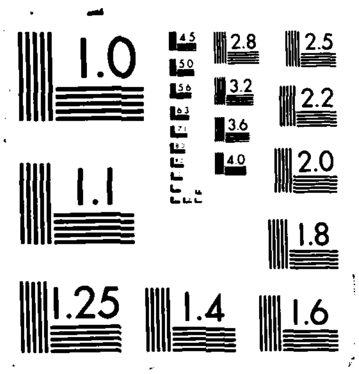
AD-A181 193 THE FRAM CAVITATION CODE(U) WEIDLINGER ASSOCIATES NEW YORK M REHAK ET AL 15 MAY 86 WA-86-7 DNA-TR-86-179 DNA001-84-C-0001 1/1

UNCLASSIFIED F/G 12/5 NL

END
7 87
DTIC


1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

AD-A181 193

(12)

DNA-TR-86-179

THE FRAM CAVITATION CODE

M. Rehak
R. Smilowitz
R. Kagel
Weidlinger Associates
Consulting Engineers
333 Seventh Avenue
New York, NY 10001

15 May 1986

Technical Report

CONTRACT No. DNA 001-84-C-0001

THIS WORK WAS SPONSORED BY THE DEFENSE NUCLEAR AGENCY
UNDER RDT&E RMSS CODE B344085466 Y99QMXSF00045 H2590D.

Prepared for
Director
DEFENSE NUCLEAR AGENCY
Washington, DC 20305-1000

CUT HERE AND RETURN

DISTRIBUTION LIST UPDATE

This mailer is provided to enable DNA to maintain current distribution lists for reports. We would appreciate your providing the requested information.

☐ Add the individual listed to your distribution list.

☐ Delete the cited organization/individual.

☐ Change of address.

NAME: ______________________________

ORGANIZATION: ______________________________

OLD ADDRESS	CURRENT ADDRESS
______________________	______________________
______________________	______________________
______________________	______________________

TELEPHONE NUMBER: (   ) ______________________________

SUBJECT AREA(s) OF INTEREST:

______________________	______________________
______________________	______________________
______________________	______________________

DNA OR OTHER GOVERNMENT CONTRACT NUMBER: ______________________

CERTIFICATION OF NEED-TO-KNOW BY GOVERNMENT SPONSOR (if other than DNA):

SPONSORING ORGANIZATION: ______________________

CONTRACTING OFFICER OR REPRESENTATIVE: ______________________

SIGNATURE: ______________________

Director
Defense Nuclear Agency
ATTN: [redacted] TITL
Washington, DC 20305-1000

Director
Defense Nuclear Agency
ATTN: [redacted] TITL
Washington, DC 20305-1000

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

ADA181193

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY N/A since Unclassified 2b DECLASSIFICATION/DOWNGRADING SCHEDULE N/A since Unclassified	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
4 PERFORMING ORGANIZATION REPORT NUMBER(S) WA 86-7	5. MONITORING ORGANIZATION REPORT NUMBER(S) DNA-TR-86-179
6a NAME OF PERFORMING ORGANIZATION Weidlinger Associates Consulting Engineers 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Director Defense Nuclear Agency
6c. ADDRESS (City, State, and ZIP Code) 333 Seventh Avenue New York, NY 10001	7b ADDRESS (City, State, and ZIP Code) Washington, DC 20305-1000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable) SPSS/Tsai	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DNA 001-84-C-0001
8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO 62715H; PROJECT NO Y99QMXS; TASK NO F; WORK UNIT ACCESSION NO DH008838

11 TITLE (Include Security Classification)
THE FRAM CAVITATION CODE

12 PERSONAL AUTHOR(S)
Rehak, M.; Smilowitz, R. and Kagel, R.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year Month Day)	15 PAGE COUNT
Technical	FROM 851031 TO 860515	860515	32

16 SUPPLEMENTARY NOTATION
This work was sponsored by the Defense Nuclear Agency under RDT&E RMSS Code B344085466 Y99QMXSF00045 H2590D.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Bulk Cavitation; Underwater Explosions; Characteristic Analysis; Surface Ship Response
13	10		
19	9		

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

This report describes the computer code FRAM. The code computes the pressure and velocity in a fluid-half space where bulk cavitation may occur. The cavitation results from an exponentially decaying pressure wave from an underwater explosion which travels to the surface and is reflected as a tension wave. Because the water does not sustain tension, a cavitated region is formed. The cavity subsequently closes under gravity (lower closing shock) and accretion of vapor particles (upper closing shock).

The theory is summarized herein. Capabilities and limitations are discussed and a description of the source code is included.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL Sandra E. Young	22b TELEPHONE (Include Area Code) (202) 325-7042 — 22c OFFICE SYMBOL DNA/CSTI

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

CONVERSION TABLE

Conversion factors for U.S. Customary to metric (SI) units of measurement.

