direction.
11-20	DISP(N,2)	Specified velocity in x-direction or specified displacement in y-direction.
21-30	DISP(N,3)	Specified velocity in y-direction.

```
****************   THIS IS A PRINTOUT OF A FASINT DATA DECK   ******************
72 ELEMENT FLUID-SOLID MODEL
      .01
    0    1
55000   30
        54      72      1      3      2      0     700     21     5     7
         0       0
      56000        0.        .002        0.
          1       .3E+08      .3E+00    .7297E-03
          2       .5E+05      0.0E+00  .0000935521
           0.
          1        .5          0.
          2        .5          4.
          3        .6          8.
          4        .5         12.
          5        .5         12.25
          6        .6         12.5
          7       2.7          0.
          8       2.7          4.
          9       2.7          8.
         10       2.7         12.
         11       2.7         12.25
         12       2.7         12.5
         13       6.9          0.
         14       6.9          4.
         15       6.9          8.
         16       6.9         12.
         17       6.9         12.25
         18       6.9         12.5
         19      11.1          0.
         20      11.1          4.
         21      11.1          8.
         22      11.1         12.
         23      11.1         12.25
         24      11.1         12.5
         25      15.3          0.
         26      15.3          4.
         27      15.3          8.
         28      15.3         12.
         29      15.3         12.25
         30      15.3         12.5
         31      19.5          0.
         32      19.5          4.
         33      19.5          8.
         34      19.5         12.
         35      19.5         12.25
         36      19.5         12.5
         37      23.7          0.
         38      23.7          4.
         39      23.7          8.
         40      23.7         12.
         41      23.7         12.25
         42      23.7         12.5
         43      27.9          0.
         44      27.9          4.
         45      27.9          8.
         46      27.9         12.
         47      27.9         12.25
         48      27.9         12.5
         49      30.           0.
         50      30.           4.
```

	51		30.	3.	
	52		30.	12.	
	53		30.	12.25	
	54		30.	12.5	
1	3				
2	3				
3	3				
4	5				
5	2				
6	2				
7	3				
8	3				
9	3				
10	5				
11	2				
12	2				
13	3				
14	3				
15	3				
16	5				
17	2				
18	2				
19	3				
20	3				
21	3				
22	5				
23	2				
24	2				
25	3				
26	3				
27	3				
28	5				
29	2				
30	2				
31	3				
32	3				
33	3				
34	5				
35	2				
36	2				
37	3				
38	3				
39	3				
40	5				
41	2				
42	2				
43	3				
44	3				
45	3				
46	5				
47	2				
48	2				
49	3				
50	3				
51	3				
52	5				
53	2				
54	2				
1	1	7	3		2
2	1	8	2		2
3	2	3	[illegible]		2

4	2	9	3				2
5	3	9	10				2
6	4	10	11	3	10	4	3
7	4	11	5				1
8	5	11	12				1
9	5	12	6				1
10	7	13	14				2
11	7	14	8				2
12	8	14	15				2
13	8	15	9				2
14	9	15	16				2
15	10	16	17	9	16	10	3
16	10	17	11				1
17	11	17	18				1
18	12	18	12				1
19	13	19	20				2
20	13	20	14				2
21	14	20	21				2
22	14	21	15				2
23	15	21	22				2
24	16	22	23	15	22	16	3
25	16	23	17				1
26	17	23	24				1
27	17	24	18				1
28	19	25	26				2
29	19	26	20				2
30	20	26	27				2
31	20	27	21				2
32	21	27	28				2
33	22	28	29	21	28	22	3
34	22	29	23				1
35	23	29	30				1
36	23	30	24				1
37	25	31	32				2
38	25	32	26				2
39	26	32	33				2
40	26	33	27				2
41	27	33	34				2
42	28	34	35	27	34	28	3
43	28	35	29				1
44	29	35	36				1
45	29	36	30				1
46	31	37	38				2
47	31	38	32				2
48	32	38	39				2
49	32	39	33				2
50	33	39	40				2
51	34	40	41	33	40	34	3
52	34	41	35				1
53	35	41	42				1
54	35	42	36				1
55	37	43	44				2
56	37	44	38				2
57	38	44	45				2
58	38	45	39				2
59	39	45	46				2
60	40	46	47	39	46	40	3
61	40	47	41				1
62	41	47	48				1
63	41	48	42				1
64	43	49	50				2

```
55    43    50    44                          2
56    44    50    51                          2
67    44    51    45                          2
59    45    51    52                          2
69    45    52    53    45    52    45        3
70    45    53    47                          1
71    47    53    54                          1
72    47    54    43                          1
 1     0
 1   202
 2   200
 3   200
 4   202
 4    11
 5    11
 6    11
 7     2
13     2
19     2
25     2
31     2
37     2
43     2
49   202
50   200
51   200
52   202
52    11
53    11
54    11
10
16
22
28
34
40
46
       0.
      10.                   0.
      10.
      10.
      10.                   0.
       0.          0.
       0.          0.
       0.          0.
                            0.
                            0.
                            0.
                            0.
                            0.
                            0.
                            0.
       0.                   0.
       0.
       0.
       0.                   0.
       0.          0.
       0.          0.
       0.          0.
************************    END OF SAMPLE DATA DECK    ************************
```

3. DESCRIPTION OF OUTPUT DATA

The output data are classified as follows:

3.1 Output data not under debug option control

The output data which are not under the control of the debug option I1 control are as follows:

Subroutine	Output
GDATA	Geometric and properties input data,the other parameters defined in the description of input data.
LOAD	Global degrees of freedom;specified freedoms; load input data; indicial arrays; global load array.
AINTEG	Time parameters such as time,iteration number,etc; numerical values of the results.

All the above data are printed out with suitable headings regardless of the debug printout discussed hereunder.

3.2 Output data under the control of the debug option

When I1 $\geq$ 1, the following data is printed out.

Subroutine	Output
FORMK	Global arrays of assembled diagonal terms of matrices.
FORCE	Forcing function array.
AINTEG	Global arrays of displacements,pressure and velocities and their corresponding first and second derivatives excluding the specified freedoms; intermediate parameters such as damped frequency,natural frequency, etc.

When Il $\geq$ 2, the following data is printed out.

Subroutine	Output
FORCE	Temporary arrays of elemental displacements,pressure and velocities and their corresponding first and second derivatives and indicial array; temporary elemental force array resulting from the off-diagonal terms of elemental mass, stiffness, inertia, fluidity, combined damping-inertia or combined stiffness-fluidity matrices.

When Il $\geq$ 3, the following data is printed out.

Subroutine	Output
FORMK	Elemental indicial array; elemetal off-diagonal terms of damping,stiffness,inertia,fluidity,combined damping-inertia,or combined stiffness-fluidity matrices.
STIFT1	Local coordinates; element connections; elemental mass, damping and stiffness matrices; area of finite element, etc.
STIFT2	Local coordinates; element connections; elemental inertia and fluidity matrices; area of finite element; etc.
STIFT3	Local coordinates,element connections,and areas of solid portion and fluid portion of quadrilateral solid-fluid element; elemental equivalent mass,combined damping-inertia and stiffness-fluidity matrices; elemental coupling matrices.

All of the above data are printed out with suitable headings. A working knowledge of the program is necessary to perform effective debugging.

3.3 Error Prinouts

If the user does not number either the nodes in an orderly fashion or the element connections in a consistently counter-clockwise manner,the following printout will occur.

Subroutine	Output
STIFT1,2 or 3	Number of element with bad connections - run is terminated.

3.4 Output Data Defining the Problem Solution

The program first prints out the heading " Finite Element Analytic Integration Solution " followed by the elapsed time, time increment, iteration number, printout iteration number, frequency of printout,and time at first iteration. The node number, freedom number and the numerical value of the result are then printed out as the problem solution.

4 OPERATING PROCEDURE

To employ the FASINT computer program effectively,it is recommended that the user familiarizes himself with the formulation of finite element model and its numerical solution,presented in Part 1 and 2 as well as the organization of the program described in Part 3,before attempting to solve a solid-fluid interaction problem using FASINT.

In developing the finite element mesh the user should number the nodes in an orderly fashion and the element connections in a consistently counterclockwise manner. Failure to do so would result in a bad connection error message printout and abnormal termination of the run. Quadrilateral solid-fluid elements are numbered starting with the solid nodes first and then the fluid nodes. The input data is then punched and prepared in accordance with the format and description presented in Part 3 , Section 2, DESCRIPTION OF INPUT DATA. These input data are then punched on cards and assembled with with FASINT source deck and the appropriate control cards. A sample of the deck assembling cards for UNIVAC SERIES 70 is given below:

```
/LOGON userid,acct. no. TIME=in second
/PARAM LIST=YES,MAP=YES,DEBUG=YES
/EXEC BGFOR
      PROGRAM NAME
      ( FASINT Source Statement)
      END
```

```
/FILE FASINT.RESTART1,LINK=DSET24,FCBTYPE=SAM[1]
/FILE FASINT.RESTART2,LINK=DSET25,FCBTYPE=SAM[2]
/EXEC *
      (FASINT data deck)
/LOGOFF
```

The following procedure is recommended for effective use of the FASINT program.

1) Input data check run

The user should make a run with the beginning time and final time of the run set equal to zero. The debug option should also be set equal to zero. The program will then print out all of the input data and all other problem data, such as sorted global load array,up to the problem solution. This enables the user to check the input data and the intermediate results for correctness. If the input data is incorrect,this run should be repeated until all user errors have been rectified. If the intermediate results appear incorrect,the program may be run with a higher debug option for several integration time steps to provide more debug printout to help the user determine the nature of the difficulty.

2) Test run

After the data has been debugged, the user may perform

---

[1] RESTART1 designates the input offline storage

[2] RESTART2 designates the output offline storage

a dynamic run using a time step on the basis of Section 4 in part one and part two.

3) Finite element mesh convergence run

The convergence of the finite element mesh is determined by making a run having a finer grid(more elements) than the desired grid. Convergence is achieved when there is no appreciable difference between the runs.

4) Time step convergence run

The convergence of the time step is determined by making a run having a time step that is twice the desired one. Convergence is reached when there is no appreciable difference between the runs. If there is a discrepancy,the time step should be halved and this run repeated until convergence is established.

5) Restart Capability

Restart switch,IRUN, controls the initialization of the program. When IRUN = 0,program will start from initial condition. When IRUN = 1,program will skip the initialization and will use previous output already stored on tape as initial values. The latter tape storage is achieved by setting IEND=1 in the previous run as discussed next. Restart controller, IEND,controls the output at the completion of a dynamic run. When IEND = 1, the final program output will be stored on tape for restarting the run at a later date. When IEND = 0, no storage of the final output will be made. This procedure may be used for any desired offline storage controller iteration number, IITIME, at which the output will be stored on tape for restarting the run at a later date.

The runs made for the completion of part one required 68k bytes of storage and was executed on the INTERDATA 7/32 mini-computer. The computations for part two required 318k bytes of storage and was executed on IBM 370/168 digital computer.

5. FASINT NOMENCLATURE

The variables used in FASINT are listed, defined, and cross-referenced with the analysis notation, to aid the user in understanding the program.

5.1 Subscripted Variables

Program Notation	Analysis Notation	Subroutine	Description
AO(3),AM(3)	a	STIFT2,STIFT3	Local area coordinates of fluid and solid triangles,respctively, in^2.
AK(20,20)	$[K_e]$,$[H_e]$, $[\bar{K}_e]$	STIFT1,STIFT2, STIFT3	In STIFT1,stiffness matrix In STIFT2, fluidity matrix In STIFT3,combined stiffness-fluidity matrix
BO(3),BM(3)	b	STIFT1,STIFT2, STIFT3	Local y-coordinate of fluid and solid triangles,respectively, in.
CO(3),CM(3)	c	STIFT1,STIFT2, STIFT3	Local x-coordinate of fluid and solid triangles,respectively, in.
CORD(54,2)	x,y	GDATA	Array of nodal coordinates, in.
DISPL(20)	$\{Z_e\}$,$\{X_e\}$	FORCE	Array of elemental freedoms(i.e., pressure, x-component of velocity and y-component of velocity, and displacements).

Program Notation	Analysis Notation	Subroutine	Description
DISP(22,3)	-	LOAD	Matrix of specified freedoms (i.e., displacements, pressure and velocities).
GM(20,20)	$[M_e],[\bar{M}_e]$	STIFT1, STIFT3	In STIFT1,mass matrix In STIFT3,equivalent mass matrix
DIAAC(270)	$\{\ddot{X}\}$	FORCE,AINTEG	Updated array of global freedoms second time derivative
DIAACC(270)	$\{\ddot{X}\}_o$	FORCE,AINTEG	Array of global freedoms(i.e., displacements, pressure and velocities) second time derivative.
DIADI(270)	$\{\bar{X}\}$	FORCE,AINTEG	Updated array of global freedoms (i.e., displacements, pressure and velocities).
DIADIS(270)	$\{\bar{X}\}_o$	FORCE,AINTEG	Array of global freedoms(i.e., displacements, pressure and velocities).
DIAVE(270)	$\{\dot{\bar{X}}\}$	FORCE,AINTEG	Updated array of global freedoms (i.e., displacements, pressure and velocities) first time derivative.
DIAVEL(270)	$\{\dot{\bar{X}}\}_o$	FORCE,AINTEG	Array of global freedoms(i.e., displacements, pressure and

Program Notation	Analysis Notation	Subroutine	Description
			velocities) first time derivative
DIAGC(270)	$\lceil\bar{C}'\rfloor$	FORCE,AINTEG	Diagonal matrix with diagonal terms equal to diagonal terms of $[\bar{C}]$.
DIAGK(270)	$\lceil\bar{K}'\rfloor$	FORCE,AINTEG	Diagonal matrix with diagonal terms equal to diagonal terms of $[\bar{K}]$.
DIAGM(270)	$\lceil\bar{M}'\rfloor$	FORCE,AINTEG	Diagonal matrix with diagonal terms equal to diagonal terms of $[\bar{M}]$.
DIAK(270)	$\{\bar{F}'\}_o$	FORCE,AINTEG	Forcing function array.
DIAK1(270)	$\{\bar{F}'\}$	FORCE,AINTEG	Updated forcing function array.
FORCEC(20)	$[\bar{C}''_e]\cdot\{\dot{X}_e\}$	FORCE	Elemental forcing function array resulting from the multiplication of the stored elemental off-diagonal damping, inertia or combined damping-inertia matrices by their respective freedom array.
FORCEK(20)	$[\bar{K}''_e]\cdot\{X_e\}$,	FORCE	Elemental forcing function array resulting from the multipilcation of the stored elemental off-diagonal stiffness, fluidity or combined stiffness-fluidity matrices by their respective freedom array.
IMAT(72)	-	GDATA,FORMK, STIFT1,2 and 3	Array of element material type numbers.
INACT(115)	-	LOAD	Indicial array of rows which contain inactive freedoms.
INDEX(32)	-	LOAD	Indicial array of rows at which specified freedoms act.

Program Notation	Analysis Notation	Subroutine	Description
NACT(32)	-	LOAD	Indicial array of active row numbers in sequence.
NBC(1)	-	GDATA,LOAD	Array of nodes at which specified loads act.
NDFIX(22)	-	GDATA,LOAD	Array of code indicators of direction of specified freedoms.
NFIX(1)	-	GDATA,LOAD	Array of code indicators of direction of applied loads.
NOP(72,6)	-	STIFT1,2 and 3	Matrix of element connections.
NSCON(7)	-	GDATA	Nodes with solid-fluid boundary constraints.
NSD(22)	-	GDATA,LOAD	Array of nodes at which specified freedoms act.
NSDF(270)	-	AINTEG	Indicator of specified freedoms: = 0 for free freedoms = 1 for specified freedoms
NTYPE(54)	-	GDATA,FORMK	Array of nodal degrees of freedom.
ORT(2,3)	-	GDATA,STIFT1, STIFT2,STIFT3	Matrix of material properties.
R(5)	$\{R'_e\}$	LOAD	Elemental applied load array.
R4(270)	$\{R'\}$	LOAD,FORCE	Array of applied loads.
S(6,3)	$[S_e]$	STIFT3	Elemental solid-fluid coupling matrix.
STOREC(20,20,72)	$[\bar{C}''_e]$	FORMK,FORCE	Damping, inertia or combined damping-inertia matrices with zeroes along their diagonals.