MULTIPLY ——————→ BY ——————→ TO GET
TO GET ←—————— BY ←—————— DIVIDE

angstrom	1.000 000 X E -10	meters (m)
atmosphere (normal)	1.013 25 X E +2	kilo pascal (kPa)
bar	1.000 000 X E +2	kilo pascal (kPa)
barn	1.000 000 X E -28	meter² (m^2)
British thermal unit (thermochemical)	1.054 350 X E +3	joule (J)
calorie (thermochemical)	4.184 000	joule (J)
cal (thermochemical)/cm^2	4.184 000 X E -2	mega joule/m^2 (MJ/m^2)
curie	3.700 000 X E +1	giga becquerel (GBq)*
degree (angle)	1.745 329 X E -2	radian (rad)
degree Fahrenheit	$t_K = (t°f + 459.67)/1.8$	degree kelvin (K)
electron volt	1.602 19 X E -19	joule (J)
erg	1.000 000 X E -7	joule (J)
erg/second	1.000 000 X E -7	watt (W)
foot	3.048 000 X E -1	meter (m)
foot-pound-force	1.355 818	joule (J)
gallon (U.S. liquid)	3.785 412 X E -3	meter³ (m^3)
inch	2.540 000 X E -2	meter (m)
jerk	1.000 000 X E +9	joule (J)
joule/kilogram (J/kg) (radiation dose absorbed)	1.000 000	Gray (Gy)**
kilotons	4.183	terajoules
kip (1000 lbf)	4.448 222 X E +3	newton (N)
kip/inch² (ksi)	6.894 757 X E +3	kilo pascal (kPa)
ktap	1.000 000 X E +2	newton-second/m^2 ($N\text{-}s/m^2$)
micron	1.000 000 X E -6	meter (m)
mil	2.540 000 X E -5	meter (m)
mile (international)	1.609 344 X E +3	meter (m)
ounce	2.834 952 X E -2	kilogram (kg)
pound-force (lbf avoirdupois)	4.448 222	newton (N)
pound-force inch	1.129 848 X E -1	newton-meter (N·m)
pound-force/inch	1.751 268 X E +2	newton/meter (N/m)
pound-force/foot²	4.788 026 X E -2	kilo pascal (kPa)
pound-force/inch² (psi)	6.894 757	kilo pascal (kPa)
pound-mass (lbm avoirdupois)	4.535 924 X E -1	kilogram (kg)
pound-mass-foot² (moment of inertia)	4.214 011 X E -2	kilogram-meter² ($kg \cdot m^2$)
pound-mass/foot³	1.601 846 X E +1	kilogram/meter³ (kg/m^3)
rad (radiation dose absorbed)	1.000 000 X E -2	Gray (Gy)**
roentgen	2.579 760 X E -4	coulomb/kilogram (C/kg)
shake	1.000 000 X E -8	second (s)
slug	1.459 390 X E +1	kilogram (kg)
torr (mm Hg, 0°C)	1.333 22 X E -1	kilo pascal (kPa)

* The becquerel (Bq) is the SI unit of radioactivity; 1 Bq = 1 event/s.
**The Gray (Gy) is the SI unit of absorbed radiation.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

SECTION 1

CAPABILITIES OF FRAM

FRAM (No acronym) is a computer code which computes the pressure and the velocity of a particle in a fluid half space where cavitation may occur. The cavitation is assumed to result from an exponentially decaying compression wave which travels to the surface and is reflected as a tension wave. Because the water does not sustain tension a cavitated region is formed. The cavity subsequently closes under gravity (lower closing shock) and accretion of vapor particles (upper closing shock).

The model of the fluid and the mathematical method used to determine the boundaries of the cavitated region are described in ref [1]. **FRAM** computes the *free-field solution*, that is, the field variables when no structure is present. The problem of the interaction with a floating structure requires the use of an approximate interactive scheme such as the plane wave approximation (*P.W.A.*) or the doubly assymptotic approximation(*D.A.A.*) in ref. [1].

FRAM does not treat bottom reflections and requires the explosion to be located a great distance away from the observation point in order to approximate the spherical waves by plane waves. It treats the incident wave as plane and steady state with respect to a system of coordinates moving with the wavefront.

The one dimensional characteristics method is used. It is applied to the two dimensional problem by transforming coordinates (X, Y, Z, T) into (y, ξ) where $y = Z, \xi = \sqrt{(X^2 + Y^2)} + sT$. Combining two or three parameters into ξ is done under the steady-state assumption that the cavity is unchanged with respect to a coordinate system moving with the apparent wave speed (s). Conservation of momentum and continuity of velocity across the closing shocks together with characteristic relations form the governing system of equations .

The analytic solution provided by **FRAM** has the advantage over numerical methods that is that it requires comparatively small computer resources and sharper wave fronts are obtained. The shortcomings result mainly from the inability to include bottom reflection effects and the divergence and decay associated with propagation away from the explosive source.

SECTION 2

DESCRIPTION OF THE CODE

2.1 PROGRAM ORGANIZATION.

Given the parameters of the explosion and of the fluid, the first step consists in finding the geometry of the cavitated region. Along with the geometry, pressure and velocity on the boundary are also computed. The second step takes the coordinates (y, ξ) of a point of observation in the fluid and computes its pressure and velocity. This is repeated for each time increment to construct a time history.

The main program calls the subroutine CAVITY which computes the geometry of the cavitated region. It is formed by three curves: *AB, BC, AC* (see figure 1, section 3). Having charted the geometry and physical properties of the free-field, one can proceed with the determination of the properties of a point of interest. Subroutine ECHO takes the coordinates of the point, finds its location with respect to the cavity using subroutine ZONE, and finally computes pressure and velocity at that point. Finally functions G and GG correspond to the characteristic functions g and to its derivative g' along the Y axis.

There are two types of parameters that must be specified: those pertaining to the material properties (c speed of sound in water , p_A atmospheric pressure, γ_0 fluid density, g gravitational accelerarion) and to the explosion (L decay length,p_S peak pressure, β decay constant) , and the coordinates (y, ξ) of the points of observation. All parameters and variables are given in a non–dimensional formulation. In the example, U.S. customary units are used. The output of the program consists in the curves AB,AC,BC forming the cavity, and in time histories of p, m, n.

2.2 DESCRIPTION OF SUBROUTINES.

The present version of the code is designed to interact with the finite element code **SAP**. A model for a ship provides the location of the wet nodes at which pressure and velocity histories are computed and input into **SAP**. The **SAP** input is read from tape 14 and written on tape 9. It is not difficult however to extract those subroutines relevant to the free–field only for use of **FRAM** with other codes. The subroutines interacting with **SAP** are **READIT, EMPIR, RESTWR.**

2.2.1 FRAM.

The main program calls **READIT** where some of the parameters are defined and the coordinates of the point of observation are provided by **SAP**. NWET is the number of wet nodes, KTIM is the number of time history points. The explosion's parameters are obtained in part from empirical data in **EMPIR**. **CAVITY** is called for each point of observation and **ECHO** is called for each time step. The output is written for **SAP** in **RESTWR**.

Although running the code does not require a complete understanding of the theory, references to equations and figures in ref.[1] have been included for the sake of completeness. The correspondence between some of the symbols in ref. [1] and in **FRAM** is as follows:

CC=c, PS= p_S, L=L, PA= p_A, GAM= γ_0, GV= g , T= θ, TT= β, AA= α, AL= αl, YB=y_B, SB=ξ_B, YC=y_C, SC=ξ_C, YA=y_A, SA=ξ_A, FH=$\frac{g}{c^3}$, F=$\frac{2p_S}{\alpha l}$, XI(1)=χ_B, XI(2)=χ_A, R=R,E=$\Delta\xi$, A=A, AH=f_U, GR=g_T, ST=ξ, HS=$\Delta\xi$, YX=y, PRE=ρ, VEL=m, UEL=n, F=F, G=G.

2.2.2 CAVITY.

In the following, references to equations and figures found in ref.[1] will appear in parentheses. This subroutine starts by defining some parameters in order to use a non–dimensional formulation. The coordinates of point B are given by finding the first point on the opening interface AB where the density has a change in sign due to closing of the cavity (eq (39)). The coordinates of point A are found on AB where the pressure first drops to zero (eq(51)) using subroutine **NEWTON**. The slopes of AC and BC at points A and B are also evaluated using Taylor series expansions (eqs(90) and(83)).

Next, curves AB and BC are constructed incrementally, with an increment size E which is determined such that ξ_B coincides with the ξ value of one of the discrete points of curve AC. The curves AC and BC are constructed one point at a time. Starting from the coordinates of a point I and the slope at that point, one can determine the coordinates of the next point, I+1 and the slope of the next increment by a first order linear approximation.

The array D(I,J) contains the properties of the curve's discrete points. J=1 corresponds to curve BC or lower curve and J=2 to curve AC or upper curve. I is the index of the discrete points; there are KK more points on AC than on BC. D(I,1) is the characteristic f_I (eq(97)), D(I,2) is the characteristic g_I (eq(99)).

Additional arrays which do not appear in ref. [1] are M(I) and P(I). These represent M=$-y+\theta\xi$, P=$y+\theta\xi$ and are used to determine the location of the point of observation with respect to the cavity, i.e. between which of the discrete points of the cavity the characteristic functions defining the properties of the point of interest fall.

S is augmented at each step by an increment and Y(I) is the result of the previous cycle of computations(eq(69)). For the upper curve, the characteristic emanating from I is reflected on the free surface at U and intersects the y axis at T or the curve AC itself at R see (fig.(8)in ref[1]). For the lower curve, R is defined as the intersection of the characteristic emanating from point I and the y axis (fig.(7.a) in ref[1]).

The upper curve is constructed first, it is more complex than the lower one since the characteristics are reflected on the free surface and fall back on the portion of the curve freshly constructed at point R. The properties at point I+1 are deduced from those at point I using conservation equations and geometry (eqs(69) and (71)) in which the expressions for f_U, g_W, g'_W are needed. If R falls on the y axis,then it corresponds to point T in fig.(7.c) of ref. [1], f_U is determined from geomtricalrealtions (eq(73)). If R falls on the beginning of AC , then an interpolation between the two adjacent points is required to evaluate f_U. The remaining two terms, g_W, g'_W are simply found by calling the functions g, g' with the argument y_W. g is given by its definition (eqs(27),(33), and(34) for a point on the y axis).

The construction of the lower curve requires the evaluation of a similar equation expressing the conservation laws and geometrical relations (eq(68)), and in particular g_W, g'_W, g_T which is simply done by calling the functions.

This procedure is repeated until the two curves intersect at point C.