Program Notation	Analysis Notation	Subroutine	Description
STOREK(20,20,72)	$[\bar{K}''_e]$	FORMK,FORCE	Stiffness,fluidity, and combined stiffness-fluidity matrices with zeroes along their diagonal.
VELOC(20)	$\{\dot{X}_e\}$	FORCE	Array of elemental freedoms(i.e., displacements, pressure and velocities) first time derivative.
XM(20,20)	$[C_e]$, $[L_e]$, $[\bar{C}_e]$	STIFT1,STIFT2, STIFT3	In STIFT1, elemental damping matrix. In STIFT2, elemental inertia matrix. In STIFT3, elemental combined damping-inertia matrix.

5.2 Nonsubscripted Variables

Program Notation	Analysis Notation	Subroutine	Description
ALPHA	α	STIFT3	Angle of solid-fluid boundary with longitudinal x-axis, radians.
AH	$\tilde{h}$	STIFT3	Constant value of x or y used in the computation of solid-fluid coupling matrix for vertical or horizontal wall, respectively.
AREAO,AREAM	A_f, A_s	STIFT1,2 and 3	Area of fluid and solid triangles respectively.
DAMP	k_s	STIFT1,STIFT3	Damping factor based on mass, sec^{-1}.
DELT	Δt	FORMK	Time step, sec.
DOF	-	LOAD	Degrees of freedom.
ERRMAX	-	FORCE	Convergence error.
I1	-	GDATA	Debug option.
IEND	-	GDATA	Restart controller.
IITIME	-	GDATA	Off-line storage controller.
I,J,K	-	STIFT1,2 and 3	Element connections.
IRUN	-	GDATA	Restart switch.
ITIME	-	GDATA	Iteration number.
LMAX	-	GDATA	Maximum number of iterations for convergence of freedoms.
MDF	-	FORMK	Dummy degree of freedom.
NB	-	GDATA,LOAD	Number of nodes with applied loads.
NCN	-	MAIN	Number of nodes per element.
NCON	-	GDATA	Number of nodes with specified solid-fluid boundary constraints.
NQ	-	LOAD	Node at which load acts.
ND	-	GDATA,LOAD	Number of nodes with specified freedoms.

Program Notation	Analysis Notation	Subroutine	Description
NDF	-	GDATA,SF	Nodal degree of freedom.
NE	-	GDATA,FORMK	Number of elements.
NIT	-	GDATA	Number of iterations.
NLD	-	GDATA	Not used.
NMAT	-	GDATA	Number of material types.
NP	-	GDATA,SF	Number of nodes.
NPRINT	-	GDATA	Frequency of printout.
NSZF	-	SF,GDATA	Size of global arrays.
NTIME	-	FORMK	Printout control parameter.
T	t	FORMK	Elapsed problem time, sec.
TBEG	-	GDATA	Time at beginning of time history.
TEND	-	GDATA	Time at end of time history.
T1	-	FORMK	Time at first iteration.

6. PROGRAM LISTING AND SAMPLE RUN

In this section, the FORTRAN program listing of FASINT together with the output of a sample run for an undamped system are presented. The model being used to illustrate the behaviour of the program is a 72 element solid-fluid model, similar to Fig. 13 . To reduce the bulk of computer printout, the output data are supplied for iteration number 0, 1, and 700 corresponding to 0.00000E+00, 0.35714E-07 and 0.24999E-04 seconds, respectively.

```
FORTRAN IV G LEVEL  21              MAIN              DATE = 78202         14/53/39              PAGE 0001
                C        CONTROL MAIN PROGRAM
      0001               COMMON/S4B/TITLE(12),ORT(2,3),NSCON(7),CORD(54,2),NOP(72,5)
                        1,IMAT(72),NBC(1 ),NFIX(1 ),NSD(22),NDFIX(22),NTYPE(54)
      0002               COMMON/L4B/R(5),R4(270),DISP(22,3),INDEX(32 ),INACT(115),NACT(140)
      0003               COMMON/MAT/AK(20,20),XM(20,20),SM(20,20)
      0004               COMMON/FUNCT/YF,YI,A0(3),AM(3),B0(3),B4(3),C0(3),CM(3),A4,AREA0
                        1,AREAM,KI,KF,SLOPE,Y0
      0005               COMMON/S4AT/STOREK(20,20,72),STOREC(20,20,72)
      0006               COMMON/DDD/DISPL(20),FORCEC(20),FORCEK(20),DIAACC(270)
                        1,DIADIS(270),DIAVEL(270),DIAGC(270),DIAGK(270),DIAK(270)
                        2,VELOC(20),DIAD(270),DIAVE(270),DIAAC(270),ACCEL(20),DIAGM(270)
      0007               COMMON/S4B/S(3,5)
      0008               COMMON/ITER/DIADIT(270),DIAKI(270)
      0009               COMMON/THREE/LS(20,72)
      0010               COMMON/DOE/NSDF(270),NSCON1(7),NSCON2(7)
      0011               COMMON/PARAM/T,ITIME,TEND,NCN,       IRUN,IEND,IITIME,LMAX
                        1,NP,NE,NB,NLD,NMAT,II,NPRINT,ND,NDF,NIT,TBEG,NCON ,ALPHA,
                        2,NSZF,NTIME,DELT,TI,          DAMP,ERRMAX
      0012               DIMENSION MROW(3)
      0013               DATA MROW(1),MROW(2),MROW(3)/5,9,20/
      0014          720  FORMAT(44H FINITE ELEMENT ANALYT. INTEGRATION SOLUTION/26H SOLUTIO
                        1N FOR TIME T=,E12.5/23H TIME INCREMENT DELT=,E12.5/21H ITERATION N
                        2UMBER IT=,I5/21H PRINTOUT VARIABLE N=,I5/13H OUTPUT EVERY,I5,2X,5H
                        3TIMES/27H TIME AT FIRST ITERATION IS,E12.5/12H DEBUG LEVEL,I4)
      0015         5000  FORMAT(1H ,14X,13HU X-DIRECTION,9X,13HU Y-DIRECTION,14X,8HPRESSURE
                        1,13X,13HV X-DIRECTION,9X,13HV Y-DIRECTION/1H ,2X,4HNODE,5(6X,3HDOF
                        2,6X,5HVALUE,3X))
      0016         5001  FORMAT(3X,I4,5(I3,2X,E12.5))
      0017         5003  FORMAT(1H1)
                C        ZERO OUT EVERYTHING IN COMMON
      0018               CALL CLEAR(TITLE,737)
      0019               CALL CLEAR(R,523)
      0020               CALL CLEAR(AK,1200)
      0021               CALL CLEAR(YF,27)
      0022               CALL CLEAR(STOREK,57600)
      0023               CALL CLEAR(DISPL,2800)
      0024               CALL CLEAR(S,I6)
      0025               CALL CLEAR(DIADIT,540)
      0026               DO 13 I=1,20
      0027               DO 13 J=1,72
      0028               LS(I,J)=0
      0029           13  CONTINUE
      0030               DO 15 K=1,270
      0031               NSDF(K)=0
      0032           15  CONTINUE
      0033               DO 15 I=1,7
      0034               NSCON1(I)=0
      0035               NSCON2(I)=0
      0036           15  CONTINUE
                C        INITIALIZE NUMBER OF CORNER NODE MAX.
      0037               NCN=3
                C        READ INPUT GEOMETRY AND PROP.
      0038               CALL GDATA
                C        READ LOADS
      0039               CALL LOAD
                C**************************    BEGIN MATRIX ASSEMBLY LOOP   *************
```

```
C          CHECK RESTART ENTRY
      IF( IRUN .EQ. 1 ) GO TO 10
      GO TO 12
   10 CONTINUE
C          READING AND WRITING FROM DISC
      REWIND 13
      READ(13) T,DELT,ITIME,NTIME,IPRINT,FB,II
      WRITE(6,7231)T,DELT,ITIME,NTIME,NPRINT,II,II
      NODE=0
      WRITE(6,6000)
      DO 11 IJ=1,NSOF,5
      IL=IJ+4
      READ(13)(DIADIS(K),DIAVEL(K),DIAACC(K), K=IJ,IL)
      NODE=NODE+1
      WRITE(6,6001) NODE,(K,DIADIS(K), K=IJ,IL)
   11 CONTINUE
      T=T+DELT
      ITIME=ITIME+1
C********* END OF DISC OUTPUT *********
   12 CONTINUE
C     REPEAT FOR ALL FREEDOMS
      DO 204 IJ=1,NSOF
C     INCLUDE SPECIFIED DISPLACEMENTS AND SOLID-FLUID BOUNDARY
C                       CONSTRAINTS
      DO 500 N=1,ND
      IFIX=IDFIX(N)
      KM=1
      IF(IFIX .EQ. 1 .OR. IFIX .EQ. 2) .OR. IFIX .EQ. 22) KM=2
      IF(IFIX .EQ. 2) KM=3
      INC=1
      I=NSD(N)
      IF(IDOF .EQ. 3) GO TO 40
      IF(IDOF .EQ. 5) GO TO 41
      NROWB=(I-1)*5
      GO TO 52
   40 NROWB=(I-1)*5+2
      GO TO 52
   41 NROWB=(I-1)*5
      IF(IFIX .EQ. 2 .OR. IFIX .GT. 11) NROWB=NROWB+2
   52 NDF=3
      IF(IFIX .EQ. 200 .OR. IFIX .EQ. 11) NDF=1
      IF(IFIX.EQ.20.OR. IFIX.EQ.220.OR.IFIX.EQ.1.OR.IFIX.EQ.11) NDF=2
      IF(IFIX .EQ. 202) INC=2
C     EXAMINE EACH DEGREE OF FREEDOM
      DO 490 M=KM,NDF,INC
      NROWBM=NROWB+M
      IF(IJ .NE. NROWBM) GO TO 490
      DIADIS(NROWBM)=DISP(N,M)
      NSDF(IJ)=3
  490 CONTINUE
      IDFIX(N)=IFIX
  500 CONTINUE
  204 CONTINUE
      IF(NDF .EQ. 3) GO TO 501
      DO 51 K=1,NCON
      I=NSC(I[illegible])
```

```
      NROW3=(I-1)*5
      NSIDE(IK,1)=NROW3+2
      NSIDE(IK,4)=NROW3+5
   51 CONTINUE
  501 CONTINUE
  300 CALL FORM
    1 CALL FORCE
C*********************** END MATRIX ASSEMBLY LOOP  ****************
      IF( ISTAD .EQ. 0 ) GO TO 5
      IF(ITIME .EQ. ITIME) GO TO 5
    5 CONTINUE
C             INCREMENT TIME AND ITERATION NUMBER
      T=T+DELT
      ITIME=ITIME+1
      NTIME=(ITIME/NPRINT)*NPRINT
      IF(T.GE.TEND)STOP
      GO TO 1
    6 CONTINUE
C***********  WRITING OUTPUT ON DISC  **************
      WRITE(6,5003)
      REWIND 14
      WRITE(14)T,DELT,ITIME,NTIME,NPRINT,TL,II
      WRITE(6,7200)T,DELT,ITIME,NTIME,NPRINT,TL,II
      NODE=0
      WRITE(6,5000)
      DO 110 IJ=1,NSIZE,5
      IL=IJ+4
      WRITE(14) (DIADIS(IK),DIAVEL(IK),DIAACC(IK),IK=IJ,IL)
      NODE=NODE+1
      WRITE(6,5001) NODE,(IK,DIADIS(IK), IK=IJ,IL)
  110 CONTINUE
      END FILE 14
      STOP
      END
```

FORTRAN IV G LEVEL 21 CLEAR DATE = 78232 14/53/39 PAGE 0001

```
      SUBROUTINE CLEAR(AMEMBR,LENGTH)
      DIMENSION AMEMBR(LENGTH)
      DO 10 I=1,LENGTH
   10 AMEMBR(I)=0.
      RETURN
      END
```

```
0001          SUBROUTINE SDATA
0002          COMMON/SAREA/TITLE(12),ORT(2,3),NSCON(7),CORD(54,2),NOP(72,5)
             1,IMAT(72),NBC(1 ),NFIX(1 ),NSD(22),NDFIX(22),NTYPE(54)
0003          COMMON/PARAM/T,TIMS,TEND,NCK,      IRUN,IEND,IITIME,LMAX
             1,NP,NE,NB,NLD,NMAT,II,NPRINT,ND,NDF,NIT,TBEG,NCON ,ALPHA
             2,NSZF,NTIME,DELT,TI,        DAMP,ERRMAX
       C      READ AND PRINT TITLE AND GENERAL SYSTEM PROPERTIES
0004          READ(5,7)(TITLE(I),I=1,12)
0005          WRITE(6,100)(TITLE(I),I=1,12)
0006          READ(5,8000) ERRMAX
0007          WRITE(6,8012) ERRMAX
0008          READ(5,8003)  IRUN, IEND
0009          WRITE(6,8002)  IRUN, IEND
0010          READ(5,8004) IITIME,LMAX
0011          WRITE(6,8005) IITIME,LMAX
0012          READ(5,1)NP,NE,NB,NLD,NMAT,II,NPRINT,ND,NDF,NCON
0013          WRITE(6,14)
0014          WRITE(6,1)NP,NE,NB,NLD,NMAT,II,NPRINT,ND,NDF,NCON
0015          READ(5,10)NIT,TBEG,TEND,ALPHA
0016          WRITE(6,11)
0017          WRITE(6,10)NIT,TBEG,TEND,ALPHA
       C      READ AND PRINT MATERIAL DATA
0018          READ(5,3)(N,(ORT(N,I),I=1,3),L=1,NMAT)
0019          WRITE(6,103)
0020          WRITE(6,102)
0021          WRITE(6,3)(N,(ORT(N,I),I=1,3),N=1,NMAT)
       C      READ AND PRINT DAMPING FACTOR
0022          READ(5,12)DAMP
0023          WRITE(6,101)DAMP
       C      READ NODAL POINT DATA
0024          READ(5,2)(N,(CORD(L,M),M=1,2),L=1,NP)
0025          READ(5,6)(I,NTYPE(I),I=1,NP)
       C      READ ELEMENT DATA
0026          READ(5,3)(N,(NOP(N,M),M=1,5),IMAT(N),L=1,NE)
       C      READ BOUNDARY DATA,SPECIFIED FREEDOMS AND BOUNDARY CONSTRAINTS
0027          READ(5,4)(NBC(I),NFIX(I),I=1,NB)
0028          READ(5,4)(NSD(I),NDFIX(I),I=1,ND)
0029          READ(5,5) (NSCON(I),I=1,NCON)
       C      PRINT INPUT DATA
0030          WRITE(6,102)
0031          WRITE(6,15)
0032          WRITE(6,2)(N,(CORD(N,M),M=1,2),N=1,NP)
0033          WRITE(6,13)
0034          WRITE(6,20)(I,NTYPE(I),I=1,NP)
0035          WRITE(6,103)
0036          WRITE(6,16)
0037          WRITE(6,3)(N,(NOP(N,M),M=1,5),IMAT(N),N=1,NE)
0038          WRITE(6,103)
0039          WRITE(6,17)
0040          WRITE(6,20)(NBC(I),NFIX(I),I=1,NB)
0041          WRITE(6,103)
0042          WRITE(6,17)
0043          WRITE(6,20)(NSD(I),NDFIX(I),I=1,ND)
0044          WRITE(6,105)
0045          WRITE(6,23) (NSCON(I),I=1,NCON)
       C      INITIALIZE TIME AND ITERATION NUMBER
```

```
0046                T=TBEG
0047                ITIME=0
0048                RETURN
0049          1     FORMAT(4D20)
0050          2     FORMAT(I10,2F10.3)
0051          3     FORMAT(8I5)
0052          4     FORMAT(2I5)
0053          5     FORMAT(I5)
0054          7     FORMAT(20A4)
0055          8     FORMAT(I10,3D12.5)
0056         10     FORMAT(I10,3F10.5)
0057         11     FORMAT(1H ,4X,3HNIT,6X,4HTBEG,5X,4HTEND,7X,5HALPHA)
0058         13     FORMAT(25H NODAL DEGREES OF FREEDOM/1H ,12(4HNODE,1X,4HTYPE,1X))
0059         14     FORMAT(1H ,3X,2HNP,6X,2HNE,5X,2HNB,5X,3HNLD,4X,4HNMAT,5X,2HII,5X,6
                   1HIPRINT,3X,2HND,5X,3HNDF,5X,5HNCDN)
0060         15     FORMAT(1H ,3X,4HNODE,5X,5HX-CORD,4X,5HY-CORD)
0061         16     FORMAT(1H ,10X,5HNODES,9X,9HMATL. NO.)
0062         17     FORMAT(1H ,12(4HNODE,1X,4HTYPE,1X))
0063         19     FORMAT(5E12.5)
0064         20     FORMAT(12(2I5))
0065         21     FORMAT(10(I5))
0066        100     FORMAT(1H1,10A4)
0067        101     FORMAT(19H DAMPING FACTOR IS,E12.5,2X,5HSEC-1)
0068        102     FORMAT(20H NODAL POINTS        )
0069        103     FORMAT(20H ELEMENTS            )
0070        104     FORMAT(20H BOUNDARY CONDITIONS)
0071        105     FORMAT(20H SPECIFIED FREEDOMS      )
0072        106     FORMAT(45H SOLID-FLUID BOUNDARY CONSTRAINTS NODES ARE)
0073        108     FORMAT(1H ,19HMATERIAL PROPERTIES)
0074        109     FORMAT(11H  MATL. NO.,5X,3HE,C,8X,4HNU,C,7X,7HDENSITY)
0075       8002     FORMAT(1H ,3X,5H IRUN=,I5   ,5X,6H IEND=,I5)
0076       8001     FORMAT(2I5)
0077       8009     FORMAT(1H ,3X,7HITIME=,I5,5X,5HLMAX=,I5)
0078       8010     FORMAT(E10.4)
0079       8012     FORMAT(1H0,3X,7HERRMAX=,E11.4)
0080                END
```

```
      SUBROUTINE LOAD
      COMMON/SAB/TITLE(12),ORI(2,3),NSCON(7),CORD(54,2),NOP(72,6)
     1,IMAT(72),NBC(  ),NFIX(  ),NSD(22),NDFIX(22),NTYPE(54)
      COMMON/LAV/R(5),RR(270),DISP(22,3),INDEX(132 ),INACT(115),NACT(140)
      COMMON/PARAM/T,ITIME,TEND,NCN,        IRUN,IEND,IIIIME,LMAK
     1,NP,NE,N3,NLD,NMAT,II,NPRINT,ND,NDF,NIT,IBEG,NCON  ,ALPHA
     2,NSZF,NTIME,DELT,T1,          DAMP,ERRMAX
C     SET PARAMETERS
      NSZF=NP*5
      WRITE(6,1000) NSZF
C       PREPARE LOAD ARRAY
      DO177N=1,NB
C     N2 IS NODE AT WHICH LOAD ACTS
      N2=NBC(N)
      I=1
      IFIX=NFIX(N)+1
      GOTO(79,80),IFIX
      GOTO81
   79 I=3
      GOTO81
   80 I=2
   81 CONTINUE
C       READ MAGNITUDE OF LOAD
  179 READ(5,20)PMAX
      WRITE(6,21)
      WRITE(6,20)PMAX
C     STEP FUNCTION
  174 R(I)=PMAX
      I=I+1
      IFIX=IFIX-1
      IF(IFIX.EQ.1)GOTO179
      NN=N+1
      IF(NBC(N).EQ.NBC(NN))GOTO177
      NROWB=(N2-1)*5
      DO 170 II=1,5
      NROWB=NROWB+1
      RR(NROWB)=R(II)
  170 CONTINUE
      WRITE(6,77)N2,(R(K),K=1,5 )
  177 CONTINUE
  175 CONTINUE
C     READ,PRINT AND STORE DISPLACEMENTS,PRESSURE AND VELOCITIES
      DO182I=1,ND
      KK=1
      IF(NDFIX(I).EQ.1.OR.NDFIX(I).EQ.20.OR.NDFIX(I).EQ.22) KK=2
      IF(NDFIX(I) .EQ. 2)  KK=3
      NDF=3
      IF(NDFIX(I) .EQ. 200) NDF=1
      IF(NDFIX(I) .EQ. 20 .OR. NDFIX(I) .EQ. 220) NDF=2
      IF(NDFIX(I) .EQ. 2 .OR. NDFIX(I) .GT. 111) GO TO 82
C     INDICATES P,X-DIRECTION AND/OR Y-DIRECTION VELOCITIES GIVEN
      IF(NDFIX(I) .EQ. 11) NDF=1
      IF(NDFIX(I) .EQ. 1 .OR. NDFIX(I) .EQ. 11) NDF=2
C       INDICATES DISPLACEMENTS ARE SPECIFIED
   82 READ(5,4)(DISP(I,K),K=KK,NDF)
  182 CONTINUE
```

```
C     DETERMINE AND STORE THE DOF IN THE GLOBAL ARRAYS WHERE
C     DISPLACEMENTS ARE SPECIFIED
      KK=0
      DO50N=1,NC
      I=NSC(N)
      IF(NDF .EQ. 3) GO TO 40
      IF(NDF .EQ. 5) GO TO 41
C        LOGIC FOR DISPLACEMENTS
       NROWB=(I-1)*3
      IF(NDFIX(N) .EQ. 7) NROWB=NROWB+1
      MDF=1
      IF(NDFIX(N) .EQ. 11)  MDF=3
      GO TO 52
C     *** LOGIC FOR P,X-DIRECTION AND Y-DIRECTION VELOCITIES ONLY ***
   40 NROWB=(I-1)*5+2
      IF(NDFIX(N) .EQ. 20 .OR. NDFIX(N) .EQ. 22) NROWB=NROWB+1
      IF(NDFIX(N) .EQ. 2)  NROWB=NROWB+2
      MDF=1
      IF(NDFIX(N) .EQ. 22 .OR. NDFIX(N) .EQ. 220) MDF=2
      IF(NDFIX(N) .EQ. 202 .OR. NDFIX(N) .EQ. 222)  MDF=3
      IF(NDFIX(N) .EQ. 202)  INC=2
      GO TO 52
C        LOGIC FOR DISPLACEMENTS,PRESSURE AND VELOCITIES
   41 NROWB=(I-1)*5
      IF(NDFIX(N) .EQ. 1) NROWB=NROWB+1
      IF(NDFIX(N) .GE. 200) NROWB=NROWB+2
      IF(NDFIX(N) .EQ. 20 .OR. NDFIX(N) .EQ. 22) NROWB=NROWB+3
      IF(NDFIX(N) .EQ. 2) NROWB=NROWB+4
      MDF=1
      IF(NDFIX(N) .EQ. 202 .OR. NDFIX(N) .EQ. 222) MDF=3
      IF(NDFIX(N).EQ.11.OR.NDFIX(N).EQ.22.OR.NDFIX(N).EQ.220) MDF=2
      IF(NDFIX(N) .EQ. 202) INC=2
   52 DO 51  M=1,MDF,INC
      NROWBM=NROWB+M
      KK=KK+1
      INDEX(KK)=NROWBM
   51 CONTINUE
   50 CONTINUE
C     FIND INACTIVE DOF NUMBERS
      IF(NDF.LT.5)GOTO191
      NN=0
      DO189N=1,NP
      NT=NTYPE(N)
      GO TO(185,185,187,185,185),NT
  186 MDF=3
      NROWB=(N-1)*NDF+2
      GOTO188
  187 MDF=2
      NROWB=(N-1)*NDF
  188 DO189I=1,MDF
      NN=NN+1
      NROWB=NROWB+1
      INACT(NN)=NROWB
  189 CONTINUE
  185 CONTINUE
  191 CONTINUE
```

```
C     FIND ACTIVE DOF NUMBERS FOR ANALYTICAL INTEGRATION
      II=1
      DO145 I=1,NSZF
      ISW=0
      DO140 J=1,IDD
      IF(I.EQ.INACT(J))ISW=ISW+1
  140 CONTINUE
      IF(ISW.NE.0)GOTO145
      DO130 J=1,ISD
      IF(I.EQ.INDEX(J))ISW=ISW+1
  130 CONTINUE
      IF(ISW.NE.0)GOTO145
      NACT(II)=I
      II=II+1
  145 CONTINUE
  150 CONTINUE
C**************** OUTPUT BLOCK ****************
      WRITE(6,111)(INDEX(KK),KK=1,32 )
      WRITE(6,115)(INACT(KK),KK=1,II5)
      WRITE(6,120)(NACT(KK),KK=1,140)
      WRITE(6,118)
      DO53 I=1,ND
   53 WRITE(6,110)(I,K,DISP(I,K),K=1,3)
      WRITE(6,109)
      WRITE(6,112)(R4(I),I=1,NSZF)
C************** END OF OUTPUT BLOCK **************
      RETURN
    4 FORMAT(3F10.3)
   20 FORMAT(F10.4)
   21 FORMAT(1H ,3X,4HPMAX)
   77 FORMAT(1H ,3X,4HNODE,5X,4HLOAD/1H ,I5,2F10.2/2(1H ,5X,3F10.2/))
  100 FORMAT(1H1,I2/4I)
  102 FORMAT(1H ,14H SYSTEM PARAMETER /1H ,5X,4HNSZF/1H ,4X,I5)
  109 FORMAT(1H ,10HLOAD ARRAY)
  110 FORMAT(1H ,5HDISP(,I4,1H,,I4,2H)=,3X,F10.3/)
  111 FORMAT(1H ,52HSPECIFIED FREEDOMS OCCUR AT THE FOLLOWING DOF
     1/6(1H ,I3(I5,1X)/))
  112 FORMAT(5(1H ,10E12.5,1X)/)
  115 FORMAT(33H INACTIVE FREEDOMS ARE LOCATED AT/5(1H ,13(I5,1X)/))
  118 FORMAT(1H ,25HTHESE SPECIFIED FREEDOMS ARE)
  120 FORMAT(35H ANALYT. INTEGRATION DOF NUMBERS ARE/5(1H ,18(I6,1X)/))
      END
```

```
      SUBROUTINE FORMK
      COMMON/SAB/TITLE(12),ORT(2,3),NSCON(7),CORD(54,2),NOP(72,6)
     1,IEAD(72),NSD(1 ),NFIX(1 ),NSD(22),NDFIX(22),NTYPE(54)
      COMMON/MAT/AK(20,20),XM(20,20),SK(20,20)
      COMMON/SAVE/STOREK(20,20,72),STORIC(20,20,72)
      COMMON/END/DISPL(20),FORCEC(20),FORCEK(20),DIAACC(270)
     1,DIADIS(270),DIAVEL(270),DIAGC(270),DIACK(270),DIAK(270)
     2,MFCLC(270),DEM(270),DIAVC(270),DIAAC(270),ACCEL(20),DIAGM(270)
      COMMON/FREE/LS(20,72)
      COMMON/PARAMET/,IEND,TEND,NCN,        LRJ,IEND,ITIME,LMAK
     1,NP,NE,NB,NLD,NMAT,II,NPRINT,ND,NDF,NIT,IBEG,NCON ,ALPHA
     2,NSZF,NTIME,DELT,T1,         DAMP,ERRMAX
      DIMENSION MROW(3)
      DATA MROW(1),MROW(2),MROW(3)/6,3,20/
C       READ TIME INTERVAL,FIRST TIME VALUE,SET PRINTOUT PARAMETER
      DELT=(TEND-TBEG)/FLOAT(NTIME)
      T1=TBEG+DELT
      NTIME=(ITIME/NPRINT)*NPRINT
C     SCAN ELEMENTS
      DO 404 N=1,NE
      LL=IMAT(N)
      GO TO(401,402,403),LL
  401 CALL STIFT1(N)
C     SOLID ELEMENT
      NDF=2
      MM=8
      GOTO404
  402 CALL STIFT2(N)
C     FLUID ELEMENT
      NDF=3
      MM=8
      GOTO404
  403 CALL STIFT3(N)
C     FLUID-SOLID ELEMENT
      NDF=5
      MM=N
  404 CONTINUE
      NROW=MROW(LL)
C  REPEAT FOR ALL ELEMENTS,FORM GLOBAL DIAGONAL ARRAYS,
C  STORE ELEMENTAL MATRICES WITH ZERO DIAGONAL
C  REPEAT FOR ALL NODES IN AN ELEMENT
      I=0
      KROW=3
      IF(LL .GT. 2) KROW=4
      DO 92  JJ=1,KROW
      NROWB=(NOP(N,JJ)-1)*5
      IF (LL .EQ. 2) NROWB=NROWB+2
      DO 93  J=1,NDF
      I=I+1
      NROWB=NROWB+1
      LS(I,N)=NROWB
   93 CONTINUE
   92 CONTINUE
C   ******* FORM GLOBAL ARRAYS  *******
      DO 132  I=1,NROW
      ISL=LS(I,N)
```

```
            IF(II .LE. 2) GO TO 500
            IF(I .EQ. 1) WRITE(5,300N)
            WRITE(5,3010) N,I,LS(I,N),NSL
      500   CONTINUE
            DIAGK(NSL)=DIAGK(NSL)+AK(I,I)
            DIAGC(NSL)=DIAGC(NSL)+XM(I,I)
            DIAGM(NSL)=DIAGM(NSL)+GM(I,I)
      132   CONTINUE
C     ** [illegible] THE DIAGONAL OF MATRICES AND STORING INTO MATRICES
C        [illegible] WITH ELEMENT NUMBER ***
            DO 203 I=1,NROW
            AK(I,I)=0.
            XM(I,I)=0.
            DO 201 J=1,NROW
            STOREK(I,J,N)=AK(I,J)
            STOREC(I,J,N)=XM(I,J)
      201   CONTINUE
            IF(II.LT.3)GOTO203
            WRITE(5,150) II,N
            WRITE(5,170)
            DO 151 II=1,NROW
      151   WRITE(5,993) II,(JJ,STOREK(II,JJ,N),JJ=1,NROW)
            WRITE(5,180)
            DO 152 II=1,NROW
      152   WRITE(5,993) II,(JJ,STOREC(II,JJ,N),JJ=1,NROW)
      203   CONTINUE
      400   CONTINUE
C****************** OUTPUT BLOCK ***************
            IF(II .LT. 1) GO TO 135
      129   DO 134 NSL=1,NSIF
            IF(NSL .EQ. 1) WRITE(5,891) II
      134   WRITE(5,994) NSL,DIAGM(NSL),DIAGC(NSL),DIAGK(NSL)
      135   CONTINUE
C****************** END OF OUTPUT BLOCK ****************
      142   CONTINUE
            RETURN
      110   FORMAT(32H ORIGINAL NODAL COORDINATE ARRAY)
      150   FORMAT(12H DEBUG LEVEL,I4/8H EL. NO.,I4)
      170   FORMAT(24H ELEMENTAL STOREK MATRIX)
      180   FORMAT(24H ELEMENTAL STOREC MATRIX)
      891   FORMAT(12H DEBUG LEVEL,I4/96H0GLOBAL DIAGONAL ARRAY OF C MATRIX
     I                              GLOBAL DIAGONAL ARRAY OF K MATRIX)
      993   FORMAT(5H ROW,I4,5(I4,1H.,1X,E12.5,2X)/7(1H ,7X,5(I4,1H.,1X,E12.5,
     1 2X)/))
      994   FORMAT(1X,4HNSL=,I4,10X,E12.5,20X,E12.5,20X,E12.5)
      3014  FORMAT( ,10X,15HSOLUTION INDICES,/)
      3013  FORMAT(1X,12HELEMENT NO.=,I4,5X,2HI=,I2,5X,8HLS(I,N)=,I4,5X,
     1 4HNSL=,I4,/).
            END
```

```
0001          SUBROUTINE STIFF1(N)
0002          COMMON/GEOM/TITLE(32),ORT(2,3),NSCON(77),CORD(54,2),NOP(72,6)
             1,IMAT(72),NCEL(  ),NFIX(  ),NSD(22),NDFIX(22),NTYPE(54)
0003          COMMON/MATRX/X(20,20),XM(20,20),SM(20,20)
0004          COMMON/FUNCT/YF,CI,AB(3),AM(3),BB(3),BM(3),CB(3),CM(3),AA,AREAB
             1,AREAM,XI,XF,SLOPE,YB
0005          COMMON/PARAM/T,ITIME,TEND,NCN,        IRUN,IEND,IITIME,LMAX
             1,MP,NC,NB,ND,NMAT,II,NPRINT,ND,NDF,NII,TDEG,NCON ,ALPHA
             2,NSP,  TIME,DELT,TI,          DAMP,ERRMAX
      C          CALCULATE MASS,DAMPING AND STIFFNESS MATRICES FOR SOLID
      C       DETERMINE ELEMENT CONNECTIONS
0006          I=NOP(N,1)
0007          J=NOP(N,2)
0008          K=NOP(N,3)
0009          L=IMAT(N)
0010          IF(II.GE.3)WRITE(6,207)
0011          IF(II.GT.3)WRITE(5,150)II,N,I,J,K
      C       SET UP LOCAL COORDINATE SYSTEM
0012          CM(3)=CORD(J,1)-CORD(I,1)
0013          CM(2)=CORD(I,1)-CORD(K,1)
0014          CM(1)=CORD(K,1)-CORD(J,1)
0015          BM(3)=CORD(I,2)-CORD(J,2)
0016          BM(2)=CORD(K,2)-CORD(I,2)
0017          BM(1)=CORD(J,2)-CORD(K,2)
0018          IF(II.GT.2)WRITE(5,140)II,N,(BM(I),CM(I),I=1,NCN)
0019          AREAM=(CM(3)*BM(2)-CM(2)*BM(3))/2.
      C       BRANCH PART FOR AND IN SECTIONS
0020          IF(NPRINT.EQ.0)GO TO 20
      C       CALCULATE STIFFNESS MATRIX
0021          CI=ORT(L,1)/(((1.-ORT(L,2)*ORT(L,2))*AREAM)*4.)
0022          AK(1,1)=CI*((BM(1)*BM(1))+(((1.-ORT(L,2))*(CM(1)*CM(1)))/2.))
0023          AK(1,2)=CI*(((1.+ORT(L,2))*(BM(1)*CM(1)))/2.)
0024          AK(1,3)=CI*((BM(1)*BM(2))+(((1.-ORT(L,2))*(CM(1)*CM(2)))/2.))
0025          AK(1,4)=CI*((ORT(L,2)*BM(1)*CM(2))+((1.-ORT(L,2))*BM(2)*CM(1)/2.))
0026          AK(1,5)=CI*((BM(1)*BM(3))+(((1.-ORT(L,2))*(CM(1)*CM(3)))/2.))
0027          AK(1,6)=CI*((ORT(L,2)*BM(1)*CM(3))+((1.-ORT(L,2))*BM(3)*CM(1)/2.))
0028          AK(2,1)=AK(1,2)
0029          AK(2,2)=CI*((CM(1)*CM(1))+(((1.-ORT(L,2))*(BM(1)*BM(1)))/2.))
0030          AK(2,3)=CI*((ORT(L,2)*BM(2)*CM(1))+((1.-ORT(L,2))*BM(1)*CM(2)/2.))
0031          AK(2,4)=CI*((CM(1)*CM(2))+(((1.-ORT(L,2))*(BM(1)*BM(2)))/2.))
0032          AK(2,5)=CI*((ORT(L,2)*BM(3)*CM(1))+((1.-ORT(L,2))*BM(1)*CM(3)/2.))
0033          AK(2,6)=CI*((CM(1)*CM(3))+(((1.-ORT(L,2))*(BM(1)*BM(3)))/2.))
0034          AK(3,1)=AK(1,3)
0035          AK(3,2)=AK(2,3)
0036          AK(3,3)=CI*((BM(2)*BM(2))+(((1.-ORT(L,2))*(CM(2)*CM(2)))/2.))
0037          AK(3,4)=CI*(((1.+ORT(L,2))*(BM(2)*CM(2)))/2.)
0038          AK(3,5)=CI*((BM(2)*BM(3))+(((1.-ORT(L,2))*(CM(2)*CM(3)))/2.))
0039          AK(3,6)=CI*((ORT(L,2)*BM(2)*CM(3))+((1.-ORT(L,2))*BM(3)*CM(2)/2.))
0040          AK(4,1)=AK(1,4)
0041          AK(4,2)=AK(2,4)
0042          AK(4,3)=AK(3,4)
0043          AK(4,4)=CI*((CM(2)*CM(2))+(((1.-ORT(L,2))*(BM(2)*BM(2)))/2.))
0044          AK(4,5)=CI*((ORT(L,2)*BM(3)*CM(2))+((1.-ORT(L,2))*BM(2)*CM(3)/2.))
0045          AK(4,6)=CI*((CM(2)*CM(3))+(((1.-ORT(L,2))*(BM(2)*BM(3)))/2.))
0046          AK(5,1)=AK(1,5)
0047          AK(5,2)=AK(2,5)
```

```
      AK(5,3)=AK(3,5)
      AK(5,4)=AK(4,5)
      AK(5,5)=CONST*(BK(3)*BK(3))+(((1.-DRT(L,2))*(CM(3)*CM(3)))/2.)
      AK(5,6)=CON*(((1.+DRT(L,2))*(BM(3)*CM(3)))/2.)