2.2.3 ECHO.

In order to obtain p, m at y, ξ it is sufficient to know f, g (eqs.(102), (103)). Using geometry and characteristic relations, the characteristics at X are expressed in terms of similar characteristics at points where these are known. Fig.(8) in ref[1] shows the possible zones in which the point may fall. A zone is the ensemble of points with same expressions for the generalized Riemann invariants (Table.1). **ECHO** calls **ZONE** where the zone number is found and pressure and velocity are computed accordingly. Results are converted in dimensional quantities and stored in O(NT,J) for plotting.

2.2.4 ZONE.

Z is the zone number, F and G are defined in Table 1.. The coordinates of the points defined in Table 2. are computed first. Then a test is performed to determine whether X belongs to a given zone. F and G are assigned appropriate values which allows the computation of p, m. In some cases it is necessary to call **SEARCH** to locate X.

2.2.5 SEARCH.

Using the arrays P, and M created in **CAVITY** this routine finds between which points of AC or BC the characteristics emanating from X falls. HFG is either F or G. When X falls in 5 or 7, then a linear interpolation between adjacent discrete points of the boundaries is performed.

SECTION 3

EXAMPLE

Examples of output are presented in the following figures. An example of the cavitated region is shown in Fig.1.

The location of the charge relative to the submerged surface of the ship and origin of SAP coordinates appear in Fig.2. The free-field values for the submerged nodes 8, 18, and 28 which are specified within SAP as nodes 33, 43, and 53 appear in Fig.3.

Fig.3 shows in (a) the free-field pressure which is characterized by a pressure pulse followed by a surface relief effect (pressure drop below zero line) and terminated by a secondary pulse due to the closure of the cavitated region. The integral of the pressure is the impulse which is plotted in (b). One can follow in (c) the correlation between vertical free-field velocities and pressures of (a). The initial upward velocity is due to the first spike in (a). A free fall under gravity inside the cavitated region follows and is terminated due to the increase in pressure resulting from the cavitation closure (second spike in (a)). These free-field velocities are integrated to give displacements shown in (d).

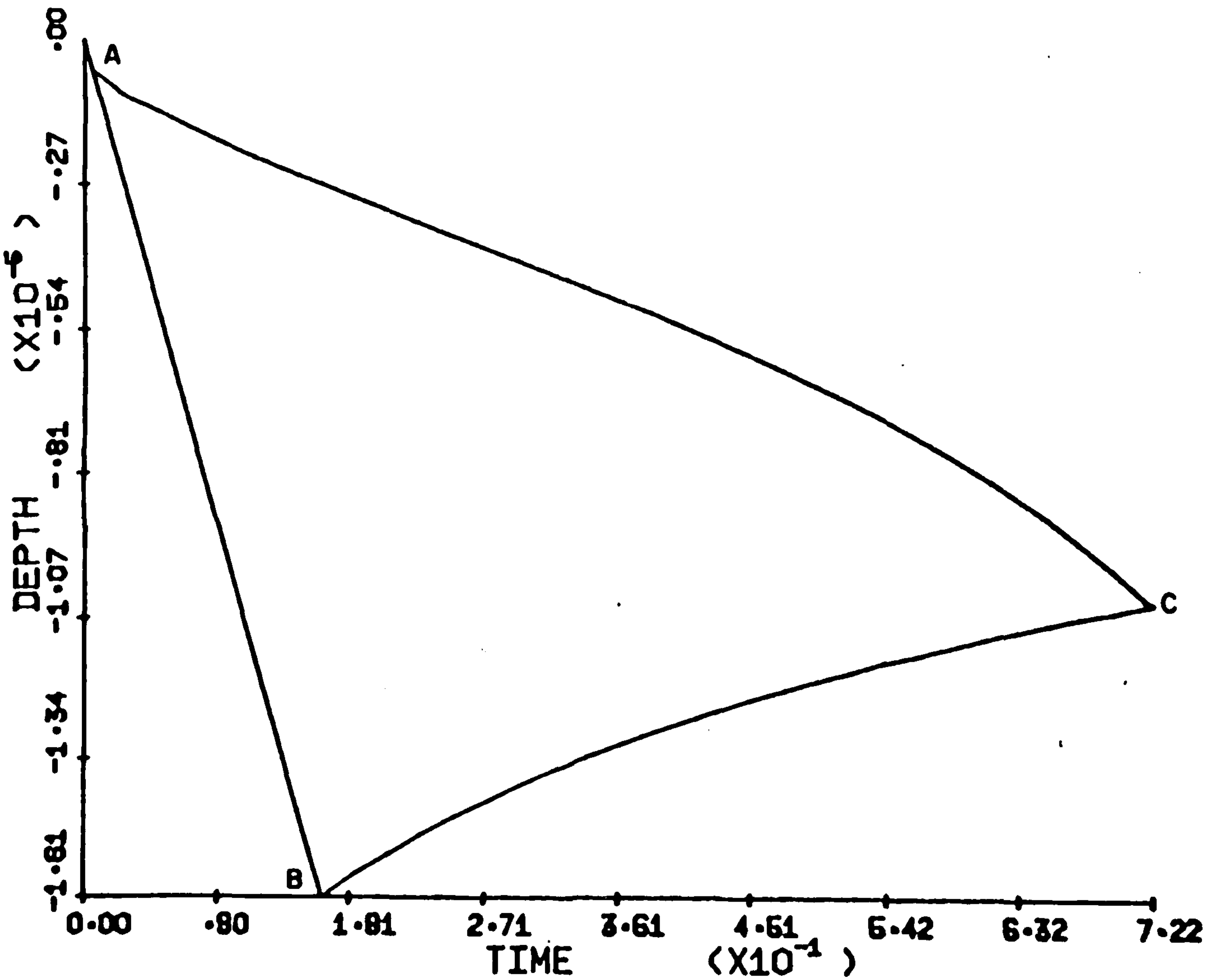


Figure 1. Cavitated region non-dimensional units.