      AK(6,1)=AK(1,6)
      AK(6,2)=AK(2,6)
      AK(6,3)=AK(3,6)
      AK(6,4)=AK(4,6)
      AK(6,5)=AK(5,6)
      AK(6,6)=CONST*((CM(3)*CM(3))+(((1.-DRT(L,2))*(BM(3)*BM(3)))/2.))
C        CALCULATE DAMPING AND MASS MATRICES
      CONSD=(DAMP*DRT(L,3)*AREAM)/FLOAT(NCN)
      CONST=DRT(L,3)*AREAM/FLOAT(NCN)
      DO 300 II=1,6
      DO 301 JJ=1,6
      CM(II,JJ)=0.
      GM(II,JJ)=0.
      IF(II.EQ.JJ)GOTO302
      GOTO301
  302 CM(II,JJ)=CONSD
      GM(II,JJ)=CONST
  301 CONTINUE
  300 CONTINUE
      IF(IR.LT.2)GOTO203
      WRITE(6,170)
      WRITE(6,150)II,N,(II,(AK(II,JJ),JJ=1,6),II=1,6)
      WRITE(6,180)
      WRITE(6,150)I,N,(II,(CM(II,JJ),JJ=1,6),II=1,6)
      WRITE(6,190)
      WRITE(6,150)I,N,(II,(GM(II,JJ),JJ=1,6),II=1,6)
  203 CONTINUE
      IF(IR.GE.2)WRITE(6,130)II,N,CONST,AREAM,CONSD,C1
      RETURN
C     ERROR EXIT FOR BAD CONNECTIONS
  220 WRITE(6,100)N
   50 CONTINUE
      STOP
  100 FORMAT(1X38H ZERO OR NEGATIVE AREA ELEMENT NO.,I4/21H EXECUTION TERM
     1INATED)
  130 FORMAT(12H DEBUG LEVEL,I4/23H PARAMETERS FOR EL. NO.,I4/1H ,5X,5HC
     1ONST=,E12.5,3X,5HAREA=,F10.4,2X,6HCONSD=,E12.5,2X,3HC1=,E12.5)
  140 FORMAT(12H DEBUG LEVEL,I4/26H LOCAL COORDINATES FOR EL.,I4,5X,1HB,
     15X,1HC/3(1H ,32X,F8.5,2X,F8.3/))
  150 FORMAT(12H DEBUG LEVEL,I4/6H EL. NO.,I4/6(4H ROW,I4,3X,6(E12.5,3X)
     1/))
  160 FORMAT(12H DEBUG LEVEL,I4/24H EL. CONNECTIONS FOR EL.,I4,3X,2HI=,I
     14,3X,2HJ=,I4,3X,2HK=,I4)
  170 FORMAT(19H ELEMENTAL K MATRIX)
  180 FORMAT(19H ELEMENTAL C MATRIX)
  190 FORMAT(19H ELEMENTAL M MATRIX)
  207 FORMAT(17H OUTPUT OF STIFTI)
      END
```

```
FORTRAN IV G LEVEL  21                STIFF2            DATE = 73202        14/58/39             PAGE 0001

 0001                SUBROUTINE STIFF2(N)
 0002                COMMON/G/S/TITLE(12),ORT(2,3),NSCON(7),CORD(54,2),NOP(72,6)
                    1,IMAT(72),NBC(I ),NFIX(I ),NSOC(22),NOFIX(22),NTYPE(54)
 0003                COMMON/MAT/AK(20,20),XM(20,20),SK(20,20)
 0004                COMMON/FUNCT/YF,(I,AB(3),AK(3),BB(3),BM(3),CB(3),CM(3),AH,AREAB
                    1,AREAM,XI,XF,SLOPE,YB
 0005                COMMON/PARAM/T,ITIME,TEND,NCN,       IRUN,IEND,ITTIME,LMAX
                    1,[illegible],NPRINT,ID,NDF,NIT,TBUG,NCON ,ALPHA
                    2,[illegible]       DAMP,ERRMAX
               C     [illegible] ELEMENT CONNECTIONS
 0006                I=NOP(N,1)
 0007                J=NOP(N,2)
 0008                K=NOP(N,3)
 0009                L=IMAT(N)
 0010                IF(IL.GE.3)WRITE(6,20)
 0011                IF(IL.GE.3)WRITE(6,150)L,N,I,J,K
               C     SET UP LOCAL COORDINATE SYSTEM
 0012                CB(3)=CORD(J,1)-CORD(I,1)
 0013                CB(2)=CORD(I,1)-CORD(K,1)
 0014                CB(1)=CORD(K,1)-CORD(J,1)
 0015                BB(3)=CORD(I,2)-CORD(J,2)
 0016                BB(2)=CORD(K,2)-CORD(I,2)
 0017                BB(1)=CORD(J,2)-CORD(K,2)
 0018                AB(3)=CORD(I,1)*CORD(J,2)-CORD(J,1)*CORD(I,2)
 0019                AB(2)=CORD(K,1)*CORD(I,2)-CORD(I,1)*CORD(K,2)
 0020                AB(1)=CORD(J,1)*CORD(K,2)-CORD(K,1)*CORD(J,2)
 0021                IF(IL.GE.3)WRITE(6,[illegible])(BB(I),CB(I),I=1,NCN)
 0022                AREAB=(CB(3)*BB(2)-CB(2)*BB(3))/2.
               C     ERROR EXIT FOR BAD CONNECTIONS
 0023                IF(AREAB.LE.0.)GO TO 220
 0024                CERM1=AREAB/(5.*ORT(L,1)*ORT(L,1)*ORT(L,3))
 0025                CERM2=CERM1/2.
 0026                CERM3=[illegible]ORT(L,3))/6.
 0027                CERM4=CERM3/2.
 0028                DO 501 II=1,9
 0029                DO 502 JJ=1,9
 0030                AK(II,JJ)=0.
 0031                XM(II,JJ)=0.
 0032                SK(II,JJ)=0.
 0033          502   CONTINUE
 0034          501   CONTINUE
               C        CALCULATE          FLUIDITY MATRIX
 0035                XM(1,1)=CERM1
 0036                XM(1,4)=CERM2
 0037                XM(1,7)=CERM2
 0038                XM(2,2)=CERM3
 0039                XM(2,5)=CERM4
 0040                XM(2,8)=CERM4
 0041                XM(3,3)=CERM3
 0042                XM(3,6)=CERM4
 0043                XM(3,9)=CERM4
 0044                XM(4,1)=CERM2
 0045                XM(4,4)=CERM1
 0046                XM(4,7)=CERM2
 0047                XM(5,2)=CERM4
 0048                XM(5,5)=CERM3
```

```
0049      KM(5,8)=CER44
0050      KM(5,7)=CER44
0051      KM(5,1)=CER43
0052      KM(5,9)=CER44
0053      KM(7,1)=CER42
0054      KM(7,4)=CER12
0055      KM(7,7)=CER41
0056      KM(8,7)=CER46
0057      KM(8,1)=CER44
0058      KM(8,7)=CER44
0059      KM(9,3)=CER44
0060      KM(9,6)=CER44
0061      KM(9,9)=CER43
C         CALCULATE INERTIA MATRIX
0062      TERM1=D(1)/6.
0063      TERM2=AD(L,1)/6.
0064      TERM3=(DRT(L,3)*DRT(L,2)*AREA)/5.
0065      TERM4=TERM3/2.
0066      TERM5=D(2)/6.
0067      TERM6=D(2)/6.
0068      TERM7=D(3)/6.
0069      TERM8=D(3)/6.
0070      AK(1,2)=TERM1
0071      AK(1,3)=TERM2
0072      AK(1,5)=TERM5
0073      AK(1,6)=TERM5
0074      AK(1,8)=TERM7
0075      AK(1,9)=TERM8
0076      AK(2,1)=TERM1
0077      AK(2,2)=TERM3
0078      AK(2,4)=TERM5
0079      AK(2,5)=TERM4
0080      AK(2,7)=TERM7
0081      AK(2,8)=TERM6
0082      AK(3,1)=TERM2
0083      AK(3,3)=TERM3
0084      AK(3,4)=TERM5
0085      AK(3,6)=TERM4
0086      AK(3,7)=TERM8
0087      AK(3,9)=TERM4
0088      AK(4,2)=TERM1
0089      AK(4,3)=TERM2
0090      AK(4,5)=TERM5
0091      AK(4,6)=TERM5
0092      AK(4,8)=TERM7
0093      AK(4,9)=TERM8
0094      AK(5,1)=TERM1
0095      AK(5,2)=TERM4
0096      AK(5,4)=TERM5
0097      AK(5,5)=TERM3
0098      AK(5,7)=TERM7
0099      AK(5,8)=TERM4
0100      AK(6,1)=TERM2
0101      AK(6,3)=TERM4
0102      AK(6,4)=TERM5
0103      AK(6,6)=TERM3
```

```
              AK(5,7)=TERM3
              AK(5,9)=TERM4
              AK(7,2)=TERM1
              AK(7,3)=TERM2
              AK(7,5)=TERM5
              AK(7,6)=TERM6
              AK(7,8)=TERM7
              AK(7,9)=TERM3
              AK(9,2)=TERM1
              AK(9,3)=TERM4
              AK(5,4)=TERM5
              AK(5,5)=TERM6
              AK(5,7)=TERM7
              AK(9,3)=TERM3
              AK(9,3)=TERM3
              AK(9,4)=TERM4
              AK(9,4)=TERM5
              AK(9,6)=TERM4
              AK(9,7)=TERM3
              AK(9,8)=TERM3
              IF(II .LT. 4)GOTO 203
              WRITE(6,170)
              WRITE(6,150)II,N,(II,(AK(II,JJ),JJ=1,9),II=1,9)
              WRITE(6,180)
              WRITE(6,150)II,N,(II,(XM(II,JJ),JJ=1,9),II=1,9)
              WRITE(6,190)
              WRITE(6,150)II,N,(II,(GM(II,JJ),JJ=1,9),II=1,9)
      203     CONTINUE
              IF(II .GE.3)WRITE(6,120)II,N,AREAD
              RETURN
C             ERROR EXIT FOR BAD CONNECTIONS
      220     WRITE(6,100)N
       50     CONTINUE
              STOP
       20     FORMAT(17H OUTPUT OF STIFT2)
      100     FORMAT(34H1ZERO OR NEGATIVE AREA ELEMENT NO.,I4/21H0EXECUTION TERM
             1INATED)
      120     FORMAT(12H DEBUG LEVEL,I4/23H PARAMETERS FOR EL. NO.,I4/1H ,
             15X,6HAREA =,F10.4)
      140     FORMAT(12H DEBUG LEVEL,I4/25H LOCAL COORDINATES FOR EL.,I4,6X,14H.
             13X,1HC/3(1H ,3X,F8.3,2X,F8.3/))
      150     FORMAT(12H DEBUG LEVEL,I4/8H EL. NO.,I4/9(4H ROW,I3,2X,9(E12.5,1X)
             1/))
      160     FORMAT(12H DEBUG LEVEL,I4/24H EL. CONNECTIONS FOR EL.,I4,3X,2HI=,I
             14,3X,2HJ=,I4,3X,2HK=,I4)
      170     FORMAT(19H ELEMENTAL H MATRIX)
      180     FORMAT(19H ELEMENTAL L MATRIX)
      190     FORMAT(19H ELEMENTAL G MATRIX)
              END
```

```
      SUBROUTINE STIFF3(N)
      COMMON/SPAR/TITLE(12),ORT(2,3),NSCON(7),CORD(54,2),NOP(72,6)
     1,IMAT(72),NBC(1 ),NFIX(1 ),NSD(22),NDFIX(22),NTYPE(54)
      COMMON/MAT/AK(20,20),AM(20,20),SA(20,20)
      COMMON/TWO/DISPL(20),FORCEC(20),FORCEK(20),DIAACC(27)
     1,DIADIS(270),DIAVEL(270),DIAGC(270),DIAS(270),DIAK(270)
     2,VELOC(20),DIADI(270),DIAVE(270),DIAAC(270),ACCEL(20),DIASM(270)
      COMMON/ELAS/D(3,3)
      COMMON/FRINT/ ... (1,1),AM(3),BM(3),EM(3),CD(3),CM(3),AA,AREA3
     1, ... SLOPE, ...
      COMMON/PARAM/T,ITI ,IEND,NCN,      IRUN,JEND,IITIME,LMAX
     1,NP,NE,  ,ND,NMAT,II,NPRINT,ND,NDF,NIT,IBEG,NCON ,ALPHA
     2,NSZF,ITIME,DELT,T1,          DAMP,ERRMAX
C        CALCULATE EQUIVALENT MASS,DAMPING AND STIFFNESS MATRICES FOR
C        QUADRILATERAL SOLID-FLUID ELEMENT
      DO 2001 II=1,20
      DO 2002 JJ=1,20
      AK(II,JJ)=0.
      AM(II,JJ)=0.
      SM(II,JJ)=0.
 2002 CONTINUE
 2001 CONTINUE
C     CALCULATE MATRICES FOR SOLID ELEMENT
      M=N
      L=1
C     DETERMINE ELEMENT CONNECTIONS
      I=NOP(M,1)
      J=NOP(M,2)
      K=NOP(M,3)
      IF(II.GT.3)WRITE(6,1102)
      IF(II.GT.3)WRITE(6,150)L,M,I,J,K
C     SET UP LOCAL COORDINATE SYSTEM
      CM(1)=CORD(J,1)-CORD(I,1)
      CM(2)=CORD(I,1)-CORD(K,1)
      CM(3)=CORD(K,1)-CORD(J,1)
      BM(1)=CORD(I,2)-CORD(J,2)
      BM(2)=CORD(K,2)-CORD(I,2)
      BM(3)=CORD(J,2)-CORD(K,2)
      AM(1)=CORD(K,1)*CORD(J,2)-CORD(J,1)*CORD(I,2)
      AM(2)=CORD(K,1)*CORD(I,2)-CORD(I,1)*CORD(K,2)
      AM(3)=CORD(J,1)*CORD(K,2)-CORD(K,1)*CORD(J,2)
      IF(II.GT.2)WRITE(6,140)(L,M,(BM(I),CM(I),AM(I),I=1,NCN)
      AREAM=(CM(3)*BM(2)-CM(2)*BM(3))/2.
C     ERROR EXIT FOR BAD CONNECTIONS
      IF(AREAM.LE.0.)GO TO 3000
C     CALCU LATE STIFFNESS MATRIX
      CI=ORT(L,1)/(((1.-ORT(L,2)*ORT(L,2))*AREAM)*4.)
      AK(1,1)=CI*((BM(1)*BM(1))+(((1.-ORT(L,2))*(CM(1)*CM(1)))/2.))
      AK(1,2)=CI*(((1.+ORT(L,2))*(BM(1)*CM(1)))/2.)
      AK(1,6 )=CI*((BM(1)*BM(2))+(((1.-ORT(L,2))*(CM(1)*CM(2)))/2.))
      AK(1,7 )=CI*(ORT(L,2)*BM(1)*CM(2)+((1.-ORT(L,2))*BM(2)*CM(1)/2.))
      AK(1,11)=CI*((BM(1)*BM(3))+(((1.-ORT(L,2))*(CM(1)*CM(3)))/2.))
      AK(1,12)=CI*(ORT(L,2)*BM(1)*CM(3)+((1.-ORT(L,2))*BM(3)*CM(1)/2.))
      AK(2,1)=AK(1,2)
      AK(2,2)=CI*((CM(1)*CM(1))+(((1.-ORT(L,2))*(BM(1)*BM(1)))/2.))
      AK(2,6 )=CI*(ORT(L,2)*CM(1)*BM(2)+((1.-ORT(L,2))*BM(1)*CM(2)/2.))
```

```
      AK(2,7 )=CI*((CM(1)*CM(2))+(((1.-DRT(L,2))*(BM(1)*BM(2)))/2.))
      AK(2,11)=CI*(DRT(L,2)*(CM(1)*CM(3))+(((1.-DRT(L,2))*BM(1)*CM(3))/2.))
      AK(2,12)=CI*((CM(1)*CM(3))+(((1.-DRT(L,2))*(BM(1)*BM(3)))/2.))
      AK(5,1)=AK(1,5 )
      AK(6 ,2)=AK(2,6 )
      AK(6 ,5 )=CI*((BM(2)*BM(2))+(((1.-DRT(L,2))*(CM(2)*CM(2)))/2.))
      AK(6 ,7 )=CI*DRT(L,2)*(BM(2)*CM(2))/2.)
      AK(5 ,11)=CI*((BM(2)*BM(3))+(((1.-DRT(L,2))*(CM(2)*CM(3)))/2.))
      AK(5 ,12)=CI*(DRT(L,2)*BM(2)*CM(3)+((1.-DRT(L,2))*BM(3)*CM(2)/2.))
      AK(7 ,1)=AK(1,7 )
      AK(7 ,2)=AK(2,7 )
      AK(7 ,5 )=AK(5 ,7 )
      AK(7 ,7 )=CI*((CM(2)*CM(2))+(((1.-DRT(L,2))*(BM(2)*BM(2)))/2.))
      AK(7 ,11)=CI*(DRT(L,2)*BM(3)*CM(2)+((1.-DRT(L,2))*BM(2)*CM(3)/2.))
      AK(7 ,12)=CI*((CM(2)*CM(3))+(((1.-DRT(L,2))*(BM(2)*BM(3)))/2.))
      AK(11,1)=AK(1,11)
      AK(11,2)=AK(2,11)
      AK(11,5)=AK(5,11)
      AK(11,7)=AK(7,11)
      AK(11,11)=CI*((BM(3)*BM(3))+(((1.-DRT(L,2))*(CM(3)*CM(3)))/2.))
      AK(11,12)=CI*DRT(L,2)*(BM(3)*CM(3))/2.)
      AK(12,1)=AK(1,12)
      AK(12,2)=AK(2,12)
      AK(12,5)=AK(5,12)
      AK(12,7)=AK(7,12)
      AK(12,11)=AK(11,12)
      AK(12,12)=CI*((CM(3)*CM(3))+(((1.-DRT(L,2))*(BM(3)*BM(3)))/2.))
C        CALCULATE DAMPING AND MASS MATRICES
      CONSD=DAMP*DRT(L,3)*AREA/FLOAT(NCN)
      CONST=DRT(L,3)*AREA/FLOAT(NCN)
      XM(1,1)=CONSD
      XM(2,2)=CONSD
      XM(6 ,6 )=CONSD
      XM(7 ,7 )=CONSD
      XM(11,11)=CONSD
      XM(12,12)=CONSD
      GM(1,1)=CONST
      GM(2,2)=CONST
      GM(6,6)=CONST
      GM(7,7)=CONST
      GM(11,11)=CONST
      GM(12,12)=CONST
C     FIND MATRICES FOR FLUID ELEMENT
      L=2
C     DETERMINE ELEMENT CONNECTIONS
      I=ICP(N,4)
      J=ICP(N,5)
      K=ICP(N,6)
      IF(IL.GE.3)WRITE(6,110)
      IF(IL.GE.3)WRITE(6,550)II,N,I,J,K
C     SET UP LOCAL COORDINATE SYSTEM
      CO(3)=CORD(J,1)-CORD(I,1)
      CO(2)=CORD(I,1)-CORD(K,1)
      CO(1)=CORD(K,1)-CORD(J,1)
      BO(3)=CORD(I,2)-CORD(J,2)
      BO(2)=CORD(K,2)-CORD(I,2)
```

```
0096        BO(1)=CORD(J,2)-CORD(K,2)
0097        AO(3)=CORD(I,1)*CORD(J,2)-CORD(J,1)*CORD(I,2)
0098        AO(2)=CORD(K,1)*CORD(I,2)-CORD(I,1)*CORD(K,2)
0099        AO(1)=CORD(J,1)*CORD(K,2)-CORD(K,1)*CORD(J,2)
0100        IF(II.GT.2)WRITE(6,140)II,N,(BO(I),CO(I),AO(I),I=1,NCN)
0101        AREAO=(CO(3)*BO(2)-CO(2)*BO(3))/2.
      C     ERROR EXIT FOR BAD CONNECTIONS
0102        IF(AREAO.LE.0.)GO TO 22
      C       CALCULATE           FLUIDITY MATRIX
0103        TERM1=BO(1)/5.
0104        TERM2=CO(1)/5.
0105        TERM3=(ORT(L,3)*ORT(L,2)*AREAO)/6.
0106        TERM4=TERM3/2.
0107        TERM5=BO(2)/5.
0108        TERM6=CO(2)/5.
0109        TERM7=BO(3)/5.
0110        TERM8=CO(3)/5.
0111        AK(3,4)=TERM7
0112        AK(3,5)=TERM8
0113        AK(3,9)=TERM5
0114        AK(3,10)=TERM6
0115        AK(3,19)=TERM1
0116        AK(3,20)=TERM2
0117        AK(4,3)=TERM7
0118        AK(4,4)=TERM4
0119        AK(4,8)=TERM5
0120        AK(4,9)=TERM4
0121        AK(4,13)=TERM1
0122        AK(4,19)=TERM3
0123        AK(5,3)=TERM8
0124        AK(5,5)=TERM4
0125        AK(5,8)=TERM6
0126        AK(5,10)=TERM4
0127        AK(5,15)=TERM2
0128        AK(5,20)=TERM3
0129        AK(8,4)=TERM7
0130        AK(8,5)=TERM8
0131        AK(8,9)=TERM5
0132        AK(8,10)=TERM6
0133        AK(8,19)=TERM1
0134        AK(8,20)=TERM2
0135        AK(9,3)=TERM7
0136        AK(9,4)=TERM4
0137        AK(9,8)=TERM5
0138        AK(9,9)=TERM3
0139        AK(9,13)=TERM1
0140        AK(9,19)=TERM4
0141        AK(10,3)=TERM8
0142        AK(10,5)=TERM4
0143        AK(10,8)=TERM6
0144        AK(10,10)=TERM3
0145        AK(10,13)=TERM2
0146        AK(10,20)=TERM4
0147        AK(18,4)=TERM7
0148        AK(18,5)=TERM8
0149        AK(18,9)=TERM5
```

```
      AK(13,13)=TERM5
      AK(13,19)=TERM1
      AK(13,20)=TERM2
      AK(19,3)=TERM7
      AK(19,4)=TERM3
      AK(19,8)=TERM5
      AK(19,9)=TERM4
      AK(19,13)=TERM1
      [illegible]
      [illegible]
      AK(20,5)=TERM3
      AK(20,8)=TERM6
      AK(20,13)=TERM4
      AK(20,19)=TERM2
      [illegible]
C     CALCULATE INERTIA MATRIX
      TERM1=AREA*(3.*DRT(L,1)*DRT(L,1)*DRT(L,3))
      TERM2=TERM1/2.
      TERM3=(AREA*DRT(L,3))/5.
      [illegible]
      XM(3,3)=TERM1
      XM(3,4)=TERM2
      XM(3,20)=TERM2
      XM(4,4)=TERM3
      XM(4,13)=TERM4
      [illegible]
      [illegible]
      [illegible]
      XM(5,13)=TERM4
      XM(13,3)=TERM2
      XM(13,3)=TERM2
      XM(13,13)=TERM1
      [illegible]
      [illegible]
      XM(19,9)=TERM4
      XM(20,5)=TERM4
      XM(20,20)=TERM3
      XM(20,13)=TERM4
      XM(8,9)=TERM2
      XM(8,13)=TERM2
      XM(8,8)=TERM1
      XM(9,4)=TERM4
      XM(9,20)=TERM4
      XM(9,9)=TERM3
      XM(10,5)=TERM4
      XM(10,10)=TERM4
      XM(10,10)=TERM3
C     CALCULATION OF SOLID-FLUID COUPLING MATRIX
C     DETERMINE INTEGRATION INTERVAL IN Y DIRECTION ONLY
C     FIND COMMON NODES
      IPASS=0
      DO35I=1,2
      IF(IPASS.EQ.2)GOTO35
      NN=NOP(N,I)
      IF(NTYPE(NN).NE.5)GOTO35
      IPASS=IPASS+1
```

```
      IF(IPASS.EQ.2)GOTO40
      NODE=NOP(N,1)
      GOTO45
   40 NODE=NOP(N,1)
   45 YI=CORD(NODE,2)
      YF=CORD(NODE1,2)
      XI=CORD(NODE,1)
      XF=CORD(NODE1,1)
      IF(XI-XF)45,47,45
C     VERTICAL WALL
C       AM IS CONSTANT VALUE OF X OR Y
   47 AM=XI
      SLOPE=0.
   45 IF(YI-YF)35,49,35
C     HORIZONTAL WALL
   49 AM=YI
   35 CONTINUE
      IF(XI.NE.XF)SLOPE=(YF-YI)/(XF-XI)
      YO=YI-SLOPE*XI
C     [illegible] SECT,JJ IS APPLICABLE ONLY TO SPECIFIC PROBLEMS
      [illegible]
      [illegible]
      JJ=(J/2)*2
      IF(JJ.EQ.J)GOTO35
      JN=(J/2)+1
      S(I,J)=SE(I,JN)*SIN(ALPHA)
      GOTO55
   50 S(I,J)=SE(I,JN)*COS(ALPHA)
   55 CONTINUE
   50 CONTINUE
      AK(1,3)=-S(3,1)
      AK(1,8)=-S(2,1)
      AK(1,13)=-S(1,1)
      AK(2,3)=-S(3,2)
      AK(2,8)=-S(2,2)
      AK(2,13)=-S(1,2)
      AK(6,3)=-S(3,3)
      AK(6,8)=-S(2,3)
      AK(6,13)=-S(1,3)
      AK(7,3)=-S(3,4)