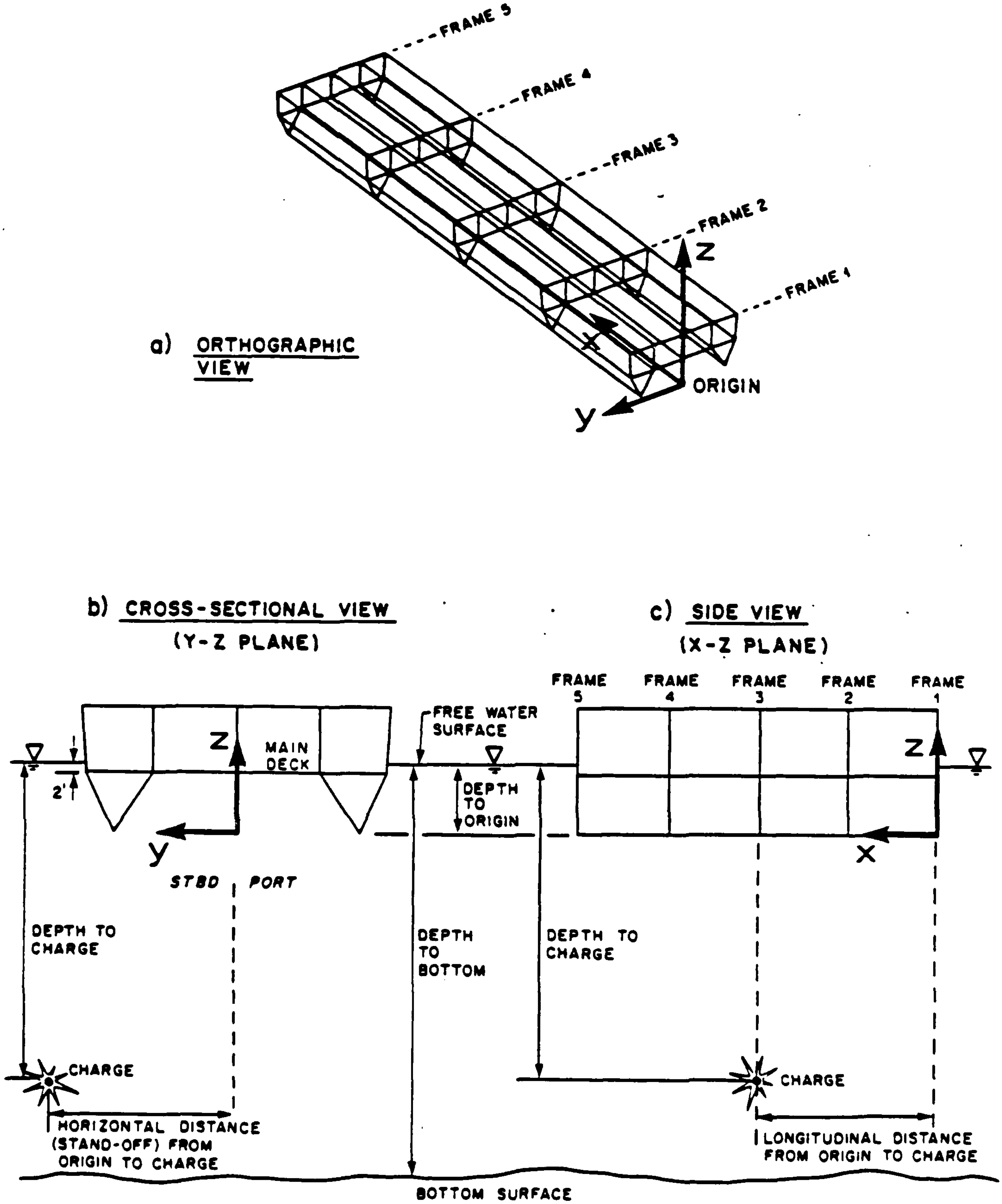


Figure 2. Global coordinate system and orientation of surface ship to underwater explosion for sample problem.

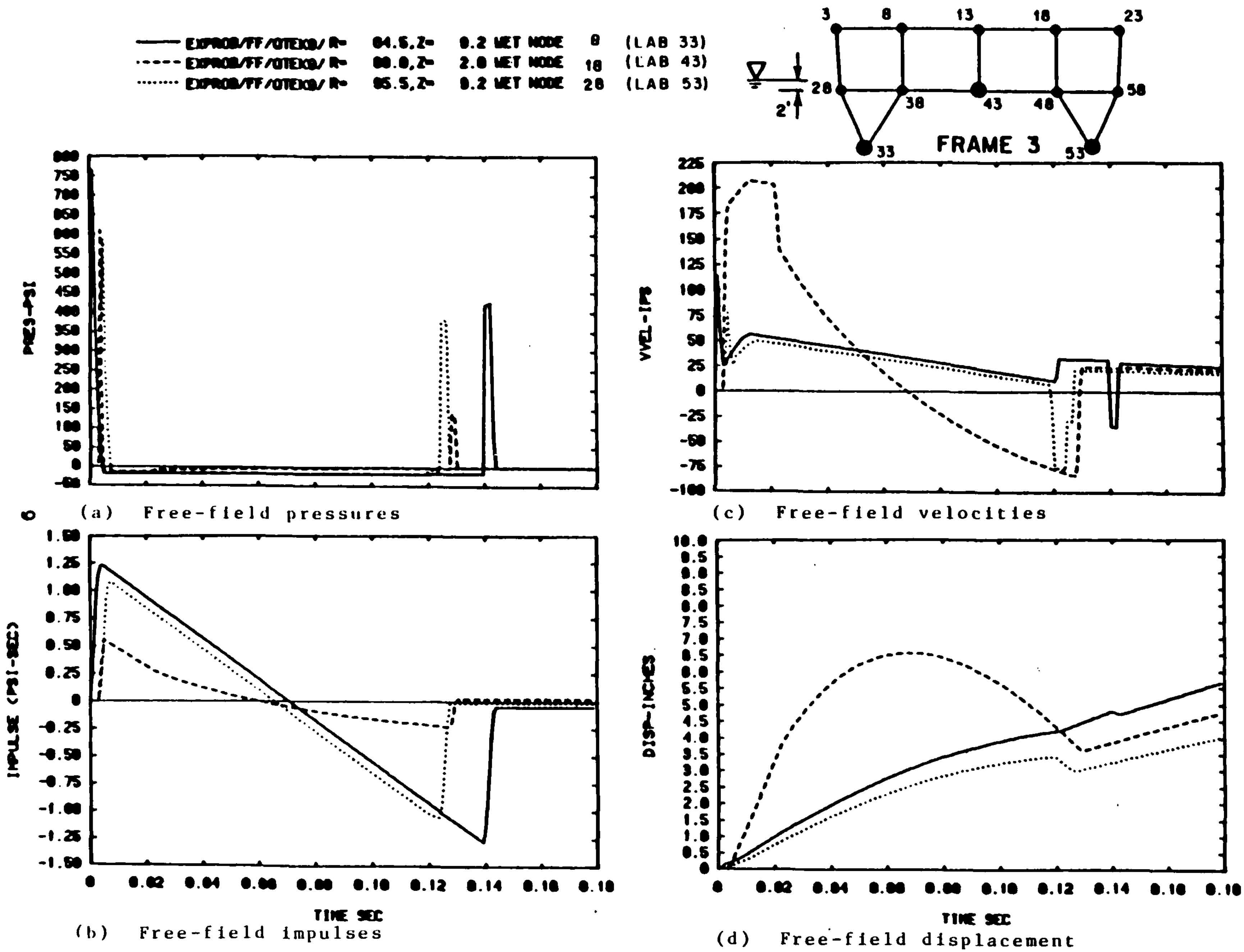


Figure 3. Free-field pressure (a), free-field impulse (b), free-field velocity (c) and free-field displacement (d) for wet nodes 8, 18 and 28 on frame 3.