      AK(7,8)=-S(2,4)
      AK(7,13)=-S(1,4)
      AK(11,3)=-S(3,5)
      AK(11,8)=-S(2,5)
      AK(11,13)=-S(1,5)
      AK(12,3)=-S(3,6)
      AK(12,8)=-S(2,6)
      AK(12,13)=-S(1,6)
      IF(II.LT.3)GOTO2031
      WRITE(5,150) II,I
      WRITE(5,1701)
      DO 151 II=1,20
  151 WRITE(5,993) II,(JJ,AK(II,JJ),JJ=1,20)
      WRITE(5,1301)
      DO 152 II=1,20
  152 WRITE(5,993) II,(JJ,TM(II,JJ),JJ=1,20)
```

```
          WRITE(5,7901)
          DO 153 II=1,20
   153    WRITE(5,1504) II,(JJ,SP(II,JJ),JJ=3,20)
          WRITE(5,1801)
          WRITE(5,1502)II,N,(II,(S(II,JJ),JJ=1,6),II=1,3)
          WRITE(5,1201) II,N,CONST,AREAM,CONSD,C1,AREAD
  2031    CONTINUE
          RETURN
C         ERROR EXIT FOR BAD CONNECTIONS
   220    WRITE(6,1000)
          STOP
  2201    WRITE(5,100)A
          STOP
   100    FORMAT(30H NEGATIVE OR ZERO AREA ELEMENT NO.,I4/21H EXECUTION TERM
         1INATED)
   110    FORMAT(20H STIFF3 FLUID OUTPUT)
   140    FORMAT(12H DEBUG LEVEL,I4/25H LOCAL COORDINATES FOR EL.,I4,6X,1HB,
         113X,1HC,3X,1HA/3(1H ,32X,3(F10.4,2X)/))
   150    FORMAT(12H DEBUG LEVEL,I4/8H EL. NO.,I4)
   160    FORMAT(12H DEBUG LEVEL,I4/24H EL. CONNECTIONS FOR EL.,I4,3X,2HI=,I
         1[illegible])
   180    FORMAT(30H COUPLING MATRIX FOR FLUID EL.,I4)
   995    FORMAT(4H ROW,I4,5(I4,1H.,1X,E12.5,2X)/7(1H ,7X,5(I4,1H.,1X,E12.5,
         12X)/))
  1101    FORMAT(20H STIFF3 SOLID OUTPUT)
  1201    FORMAT(12H DEBUG LEVEL,I4/23H PARAMETERS FOR EL. NO.,I4/1H ,5X,6HC
         1ONST=,E12.5,2X,7HAREAM=,F10.4,2X,6HCONSD=,E12.5,2X,3HC1=,E12.5,2X,5
         2HAREA=,F10.4)
  1502    FORMAT(12H DEBUG LEVEL,I3/8H EL. NO.,I4/3(4H ROW,I4,3X,5(E12.5,3X)
         1/))
  1701    FORMAT(27H ELEMENTAL STIFFNESS MATRIX)
  1801    FORMAT(25H ELEMENTAL DAMPING MATRIX)
  1901    FORMAT(22H ELEMENTAL MASS MATRIX)
          END
```

```
FORTRAN IV G LEVEL  21                SF                DATE = 73202          14/58/39              PAGE 0001

 0001                   FUNCTION SF(I,J)
 0002                   COMMON/FUNCT/YF,YI,AD(3),AM(3),BD(3),BM(3),CD(3),CM(3),AH,AREAD
                       1,AREAM,XI,XF,SLOPE,Y0
                C       CALCULATES FLUID/SOLID COUPLING FUNCTION
                C          VERTICAL WALL
                C       FIND DIFFERENCE FUNCTIONS
 0003                   DIFF=ABS(YF-YI)
 0004                   DIFSQ=ABS(YF*YF-YI*YI)/2.
 0005                   DIFCU=ABS(YF*YF*YF-YI*YI*YI)/3.
 0006                   DIFFX=ABS(XF-XI)
 0007                   DIFSQX=ABS(XF*XF-XI*XI)/2.
 0008                   DIFCUX=ABS(XF*XF*XF-XI*XI*XI)/3.
 0009                   IF(DIFF.EQ.0.)GO TO 10
 0010                   IF(DIFFX.NE.0.)GO TO 20
                C       FIND LINEAR TERM
 0011                   ALIN=(AD(I)*AM(J)+(AM(J)*BD(I)+AD(I)*BM(J))*AH+BD(I)*BM(J)*AH*AH)*
                       1DIFF
                C       FIND SQUARED TERM
 0012                   SQU=(AM(J)*CD(I)+AD(I)*CM(J)+(BD(I)*CM(J)+BM(J)*CD(I))*AH)*DIFSQ
                C       FIND CUBIC TERM
 0013                   CUB=CD(I)*CM(J)*DIFCU
 0014                   GO TO 30
 0015              10   CONTINUE
                C       FOR HORIZONTAL WALL
                C       FIND LINEAR TERM
 0016                   ALIN=(AD(I)*AM(J)+(AD(I)*CM(J)+AM(J)*CD(I))*AH+CD(I)*CM(J)*AH*AH)*
                       1DIFFX
                C       FIND SQUARED TERM
 0017                   SQU=(AD(I)*BM(J)+AM(J)*BD(I)+(BD(I)*CM(J)+BM(J)*CD(I))*AH)*DIFSQX
                C       FIND CUBIC TERM
 0018                   CUB=BD(I)*BM(J)*DIFCUX
 0019                   GO TO 30
 0020              20   CONTINUE
                C       SLOPED WALL
                C       FIND LINEAR TERM
 0021                   ALIN=(AD(I)*AM(J)+(AD(I)*CM(J)+AM(J)*CD(I))*Y0+CD(I)*CM(J)*Y0*Y0)*
                       1DIFFX
                C       FIND SQUARED TERM
 0022                   SQU=(AD(I)*BM(J)+AM(J)*BD(I)+(AD(I)*CM(J)+AM(J)*CD(I))*SLOPE+(BD(I
                       1)*CM(J)+BM(J)*CD(I))*Y0+CD(I)*CM(J)*2.*Y0*SLOPE)*DIFSQX
                C       FIND CUBIC TERM
 0023                   CUB=(BD(I)*BM(J)+(BD(I)*CM(J)+BM(J)*CD(I))*SLOPE+CD(I)*CM(J)*SLOPE
                       1*SLOPE)*DIFCUX
 0024              30   CONTINUE
                C       FIND FUNCTION
 0025                   SF=(ALIN+SQU+CUB)/(4.*AREAD*AREAM)
 0026                   RETURN
 0027                   END
```

```
      SUBROUTINE FORCE
      COMMON/ONE/TITLE(72),ORT(2,3),NSCON(77),CORD(54,2),NOP(72,6)
     1,IFIX(72),PL(1 ),NFIX( ),NSD(22),NDFIX(22),NTYPE(5)
      COMMON/LEN/R(5),S(270),DISP(2,3),INDEX(32 ),IVACT(115),VACT(140)
      COMMON/FRAT/STOREK(20,20,72),STOREC(20,20,72)
      COMMON/TWO/DISPL(20),FORCEC(20),FORCEK(20),DIAACC(270)
     1,DIADIS(270),DIAVEL(270),DIAGC(270),DIACK(270),DIAK(270)
     2,FLOC(20),DIADI(270),DIAVE(270),DIAAC(270),ACCEL(20),DIAGM(270)
      COMMON/ /ELES(20,72)
      COMMON/ /DIADIT(270),DIAK1(270)
      COMMON/PAR/NT,ITIME,NEND,NCN,       IRUN,IEND,IITIME,LMAX
     1,NP,NE,NB,NLD,NMAT,II,NPRINT,ND,NDF,KIT,IBEG,NCON ,ALPHA
     2,NSZE,TIME,DELT,TL,          DAMP,ERRMAX
      DIMENSION NROW(3)
      DATA    (KK(I),KAINE(I),NROW(I)/5,3,20/
C******** FORCES FOR ANALYTICAL INTEGRATION **********
C        UNLOAD GLOBAL DISPLACEMENTS,X-DIRECTION AND Y-DIRECTION
C        VELOCITIES,VELOCITY ARRAY AND ACCELERATION ARRAY INTO
C                      ELEMENTAL ARRAY
      DO 900 IJ=1,NSZE
      DIADI(IJ)=DIADIS(IJ)
      DIAVE(IJ)=DIAVEL(IJ)
      DIAAC(IJ)=DIAACC(IJ)
  900 CONTINUE
      IF(ITIME .EQ. 0) GO TO 301
      L=L+1
      IF(ITIME .EQ. 1) L=1
      DO 100 L=1,LK
      IF(L .LT. LMAX) GO TO 455
      WRITE(5,5012)
      STOP
  455 CONTINUE
      IF(L .EQ. 1) GO TO 839
      DO 838 KK=1,NSZE
      DIADIT(KK)=DIADIS(KK)
  838 CONTINUE
  839 CONTINUE
      DO 700 KJ=1,NSZE
      DIAK(KJ)=0.
      DIADIS(KJ)=0.5*DIADI(KJ)+0.5*DIADIS(KJ)
      DIAVEL(KJ)=0.5*DIAVE(KJ)+0.5*DIAVEL(KJ)
  700 CONTINUE
      DO 500 N=1,NE
      LL=INAT(N)
      NRO=NROW(LL)
      IF(II .LE. 1) GO TO 121
      WRITE(5,3024) ITIME
  121 CONTINUE
      DO 301 I=1,NROW
      NSL=LS(I,N)
      DISPL(I)=DIADIS(NSL)
      VELOC(I)=DIAVEL(NSL)
      IF(II.LE.1) GO TO 250
      WRITE(5,3010) N,I,NSL
      WRITE(5,2000)DISPL(I),VELOC(I)
  250 CONTINUE
```

```
      301 CONTINUE
C************** CALCULATE FORCES ***************
          DO 202 I=1,NROW
          FORCEK(I)=0.
          FORCEC(I)=0.
          DO 203 K=1,NROW
          FORCEK(I)=FORCEK(I)+STOREK(I,K,N)*DISPL(K)
          FORCEC(I)=FORCEC(I)+STOREC(I,K,N)*VELOC(K)
      203 CONTINUE
      202 CONTINUE
C********** ASSEMBLE THE RESULTANT FORCES ***********
          IF(II .NE. 1) GO TO 426
          WRITE(5,3027)
      426 CONTINUE
          DO 206 I=1,NROW
          NSL=NSL(I,N)
          DIAK(NSL)=DIAK(NSL)-FORCEC(I)-FORCEK(I)
          IF(II .LE.1) GO TO 252
          WRITE(5,3030) N,I,NSL
          PRINT ...,FORCEC(I),FORCEK(I),DIAK(NSL)
      252 CONTINUE
      206 CONTINUE
      500 CONTINUE
          IF(NOF .EQ. 3) GO TO 1001
          DO 1000 JJ=1,NSZF
          DIAK(JJ)=DIAK(JJ)+R4(JJ)
          IF(II .LT. 1) GO TO 253
          WRITE(5,2003) DIAK(JJ),R4(JJ)
      253 CONTINUE
     1000 CONTINUE
     1001 CONTINUE
          CALL RIGIES
          IF(II .EQ. 1) GO TO 940
          DO 335 JJ=1,NSZF
          ERR =DIADIS(JJ)-DIADIT(JJ)
          IF(ABS(ERR ) .GT. ERRMAX) GO TO 800
      335 CONTINUE
          GO TO 301
      800 CONTINUE
      940 CONTINUE
          IF(T.EQ.TIO)GOTO210
          IF(NTIME.NE.ITIME)RETURN
      210 CONTINUE
C************** OUTPUT BLOCK ***************
          WRITE(5,720)T,DELT,ITIME,NTIME,NPRINT,TI,II
          NODE=0
          WRITE(5,5000)
          DO 110 IJ=1,NSZF,5
          IL=IJ+4
          NODE=NODE+1
          WRITE(5,5001) NODE,(IK,DIADIS(IK), IK=IJ,IL)
C************** END OF OUTPUT BLOCK ***************
      110 CONTINUE
          RETURN
      720 FORMAT(44H FINITE ELEMENT ANALYT. INTEGRATION SOLUTION/21H SOLUTIO
         1N FOR TIME T=,E12.5/21H TIME INCREMENT DELT=,E12.5/21H ITERATION N
```

```
FORTRAN IV G LEVEL  21                 FORCE              DATE = 78202          14/58/39              PAGE 0003

                      NUMBER IT=,I5/21H PRINTOUT VARIABLE N=,I5/13H OUTPUT EVERY,I5,2X,5H
                      3TIMES/27H TIME AT FIRST ITERATION IS,E12.5/12H DEBUG LEVEL,I4)
 0099           2000 FORMAT(1X,5E12.5)
 0100           3010 FORMAT(5X,5I7)
 0101           3024 FORMAT(1H ,10X,33HDISPLACEMENT AND VELOCITY ITIME= ,I5)
 0102           3027 FORMAT(1H ,10X,19HFORCED FORCEK DIAK ,/)
 0103           6000 FORMAT(1H ,14X,13HU X-DIRECTION,9X,13HU Y-DIRECTION,14X,8HPRESSURE
                      1,13X,13HV X-DIRECTION,8X,13HV Y-DIRECTION/1H ,2X,4HNODE,5(5X,3HDOF
                      2,5X,5H[illegible],//)
 0104           6001 FORMAT(1X,I4,5(I3,2X,E12.5))
 0105           6002 FORMAT(1H0,2X,22HL IS GREATER THAN LMAX)
 0106                 END
```

```
0001            SUBROUTINE AINTEG
0002            COMMON/TAPE/TITLE(12),ORI(2,3),NSCON(7),CORD(54,2),NOP(72,5)
               1,INAL(72),NBC(1 ),NFIX(1 ),NSD(22),NDFIX(22),NTYPE(54)
0003            COMMON/LARG/R(5),AA(270),DISP(22,3),INDEX(32 ),INVACT(115),NACT(140)
0004            COMMON/DNE/NSDF(270),NSCON1(7),NSCON2(7)
0005            COMMON/ENO/DISPL(20),FORCEC(20),FORCEK(20),DIAACC(270)
               1,DIADIS(270),DIAVEL(270),DIAGC(270),DIAGK(270),DIAK(270)
               2,VELOC(20),DI(270),DIAVE(270),DIAAC(270),ACCEL(20),DIAGM(270)
0006            COMMON/FIVE/[illegible],DIAKI(270)
0007            COMMON/PARAM/TIME,[illegible],NCN,      IRUN,IEND,IITIME,LAAK
               1,NP,NF,NB,ICD,NMAT,II,NPRINT,ND,NDF,NIT,TBEG,NCON ,ALPHA
               2,NSZF,NTIME,DELT,TI,        DAMP,ERRMAX
      C         REPEAT FOR ALL FREEDOMS
0008            DO 204 IJ=1,NSZF
      C         INCLUDE SPECIFIED DISPLACEMENTS AND SOLID-FLUID BOUNDARY
      C                                CONSTRAINTS
0009            IF(NSDF(IJ) .EQ. 1) GO TO 20.
0010            DO 51 N=1,NCON
0011            NROWB=NSCON2(N)
0012            IF(IJ .NE. NROWB) GO TO 51
0013            I=NSCON1(N)
0014            J=NSCON2(N)
0015            DIADIS(J)=DIAVEL(I)
0016            GO TO 20.
0017     51     CONTINUE
      C           * SELECT STATIC SOLUTION CHANNEL *
0018            IF(DIAGM(IJ) .EQ. 0.) GO TO 700
0019            IF(DIAGC(IJ) .EQ. 0.) GO TO 701
0020            IF(DIAGK(IJ) .EQ. 0.) GO TO 720
0021            GO TO 710
0022    701     CONTINUE
0023            IF(DIAGK(IJ) .EQ. 0.) GO TO 750
0024            GO TO 710
0025    700     CONTINUE
0026            IF(DIAGC(IJ).EQ.0.)GO TO 702
0027            IF (DIAGK(IJ).EQ.0.)GO TO 750
0028            GO TO 730
0029    702     CONTINUE
0030            IF(DIAGK(IJ).EQ.0.)GO TO 770
0031            GO TO 740
0032    710     CONTINUE
      C         *FORMULATION I (M.NE.0,C.NE.0,K.NE.0)OR(M.NE.0,C.EQ.0,K.NE.0)*
0033            FDK=DIAK(IJ)/DIAGK(IJ)
0034            EC1=DIADIS(IJ)-FDK
0035            OMEGN=SQRT(ABS(DIAGK(IJ)/DIAGM(IJ)))
0036            ZZZ=DIAGC(IJ)/(2.*OMEGN*DIAGM(IJ))
0037            IF(ZZZ-1.)  1,2,3
0038      3     CONTINUE
      C           * OVERDAMPED *
0039            ROOT=SQRT(ZZZ**2-1.)
0040            RONE=OMEGN*(-ZZZ+ROOT)
0041            RTWO=OMEGN*(-ZZZ-ROOT)
0042            EC2=EXP(RONE*DELT)
0043            EC3=EXP(RTWO*DELT)
0044            EC4=-(DIAVE(IJ)-RTWO*EC1)*EC2
0045            EC5=(DIAVE(IJ)-RONE*EC1)*EC3
```

```
      EC5=(RTW3-RONE)
      DIADIS(IJ)=(EC4+EC5)/EC5*FDK
      DIAVEL(IJ)=(RONE*EC4+RTW3*EC5)/EC5
      DIAACC(IJ)=((RONE**2)*EC4*(RTWO**2)*EC5)/EC6
      IF(IJ.LT.I) GO TO 204
      WRITE(6,3030)
      WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
      WRITE(6,6001) DELT,OMEGN,ZZZ,RONE,RTWO
      GO TO 204
    2 CONTINUE
C     * CRITICALLY DAMPED *
      EC2=DIAVE(IJ)+EC1*OMEGN
      EC3=EC1+EC2*DELT
      EC4=EXP(OMEGN*(-DELT))
      FDK=[illegible]
      EC5=[illegible]
      DIADIS(IJ)=EC5*FDK
      DIAVEL(IJ)=(-OMEGN)*EC5*EC6
      DIAACC(IJ)=(OMEG(**2)*EC5*2.*(OMEGN)*EC6
      IF(IJ.LT.I) GO TO 204
      WRITE(6,3030)
      WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
      WRITE(6,5000) DELT,OMEGN,ZZZ,EC1,EC2,EC4
      GO TO 204
    1 CONTINUE
C     * UNDERDAMPED *
      OMEGD=OMEGN*SQRT(1.-ZZZ**2)
      ZZGN=ZZZ*OMEGN
      EC2=ZZGN*EC1+DIAVE(IJ)
      EC3=EXP(ZZGN*(-DELT))
      ARGUM=OMEGD*DELT
      SINE=SIN(ARGUM)
      COSINE=COS(ARGUM)
      EC4=EC3*(EC1*COSINE+EC2*SINE/OMEGD)
      DIADIS(IJ)=FDK+EC4
      EC5=EC3*(EC2*COSINE-EC1*OMEGD*SINE)
      DIAVEL(IJ)=(-ZZGN)*EC4+EC5
      DIAACC(IJ)=(-ZZGN)*(DIAVEL(IJ)+EC5)-EC4*OMEGD*OMEGD
      IF(IJ.LT.I) GO TO 204
      WRITE(6,3030)
      WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
      WRITE(6,2000) DELT,OMEGN,ZZZ,OMEGD,SINE,COSINE
      WRITE(6,2000) FDK,EC1,EC2,EC3,EC4,EC5
      GO TO 204
  720 CONTINUE
C     *** FORMULATION 2 (M.NE.0,C.NE.0,K.EQ.0) ***
      CDM=DIAGC(IJ)/DIAGM(IJ)
      FDC=DIAF(IJ)/DIAGC(IJ)
      EC1=DIAVE(IJ)-FDC
      EC2=EXP(CDM*(-DELT))
      EC3=EC1*EC2
      DIADIS(IJ)=DIADI(IJ)+(EC1-EC3)/CDM+FDC*DELT
      DIAVEL(IJ)=EC3+FDC
      DIAACC(IJ)=-CDM*EC3
      IF(IJ.LT.I) GO TO 205
      WRITE(6,3030)
```

```
          WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
          WRITE(6,2000) DELT,EC1,EC2,EC3
          GO TO 234
  730     CONTINUE
C         ***FORMULATION 3 (C.NE.0,K.NE.0)***
          FDK=DIAK(IJ)/DIAGK(IJ)
          ZKC=DIAGK(IJ)/DIAGC(IJ)
          DIADIS(IJ)=FDK*DELT/[illegible](IJ)-FDK*EXP(DIAGK(IJ)*(-DELT)/DIAGC(IJ))
          DIAVEL(IJ)=(-DIAGK(IJ)/DIAGC(IJ))*(DIADI(IJ)-FDK)*
         1EXP(DIAGK(IJ)*(-DELT)/DIAGC(IJ))
          DIAACC(IJ)=(-ZKC)*DIAVEL(IJ)
          IF(II.LT.3) GO TO 234
          WRITE(6,5030)
          WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
          GO TO 234
  740     CONTINUE
C         *** FORMULATION 4 (C.EQ.0,K.NE.0)**
          DIADIS(IJ)=DIAK(IJ)/DIAGK(IJ)
          DIAVEL(IJ)=0.
          DIAACC(IJ)=0.
          IF(II.LT.3) GO TO 234
          WRITE(6,5040)
          WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
          GO TO 234
  750     CONTINUE
C         *** FORMULATION 5 (C.NE.0,K.EQ.0)***
          FDC=DIAK(IJ)/DIAGC(IJ)
          DIADIS(IJ)=DIADI(IJ)+FDC*DELT
          DIAVEL(IJ)=FDC
          DIAACC(IJ)=0.
          IF(II.LT.3) GO TO 234
          WRITE(6,5050)
          WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
          GO TO 234
  760     CONTINUE
C             *** FORMULATION 6 (M.NE.0,C.EQ.0,K.EQ.0) ***
          FDM=DIAK(IJ)/DIAGM(IJ)
          DIADIS(IJ)=DIADI(IJ)+DIAVE(IJ)*DELT+FDM*(DELT**2)/2.
          DIAVEL(IJ)=DIAVE(IJ)+FDM*DELT
          DIAACC(IJ)=FDM
          IF(II.LT.3) GO TO 234
          WRITE(6,5060)
          WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
          GO TO 234
  770     CONTINUE
C         ***FORMULATION 7 (C.EQ.0,K.EQ.0)**
          DIADIS(IJ)=1.409E-23
          DIAVEL(IJ)=DIADIS(IJ)
          DIAACC(IJ)=DIADIS(IJ)
          IF(II.LT.3) GO TO 234
          WRITE(6,5070)
          WRITE(6,5002) IJ,DIADIS(IJ),DIAVEL(IJ),DIAACC(IJ)
  234     CONTINUE
          RETURN
  820     FORMAT(5(1H ,10(E12.5,1X)/))
 2000     FORMAT(1X,5E12.5)
```

```
0149         5002 FORMAT(1X,3HIJ=,I5,5X,4HDIS=,E12.5,5X,4HVEL=,E12.5,5X,4HACC=,E12.5
                 1)
0150         5010 FORMAT(1H ,10X,25HFORMULATION 1-UNDERDAMPED)
0151         5020 FORMAT(1H ,10X,13HFORMULATION 2)
0152         5030 FORMAT(1H ,10X,13HFORMULATION 3)
0153         5040 FORMAT(1H ,10X,13HFORMULATION 4)
0154         5050 FORMAT(1H ,10X,13HFORMULATION 5)
0155         5060 FORMAT(1H ,10X,13HFORMULATION 6)
0156         5070 FORMAT(1H ,10X,13HFORMULATION 7)
0157         5080 FORMAT(1H ,10X,24HFORMULATION 1-OVERDAMPED)
0158         5090 FORMAT(1H ,10X,32HFORMULATION 1-CRITICALLY DAMPED)
0159              END
```