SECTION 4

SOURCE CODE

```
      PROGRAM FRAM (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT
     *,TAPE9,TAPE14,TAPE99)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      REAL L,M
      CALL READIT
      DO 2 MN=1,NWET
      CALL EMPIR(MN)
C      GENERATE THE  CAVITY"S BOUNDARY
      CALL CAVITY
      DO 1 NT=1,KTIM
      CALL ECHO(NT,MN)
1     CONTINUE
      CALL RESTWR(MN)
2     CONTINUE
      REWIND 99
      STOP
      END
      SUBROUTINE READIT
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      REAL L,M
      GRAV=0.0000322
      GAM=1.94
      CC=5236.
      GV=32.2
      RHO=62.4/144./GRAV
      PEDFAC=RHO*CC*SQRT(8.)
      RMIN=1.E9
      RMAX=0.
      MIN=1
      FH=GV/CC**2
      WRITE(6,75)CC,GV,FH
75    FORMAT(" CC,GV,FH",3E10.3)
      REWIND 14
      READ(14)W,DD,H,IBOT,NWET,ZSURF,XXI,SSS,DT
       IF(DT.LT.0.01) DT = DT*1000.
C     REWIND 9
```

```
      WRITE(6,11)
11    FORMAT(" W,D,H,IBOT,NWET,ZSURF,XI,SSS,KTIM,DT",/)
      WRITE(6,9)W,DD,H,IBOT,NWET,ZSURF,XXI,SSS,KTIM,DT,FH
9     FORMAT(3F10.3,2I5,3F10.3,I5,2E10.3)
      WRITE(9) W,DD,H,IBOT,NWET,ZSURF,XXI,SSS,DT,KTIM
      ENCODE(48,999,TITLE)W,DD,XXI/12.0,-SSS/12.0
999   FORMAT(4H  W=,F8.0,7H DEPTH=,F6.1,4H XI=,F6.1,7H STAND=,F6.1)
      XXI = -XXI
      CALL REMARK(TITLE)
      DO 10 I=1,NWET
      READ(14)IDUM,XSAP,YSAP,ZSAP
C     WRITE(6,78)XSAP,YSAP,ZSAP
78    FORMAT(" X,Y,Z",3E10.3)
      RHIS(I)=(SQRT((XSAP+XXI)**2+(YSAP+SSS)**2))/12.
      ZHIS(I)=AMAX1(0.,(ZSURF-ZSAP)/12.)
C     WRITE(6,77)RHIS(I),ZHIS(I)
77    FORMAT(" R Z",2E10.3)
      WRITE(9)IDUM,XSAP,YSAP,ZSAP
10    CONTINUE
      CALL DATE(TITLE(9))
      CALL CLOCK(TITLE(10))
      R1=1.E100
      RMAX=-R1
      DO 100 MN=1,NWET
      ER= SQRT(RHIS(MN )**2+(DD-ZHIS(MN ))**2)
      IF(ER.LE.RMAX) GO TO 90
      RMAX = ER
      MAX = MN
90    CONTINUE
      IF(ER.GT.R1)GO TO 100
      R1=ER
      MIN=MN
100   CONTINUE
      TOTIME = AMAX1(1000.*(2.*RMAX-R1)/CC,180.0)
      ET=1.E100
      DO 200 MN=1,NWET
      T=RHIS(MN)/(DD-ZHIS(MN))
      TT=ATAN(T)
      YT=ZHIS(MN)/T
      IF(YT.GT.ET)GO TO 200
      ET=YT
      MINT=MN
200   CONTINUE
      CALL EMPIR(MINT)
      HS = ET*FH/4.
      HHS=FH*L*SIN(TT)/10.
      IF(HHS.LT.HS)HS=HHS
      KTIM = 1+(DT*GV)/(HS*1000.*CC)
      ISKIP = MAX0(((KTIM-12)/8)+1,1)
      KTIM =((KTIM-1)/ISKIP+1)*ISKIP
      HS = (DT*GV)/(KTIM*1000.*CC)
      KTIM = (KTIM*TOTIME)/DT
C     WRITE(6,87)MIN
87    FORMAT(" MINIMUM",I5)
      R11=SQRT(RHIS(MIN)**2+(DD-ZHIS(MIN))**2)
      R1=(10+ISKIP)*HS*CC*CC/GV
      CALL CUEINIT(ET)
      RETURN
      END
      SUBROUTINE CUEINIT(ET)
```

```
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EQUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      IJDIM=225000
      RMIN = 1.E9
      RMAX = -RMIN
      DO 1 MN=1,NWET
      RMAX = AMAX1(RMAX,RHIS(MN))
      RMIN = AMIN1(RMIN,RHIS(MN))
1     CONTINUE
      BRITE = 1.2*RMAX
      IARG = SQRT(IJDIM*H/BRITE*2.)
      JDIM = MAX0(IARG,5)/2*2-1
      IDIM = IJDIM/(JDIM+1)
      DR = H/(JDIM - 4)
      C  = CC*0.001
      N  = IFIX(DT*C/(DR*SQRT(2.0)))
      IF(N.EQ.0) GO TO 20
      DR = DT*C/(N*SQRT(2.0))
10    HBAR = (JDIM-4)*DR
      IF(ABS(HBAR-H).LT.DR) GO TO 20
      JDIM = JDIM - 2
      GO TO 10
20    NTIME = IDIM/6*12
      RCAV  = AMIN1(DD,H-DD)-DR
      DELT = SQRT(2.)*DR/C
      ITHIS = (RMIN-RCAV)/C/DELT - 1
      NPRIN = MAX0(ITHIS,0)/6*6
      NTIME = NTIME - NPRIN
      NT = 2 * NTIME
      DELT = DELT * .5
      HS = ET*FH/4.
      HHS= FH*L*SIN(TT)/10.
      IF(HHS.LT.HS) HS=HHS
      ISKIP = 1+(DELT*GV)/(HS*1000.*CC)
      HS   = (DELT*GV)/(ISKIP*1000.*CC)
      KTIM = ISKIP*NT
      T1ST = RCAV/C + NPRIN*DELT*2.
      R1 = R11 - T1ST*C
      RETURN
      END
      SUBROUTINE EMPIR(MN)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EQUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