72 ELEMENT FLUID-SOLID MODEL

```
ERRMAX= 0.1000E-01
ITERA=     3      IEVD=    1
ITEREF=    2      LMAX=   30
NP      NE      NQ      NUD     NKAE     II      NPRINT    ND      NDF     NCON
45      72      1       3       2        0       700       21      5       7
NDT     FBEG    TEND    ALPHA
5000    0.0     0.002000   0.0
MATERIAL PROPERTIES
MATL NO.     E,C            NU,K          DENSITY
   1   0.300000E 08  0.300000 00  0.729700E-03
   2   0.360000E 05  0.0           0.935520E-04
DAMPING FACTOR IS 0.0          SEC-1
NODAL POINTS
```

NODE	X-COORD	Y-COORD
1	0.500	0.0
2	0.500	4.000
3	0.500	8.000
4	0.600	12.000
5	0.500	12.250
6	0.500	12.500
7	2.700	0.0
8	2.700	4.000
9	2.700	8.000
10	2.700	12.000
11	2.700	12.250
12	2.700	12.500
13	6.900	0.0
14	6.900	4.000
15	6.900	8.000
16	6.900	12.000
17	6.900	12.250
18	6.900	12.500
19	11.100	0.0
20	11.100	4.000
21	11.100	8.000
22	11.100	12.000
23	11.100	12.250
24	11.100	12.500
25	15.300	0.0
26	15.300	4.000
27	15.300	8.000
28	15.300	12.000
29	15.300	12.250
30	15.300	12.500
31	19.500	0.0
32	19.500	4.000
33	19.500	8.000
34	19.500	12.000
35	19.500	12.250
36	19.500	12.500
37	23.700	0.0
38	23.700	4.000
39	23.700	8.000
40	23.700	12.000
41	23.700	12.250
42	23.700	12.500
43	27.900	0.0
44	27.900	4.000
45	27.900	8.000

46	27.900	12.000
47	27.900	12.250
48	27.900	12.500
49	30.000	0.0
50	30.000	4.000
51	30.000	8.000
52	30.000	12.000
53	30.000	12.250
54	30.000	12.500

NODAL DEGREES OF FREEDOM

NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE	NODE	TYPE
1	3	2	3	3	3	4	5	5	2	6	2	7	3	8	3	9	3	10	5	11	2	12	2
13	3	14	3	15	3	16	5	17	2	18	2	19	3	20	3	21	3	22	5	23	2	24	2
25	3	26	3	27	3	28	5	29	2	30	2	31	3	32	3	33	3	34	5	35	2	36	2
37	3	38	3	39	3	40	5	41	2	42	2	43	3	44	3	45	3	46	5	47	2	48	2
49	3	50	3	51	3	52	5	53	2	54	2												

ELEMENTS

	NODES						MATL. NO.
1	1	7	8	0	0	0	2
2	1	8	2	0	0	0	2
3	2	8	9	0	0	0	2
4	2	9	3	0	0	0	2
5	3	9	10	0	0	0	2
6	4	10	11	3	10	4	3
7	4	11	5	0	0	0	1
8	5	11	12	0	0	0	1
9	5	12	6	0	0	0	1
10	7	13	14	0	0	0	2
11	7	14	8	0	0	0	2
12	8	14	15	0	0	0	2
13	8	15	9	0	0	0	2
14	9	15	16	0	0	0	2
15	10	16	17	9	15	10	3
16	10	17	11	0	0	0	1
17	11	17	18	0	0	0	1
18	11	18	12	0	0	0	1
19	13	19	20	0	0	0	2
20	13	20	14	0	0	0	2
21	14	20	21	0	0	0	2
22	14	21	15	0	0	0	2
23	15	21	22	0	0	0	2
24	16	22	23	15	22	16	3
25	16	23	17	0	0	0	1
26	17	23	24	0	0	0	1
27	17	24	18	0	0	0	1
28	19	25	26	0	0	0	2
29	19	26	20	0	0	0	2
30	20	26	27	0	0	0	2
31	20	27	21	0	0	0	2
32	21	27	28	0	0	0	2
33	22	28	29	21	28	22	3
34	22	29	23	0	0	0	1
35	23	29	30	0	0	0	1
36	23	30	24	0	0	0	1
37	25	31	32	0	0	0	2
38	25	32	26	0	0	0	2
39	26	32	33	0	0	0	2
40	26	33	27	0	0	0	2
41	27	33	34	0	0	0	2
42	28	34	35	27	34	28	3
43	28	35	29	0	0	0	1

```
  44   29   35   35    0    0    0   1
  45   29   36   30    0    0    0   1
  46   31   37   38    0    0    0   2
  47   31   38   37    0    0    0   2
  48   32   35   39    0    0    0   2
  49   33   34   33    0    0    0   2
  50   33   39   43    0    0    0   2
  51   34   40   41   33   40   34   3
  52   34   41   35    0    0    0   1
  53   35   41   47    0    0    0   1
  54   35   42   35    0    0    0   1
  55   37   43   44    0    0    0   2
  56   37   44   38    0    0    0   2
  57   38   44   45    0    0    0   2
  58   38   45   39    0    0    0   2
  59   39   45   45    0    0    0   2
  60   40   45   47   39   45   40   3
  61   40   47   41    0    0    0   1
  62   41   47   48    0    0    0   1
  63   41   46   42    0    0    0   1
  64   43   49   50    0    0    0   2
  65   43   50   45    0    0    0   2
  66   44   50   51    0    0    0   2
  67   44   50   45    0    0    0   2
  68   45   51   52    0    0    0   2
  69   45   52   53   45   52   45   3
  70   46   53   47    0    0    0   1
  71   47   53   54    0    0    0   1
  72   47   54   49    0    0    0   1
BOUNDARY CONDITIONS
NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE
   1    0
SPECIFIED FREEDOMS
NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE NODE TYPE
   1  202    2  230    3  200    4  232    4   11    5   11    6   11    7    2   13    2   19    2   25    2   31    2
  37    2   43    2   49  202   50  230   51  200   52  232   52   11   53   11   54   11
SOLID-FLUID BOUNDARY CONSTRAINTS NODES ARE
  10   16   22   28   34   40   46
SYSTEM PARAMETER
    NOUT
     270
   DMAX
   0.0
   NODE      LOAD
    3        0.0        0.0
             0.0        0.0        0.0

SPECIFIED FREEDOMS OCCUR AT THE FOLLOWING DOF
     3      5      3     13     13     20     16     21     26     35     65     95    125    155    185    215    243    245
   248    253    258    250    256    261    266      0      0      0      0      0      0      0
INACTIVE FREEDOMS ARE LOCATED AT
     1      2      5      7     11     12     23     24     25     28     29     30     31     32     35     37     41     42
    53     54     55     56     59     50     61     62     55     67     71     72     83     34     85     33     89     90
    91     92     95     97    101    102    113    114    115    118    119    120    121    122    125    127    131    132
   143    144    145    148    149    150    151    152    155    157    161    162    173    174    175    173    179    180
   181    132    135    137    191    192    203    204    205    208    209    210    211    212    215    217    221    222
   233    234    235    238    239    240    241    242    245    247    251    252    263    254    255    253    269    270

     0      0      0      0      0      0      0
ANALYTE - INTEGRATION DOF NUMBERS ARE
     4      9     10     14     15     17     19     22     27     33     34     33     39     40     43     44     45     46
```

```
   47     48     49     50     51     52     56     57     58     64     68     69     70     73     74     75     76     77
   78     79     80     81     82     85     87     93     94     98     99    100    103    104    105    106    107    108
  109    110    111    112    116    117    123    124    125    129    130    133    134    135    135    137    138    139
  140    141    142    145    147    153    154    158    159    160    163    164    165    165    167    168    169    170
  171    172    175    177    183    184    188    189    190    193    194    195    196    197    198    199    200    201

  202    205    207    213    214    218    219    220    223    224    225    226    227    228    229    230    231    232
  235    237    244    249    250    251    252    254    255    257    259    262    263    264
THESE SPECIFIED FREEDOMS ARE
DISP(  1,   1)=      10.000

DISP(  1,   2)=       0.0

DISP(  1,   3)=       0.0

DISP(  2,   1)=      10.000

DISP(  2,   2)=       0.0

DISP(  2,   3)=       0.0

DISP(  3,   1)=      10.000

DISP(  3,   2)=       0.0

DISP(  3,   3)=       0.0

DISP(  4,   1)=      10.000

DISP(  4,   2)=       0.0

DISP(  4,   3)=       0.0

DISP(  5,   1)=       0.0

DISP(  5,   2)=       0.0

DISP(  5,   3)=       0.0

DISP(  6,   1)=       0.0

DISP(  6,   2)=       0.0

DISP(  6,   3)=       0.0

DISP(  7,   1)=       0.0

DISP(  7,   2)=       0.0

DISP(  7,   3)=       0.0

DISP(  8,   1)=       0.0

DISP(  8,   2)=       0.0

DISP(  8,   3)=       0.0

DISP(  9,   1)=       0.0

DISP(  9,   2)=       0.0
```

```
DISP(  9,  3)=        0.0
DISP( 10,  1)=        0.0
DISP( 10,  2)=        0.0
DISP( 10,  3)=        0.0
DISP( 11,  1)=        0.0
DISP( 11,  2)=        0.0
DISP( 11,  3)=        0.0
DISP( 12,  1)=        0.0
DISP( 12,  2)=        0.0
DISP( 12,  3)=        0.0
DISP( 13,  1)=        0.0
DISP( 13,  2)=        0.0
DISP( 13,  3)=        0.0
DISP( 14,  1)=        0.0
DISP( 14,  2)=        0.0
DISP( 14,  3)=        0.0
DISP( 15,  1)=        0.0
DISP( 15,  2)=        0.0
DISP( 15,  3)=        0.0
DISP( 16,  1)=        0.0
DISP( 16,  2)=        0.0
DISP( 16,  3)=        0.0
DISP( 17,  1)=        0.0
DISP( 17,  2)=        0.0
DISP( 17,  3)=        0.0
DISP( 18,  1)=        0.0
DISP( 18,  2)=        0.0
DISP( 18,  3)=        0.0
DISP( 19,  1)=        0.0
DISP( 19,  2)=        0.0
DISP( 19,  3)=        0.0
```

```
DISP( 20, 1)= 0.0
DISP( 20, 2)= 0.0
DISP( 20, 3)= 0.0
DISP( 21, 1)= 0.0
DISP( 21, 2)= 0.0
DISP( 21, 3)= 0.0
```

LOAD ARRAY

0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

```
FINITE ELEMENT ANALYSIS, INTEGRATION SOLUTION
SOLUTION FOR TIME T= 0.0
TIME INCREMENT DELT= 0.357714E-07
ITERATION NUMBER IT= 0
PRINTOUT VARIABLE N= 0
OUTPUT EVERY 700 TIMES
TIME AT FIRST ITERATION IS 0.357714E-07
DEBUG LEVEL 0
```

NODE	U X-DIRECTION DOF	VALUE	U Y-DIRECTION DOF	VALUE	PRESSURE DOF	VALUE	V X-DIRECTION DOF	VALUE	V Y-DIRECTION DOF	VALUE
1	1	0.0	2	0.0	3	0.10000E 02	4	0.0	5	0.0
2	6	0.0	7	0.0	8	0.10000E 02	9	0.0	10	0.0
3	11	0.0	12	0.0	13	0.10000E 02	14	0.0	15	0.0
4	16	0.0	17	0.0	18	0.10000E 02	19	0.0	20	0.0
5	21	0.0	22	0.0	23	0.0	24	0.0	25	0.0

6	26	0.0	27	0.0	28	0.0	29	0.0	30	0.0
7	31	0.0	32	0.0	33	0.0	34	0.0	35	0.0
8	36	0.0	37	0.0	38	0.0	39	0.0	40	0.0
9	41	0.0	42	0.0	43	0.0	44	0.0	45	0.0
10	46	0.0	47	0.0	48	0.0	49	0.0	50	0.0
11	51	0.0	52	0.0	53	0.0	54	0.0	55	0.0
12	56	0.0	57	0.0	58	0.0	59	0.0	60	0.0
13	61	0.0	62	0.0	63	0.0	64	0.0	65	0.0
14	66	0.0	67	0.0	68	0.0	69	0.0	70	0.0
15	71	0.0	72	0.0	73	0.0	74	0.0	75	0.0
16	76	0.0	77	0.0	78	0.0	79	0.0	80	0.0
17	81	0.0	82	0.0	83	0.0	84	0.0	85	0.0
18	86	0.0	87	0.0	88	0.0	89	0.0	90	0.0
19	91	0.0	92	0.0	93	0.0	94	0.0	95	0.0
20	96	0.0	97	0.0	98	0.0	99	0.0	100	0.0
21	101	0.0	102	0.0	103	0.0	104	0.0	105	0.0
22	106	0.0	107	0.0	108	0.0	109	0.0	110	0.0
23	111	0.0	112	0.0	113	0.0	114	0.0	115	0.0
24	116	0.0	117	0.0	118	0.0	119	0.0	120	0.0
25	121	0.0	122	0.0	123	0.0	124	0.0	125	0.0
26	126	0.0	127	0.0	128	0.0	129	0.0	130	0.0
27	131	0.0	132	0.0	133	0.0	134	0.0	135	0.0
28	136	0.0	137	0.0	138	0.0	139	0.0	140	0.0
29	141	0.0	142	0.0	143	0.0	144	0.0	145	0.0
30	146	0.0	147	0.0	148	0.0	149	0.0	150	0.0
31	151	0.0	152	0.0	153	0.0	154	0.0	155	0.0
32	156	0.0	157	0.0	158	0.0	159	0.0	160	0.0
33	161	0.0	162	0.0	163	0.0	164	0.0	165	0.0
34	166	0.0	167	0.0	168	0.0	169	0.0	170	0.0
35	171	0.0	172	0.0	173	0.0	174	0.0	175	0.0
36	176	0.0	177	0.0	178	0.0	179	0.0	180	0.0
37	181	0.0	182	0.0	183	0.0	184	0.0	185	0.0
38	186	0.0	187	0.0	188	0.0	189	0.0	190	0.0
39	191	0.0	192	0.0	193	0.0	194	0.0	195	0.0
40	196	0.0	197	0.0	198	0.0	199	0.0	200	0.0
41	201	0.0	202	0.0	203	0.0	204	0.0	205	0.0
42	206	0.0	207	0.0	208	0.0	209	0.0	210	0.0
43	211	0.0	212	0.0	213	0.0	214	0.0	215	0.0
44	216	0.0	217	0.0	218	0.0	219	0.0	220	0.0
45	221	0.0	222	0.0	223	0.0	224	0.0	225	0.0
46	226	0.0	227	0.0	228	0.0	229	0.0	230	0.0
47	231	0.0	232	0.0	233	0.0	234	0.0	235	0.0
48	236	0.0	237	0.0	238	0.0	239	0.0	240	0.0
49	241	0.0	242	0.0	243	0.0	244	0.0	245	0.0
50	246	0.0	247	0.0	248	0.0	249	0.0	250	0.0
51	251	0.0	252	0.0	253	0.0	254	0.0	255	0.0
52	256	0.0	257	0.0	258	0.0	259	0.0	260	0.0
53	261	0.0	262	0.0	263	0.0	264	0.0	265	0.0
54	266	0.0	267	0.0	268	0.0	269	0.0	270	0.0

FINITE ELEMENT ANALYT. INTEGRATION SOLUTION
SOLUTION FOR TIME T= 0.357134E-07
TIME INCREMENT DELT= 0.357134E-07
ITERATION NUMBER IT= 1
PRINTOUT VARIABLE N= 0
OUTPUT EVERY 700 TIMES
TIME AT FIRST ITERATION IS 0.357134E-07
DEBUG LEVEL 0

	U X-DIRECTION		U Y-DIRECTION		PRESSURE		V X-DIRECTION		V Y-DIRECTION	
NODE	DOF	VALUE	DOF	VALUE	DOF	VALUE	DOF	VALUE	DOF	VALUE
1	1	0.140930E-23	2	0.140930E-23	3	0.100000E 02	4	0.363558E-02	5	0.0
2	6	0.140930E-23	7	0.140930E-28	8	0.100000E 02	9	0.363568E-02	10	0.0

3	11	0.14090E-28	12	0.14090E-28	13	0.10000E 02	14	0.36358E-02	15	0.0
4	16	0.0	17	0.0	18	0.10000E 02	19	0.36358E-02	20	0.0
5	21	0.0	22	0.0	23	0.14090E-28	24	0.14090E-28	25	0.14090E-28
6	26	0.0	27	0.0	28	0.14090E-28	29	0.14090E-28	30	0.14090E-28
7	31	0.14170E-28	32	0.14090E-28	33	0.0	34	0.72716E-03	35	0.0
8	36	0.14090E-28	37	0.14090E-28	38	0.0	39	0.12119E-02	40	0.0
9	41	0.14090E-28	42	0.14090E-28	43	0.0	44	0.12119E-02	45	0.0
10	46	0.0	47	0.69917E-11	48	0.0	49	0.18179E-02	50	0.39139E-03
11	51	0.0	52	-0.39909E-15	53	0.14090E-28	54	0.14090E-28	55	0.14090E-28
12	56	0.0	57	0.0	58	0.14090E-28	59	0.14090E-28	60	0.14090E-28
13	61	0.14090E-28	62	0.14090E-28	63	0.0	64	0.0	65	0.0
14	66	0.14090E-28	67	0.14090E-28	68	0.0	69	0.0	70	0.0
15	71	0.14090E-28	72	0.14090E-28	73	0.0	74	0.0	75	0.0
16	76	0.0	77	0.0	78	0.0	79	0.0	80	0.0
17	81	0.0	82	0.0	83	0.14090E-28	84	0.14090E-28	85	0.14090E-28
18	86	0.0	87	0.0	88	0.14090E-28	89	0.14090E-28	90	0.14090E-28
19	91	0.14074E-28	92	0.14090E-28	93	0.0	94	0.0	95	0.0
20	96	0.14090E-28	97	0.14090E-28	98	0.0	99	0.0	100	0.0
21	101	0.14090E-28	102	0.14090E-28	103	0.0	104	0.0	105	0.0
22	106	0.0	107	0.0	108	0.0	109	0.0	110	0.0
23	111	0.0	112	0.0	113	0.14090E-28	114	0.14090E-28	115	0.14090E-28
24	116	0.0	117	0.0	118	0.14090E-28	119	0.14090E-28	120	0.14090E-28
25	121	0.14090E-28	122	0.14090E-28	123	0.0	124	0.0	125	0.0
26	126	0.14090E-28	127	0.14090E-28	128	0.0	129	0.0	130	0.0
27	131	0.14080E-28	132	0.14090E-28	133	0.0	134	0.0	135	0.0
28	136	0.0	137	0.0	138	0.0	139	0.0	140	0.0
29	141	0.0	142	0.0	143	0.14090E-28	144	0.14090E-28	145	0.14090E-28
30	146	0.0	147	0.0	148	0.14090E-28	149	0.14090E-28	150	0.14090E-28
31	151	0.14040E-28	152	0.14090E-28	153	0.0	154	0.0	155	0.0
32	156	0.14090E-28	157	0.14090E-28	158	0.0	159	0.0	160	0.0
33	161	0.14090E-28	162	0.14090E-28	163	0.0	164	0.0	165	0.0
34	166	0.0	167	0.0	168	0.0	169	0.0	170	0.0
35	171	0.0	172	0.0	173	0.14090E-28	174	0.14090E-28	175	0.14090E-28
36	176	0.0	177	0.0	178	0.14090E-28	179	0.14090E-28	180	0.14090E-28
37	181	0.14090E-28	182	0.14090E-28	183	0.0	184	0.0	185	0.0
38	186	0.14090E-28	187	0.14090E-28	188	0.0	189	0.0	190	0.0
39	191	0.14090E-28	192	0.14090E-28	193	0.0	194	0.0	195	0.0
40	196	0.0	197	0.0	198	0.0	199	0.0	200	0.0
41	201	0.0	202	0.0	203	0.14090E-28	204	0.14090E-28	205	0.14090E-28
42	206	0.0	207	0.0	208	0.14090E-28	209	0.14090E-28	210	0.14090E-28
43	211	0.14090E-28	212	0.14090E-28	213	0.0	214	0.0	215	0.0
44	216	0.14090E-28	217	0.14090E-28	218	0.0	219	0.0	220	0.0
45	221	0.14090E-28	222	0.14090E-28	223	0.0	224	0.0	225	0.0
46	226	0.0	227	0.0	228	0.0	229	0.0	230	0.0
47	231	0.0	232	0.0	233	0.14090E-28	234	0.14090E-28	235	0.14090E-28
48	236	0.0	237	0.0	238	0.14090E-28	239	0.14090E-28	240	0.14090E-28
49	241	0.14090E-28	242	0.14090E-28	243	0.0	244	0.0	245	0.0
50	246	0.14090E-28	247	0.14090E-28	248	0.0	249	0.0	250	0.0
51	251	0.14090E-28	252	0.14090E-28	253	0.0	254	0.0	255	0.0
52	256	0.0	257	0.0	258	0.0	259	0.0	260	0.0
53	261	0.0	262	0.0	263	0.14090E-28	264	0.14090E-28	265	0.14090E-28
54	266	0.0	267	0.0	268	0.14090E-28	269	0.14090E-28	270	0.14090E-28

```
FINITE ELEMENT ANALYS. INTEGRATION SOLUTION
SOLUTION FOR TIME T= 0.24999E-04
TIME INCREMENT DELT= 0.35714E-07
ITERATION NUMBER (I= 700)
PRINTOUT VARIABLE N= 700
OUTPUT EVERY 700 TIMES
TIME AT FIRST ITERATION IS 0.35714E-07
DEBUG LEVEL 0
```

	U X-DIRECTION		U Y-DIRECTION		PRESSURE		V X-DIRECTION		W Y-DIRECTION	
NODE	DOF	VALUE	DOF	VALUE	DOF	VALUE	DOF	VALUE	DOF	VALUE
1	1	0.14090E-28	2	0.14090E-28	3	0.10000E 02	4	0.17377E 01	5	0.0
2	6	0.14090E-28	7	0.14090E-28	8	0.10000E 02	9	0.15587E 01	10	0.55560E-02
3	11	0.14090E-28	12	0.14090E-28	13	0.10000E 02	14	0.15873E 01	15	-0.48917E-02
4	16	0.0	17	0.0	18	0.10000E 02	19	0.12375E 01	20	0.0
5	21	0.0	22	0.0	23	0.14090E-28	24	0.14090E-28	25	0.14090E-28
6	26	0.0	27	0.0	28	0.14090E-28	29	0.14090E-28	30	0.14090E-28
7	31	0.14090E-28	32	0.14090E-28	33	0.13354E 01	34	0.28053E 00	35	0.0
8	36	0.14090E-28	37	0.14090E-28	38	0.19793E 01	39	0.41130E 00	40	-0.79871E-02
9	41	0.14090E-28	42	0.14090E-28	43	0.17955E 01	44	0.39113E 00	45	-0.37749E-03
10	46	0.31343E-03	47	0.39257E-05	48	0.20559E 01	49	0.57381E 00	50	0.75407E-01
11	51	0.95449E-03	52	0.99057E-05	53	0.14090E-28	54	0.14090E-28	55	0.14090E-28
12	56	-0.15742E-03	57	0.97554E-05	58	0.14090E-28	59	0.14090E-28	60	0.14090E-28
13	61	0.14090E-28	62	0.14090E-28	63	-0.39997E-01	64	-0.98247E-02	65	0.0
14	66	0.14090E-28	67	0.14090E-28	68	-0.13522E 00	69	-0.53774E-01	70	-0.22482E-02
15	71	0.14090E-28	72	0.14090E-28	73	-0.21759E 00	74	-0.54705E-01	75	0.48478E-02
16	76	-0.21643E-07	77	0.55752E-07	78	-0.25293E 00	79	-0.14971E 00	80	0.10454E-01
17	81	0.79173E-08	82	0.55335E-07	83	0.14090E-28	84	0.14090E-28	85	0.14090E-28
18	86	0.47422E-07	87	0.55532E-07	88	0.14090E-28	89	0.14090E-28	90	0.14090E-28
19	91	0.14090E-28	92	0.14090E-28	93	0.10322E-01	94	-0.13573E-02	95	0.0
20	96	0.14090E-28	97	0.14090E-28	98	-0.17308E-01	99	0.35993E-02	100	0.17242E-02
21	101	0.14090E-28	102	0.14090E-28	103	0.11247E-01	104	0.11172E-01	105	0.13955E-03
22	106	-0.10341E-03	107	-0.21376E-07	108	-0.57572E-11	109	0.35535E-01	110	-0.32338E-02
23	111	0.98775E-09	112	-0.21242E-07	113	0.14090E-28	114	0.14090E-28	115	0.14090E-28
24	116	0.23558E-09	117	-0.21145E-07	118	0.14090E-28	119	0.14090E-28	120	0.14090E-28
25	121	0.14090E-28	122	0.14090E-28	123	-0.55056E-02	124	0.99714E-03	125	0.0
26	126	0.14090E-28	127	0.14090E-28	128	0.73054E-02	129	-0.11283E-02	130	-0.13795E-03
27	131	0.14090E-28	132	0.14090E-28	133	0.47359E-02	134	-0.13150E-02	135	-0.11556E-02
28	136	0.50585E-08	137	0.25644E-08	138	0.51909E-11	139	-0.10496E-01	140	0.31450E-03
29	141	0.11538E-10	142	0.24407E-08	143	0.14090E-28	144	0.14090E-28	145	0.14090E-28
30	146	-0.57285E-09	147	0.23838E-08	148	0.14090E-28	149	0.14090E-28	150	0.14090E-28
31	151	0.14090E-28	152	0.14090E-28	153	0.22993E-02	154	-0.33156E-03	155	0.0
32	156	0.14090E-28	157	0.14090E-28	158	-0.11151E-02	159	0.11511E-03	160	-0.12030E-03
33	161	0.14090E-28	162	0.14090E-28	163	0.95531E-04	164	-0.76907E-04	165	0.55587E-03
34	166	-0.44929E-10	167	-0.63903E-09	168	-0.21551E-11	169	0.31551E-02	170	-0.31415E-04
35	171	0.23934E-10	172	-0.60250E-09	173	0.14090E-28	174	0.14090E-28	175	0.14090E-28
36	176	0.95234E-10	177	-0.53395E-09	178	0.14090E-28	179	0.14090E-28	180	0.14090E-28
37	181	0.14090E-28	182	0.14090E-28	183	-0.50594E-03	184	0.92175E-04	185	0.0
38	186	0.14090E-28	187	0.14090E-28	188	0.58439E-04	189	0.93303E-05	190	0.72507E-04
39	191	0.14090E-28	192	0.14090E-28	193	-0.10535E-02	194	0.15324E-03	195	-0.19359E-03
40	196	0.10835E-10	197	0.57015E-10	198	0.55116E-02	199	-0.89750E-03	200	0.57542E-05
41	201	-0.54550E-11	202	0.52895E-10	203	0.14090E-28	204	0.14090E-28	205	0.14090E-28
42	206	-0.25245E-10	207	0.51452E-10	208	0.14090E-28	209	0.14090E-28	210	0.14090E-28
43	211	0.14090E-28	212	0.14090E-28	213	0.24951E-03	214	-0.37010E-04	215	0.0
44	216	0.14090E-28	217	0.14090E-28	218	-0.90739E-04	219	-0.79457E-05	220	-0.40719E-04
45	221	0.14090E-28	222	0.14090E-28	223	0.72702E-03	224	-0.11534E-03	225	0.72752E-04
46	226	-0.15472E-11	227	0.50015E-10	228	-0.15837E-02	229	0.25157E-03	230	0.91171E-05
47	231	0.44034E-12	232	0.56321E-10	233	0.14090E-28	234	0.14090E-28	235	0.14090E-28
48	236	0.22993E-11	237	0.57534E-10	238	0.14090E-28	239	0.14090E-28	240	0.14090E-28
49	241	0.14090E-28	242	0.14090E-28	243	0.0	244	0.39122E-04	245	0.0
50	246	0.14090E-28	247	0.14090E-28	248	0.0	249	-0.57305E-05	250	0.21515E-04
51	251	0.14090E-28	252	0.14090E-28	253	0.0	254	0.93542E-04	255	-0.25547E-04
52	256	0.0	257	0.0	258	0.0	259	-0.75467E-04	260	0.0
53	261	0.0	262	0.0	263	0.14090E-28	264	0.14090E-28	265	0.14090E-28
54	266	0.0	267	0.0	268	0.14090E-28	269	0.14090E-28	270	0.14090E-28

VITA

Rahim Falsafi-haghighi was born in on . In June,1965,he was awarded a Bachelor of Science degree from The American University of Beirut,Beirut,Lebanon. He received a Master of Science degree from University of Maine,Orono,Maine in January 1968. He started working toward the Doctor of Engineering Science degree in Mechanical Engineering at New Jersey Institute of Technology,Newark College of Engineering,Newark,New Jersey,in September 1972.

The research upon which this dissertation is based was conducted at New jersey Institute of Technology from November 1976 to August 1978.