```

```
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      REAL L,M
      R = SQRT(RHIS(MN )**2+(DD-ZHIS(MN ))**2)
      W3R=W**(1./3.)/R
      PA=14.7
      PS=22347.6*W3R**1.144
      L=0.056*R*W3R**0.753
      IF(L.EQ.0.0) L=10.
      L=L*CC/1000.
      T=RHIS(MN)/(DD-ZHIS(MN))
      TT=ATAN(T)
      PPS=PS
      XLE=L
      WRITE(6,1)PS,L,T,TT
1     FORMAT("  PS,L,TAN(THET),THET",/,4E10.3)
      INOW = 1
      RETURN
      END
      SUBROUTINE CAVITY
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      REAL L,M
       INTEGER Z
      PS=PS*144./(GAM*CC**2)
      PA=PA*144./(GAM*CC**2)
      AA=1./COS(TT)
      AL=AA*L*FH
C         J=1)FORLOWERCURVE,J=2 FOR UPPER ONE
C     FIND POINT B AND INITIAL SLOPE
C
      F=2.*PS/AL
      YB=(AL/2.)*ALOG(F)
C     YB=TO/2.2
      Y(1,1)=YB
      SB=YB/T
      XI(1)=T/5.
C
C       FIND POINT A AND INITXIAL SLOPE
C       CALL NEWTO6 TO SOLVE Y=F(Y)
      CALL NEWTON
      Y(1,2)=YA
      SA=YA/T
      R=(F*EXP(-2.*YA/AL)-1.)/(F-1.)
      XI(2) =T/(1.+4.*R)
C
C       FIND CLOSURE CURVES
C       GENRATE THE COORDINATES
C
```

```
C       REDEFINE TIME STEP SUCH THAT S(I+KK,1),S(I,2) COINCIDE
      NPTS=LOCF(Y(1,2))-LOCF(Y(1,1))
      XK=(SB-SA)/(2.*SA)
      KK=XK+1
      WRITE(6,77)XK,KK
77    FORMAT("  XK=",E10.3," KK=",I5)
      E=(SB-SA)/FLOAT(KK)
      DEL=1.
C       D"S ARE PROPERTIES AT EACH POINT TO BE USED IN FIELD
      D(1,2)=G(2.*YA)
      D(1,1)=PA
      M(1)=0.
      P(1)=2.*YA
      IMAX=NPTS-1
C**************************************************************
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C--------------------------------------------------------------
      ICMIN=1
      DO 5 I=1,IMAX
      N2=I+1
      DO 5 J=1,2
      IFAC=3-2*J
      S=SA+(I+KK*(2-J))*E
      Y(I+1,J)=Y(I,J)-XI(J)*E*IFAC
      P1=Y(I+1,J)+T*S
      YR=-Y(I+1,J)+T*S
      YW=2.*Y(I+1,J)
C         J IS EQ. TO 1 LOWER CURVE
C****************************************************************
      IF(J.EQ.2) GO TO 14
      M(I+1)=-Y(I+1,J)+T*S
      AH=G(P1)
      GO TO 18
14    CONTINUE
C          J IS EQ. TO 2 UPPER CURVE
C**************************************************************
      P(I+1)=Y(I+1,J)+T*S
C
C         CHECK FOR CLOSURE
      IF(I.GT.KK) DEL=Y(I+1-KK,1)-Y(I+1,2)
C         COMPUTE THE VALUE OF THE SLOPE
C          YR TAKES TWO DIFFERENT VALUES WHEN J=2
C         POINT R FALLS ON Y AXIS
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      IF(YR.GT.2.*YA) GO TO 21
      GR=G(YR)
      GO TO 20
21    CONTINUE
C          POINT R FALLS ON CURVE BETWEEN N AND N+1
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      DO 30 IC=ICMIN,I
      IF(YR.LT.P(IC)) GO TO 31
30    CONTINUE
31    N=IC-1
      ICMIN =N
      X=(Y(N+1,2)-Y(N,2))/E
      ZMN=-(Y(N,2)+Y(I+1,2))+T*(I-N+1)*E
      ZMN=ZMN/(E*(T+X))
C       N IS NOT EQUAL TO I
C------------------------------------------------------------
```

```
      IF(N.EQ.I) GO TO 40
      GR=D(N,2)+(D(N+1,2)-D(N,2))*ZMN
40    CONTINUE
      IF(N.NE.I) GO TO 20
C        N IS EQUAL TO I
C-----------------------------------------------------------
      B=Y(N,2)-X*(S-E)
      YRR=(X*YR+T*B)/(X+T)
      GR=G(2.*YRR)
      DO 41 IK=1,IMAX
      AH=GR+2.*PA
      A=IFAC*2.*GG(YW)*YR/(G(Y)-AH)
      D(I+1,J)=(A*AH+2.*G(YW))/(A+2.)
      GRO=GR
      GR=D(N,2)+(D(N+1,2)-D(N,2))*ZMN
      GRN=GR
      DIFF=GRN-GRO
      IF(ABS(DIFF).LT..000000000001)GO TO 101
      IF(IK.EQ.IMAX)WRITE(6,42)
42    FORMAT(/," FAILURE TO ITERATE INIMAX STEPS",/)
41    CONTINUE
101   CONTINUE
      WRITE(6,100)N2,J,IK,DIFF
100   FORMAT(/," N2=",I3,"J=",I3,"IK=",I3,"DIFF=",E20.8)
20    AH=GR+2.*PA
C        COMPUTE THE SLOPE
18    A=IFAC*2.*GG(YW)*YR/(G(YW)-AH)
      XI(J)=T/(1.+A)
C          STORE PROPERTIES
C        WHEN J=2,D=G
C        WHEN J=1,D=F
      D(I+1,J)=(A*AH+2.*G(YW))/(A+2.)
      IF(DEL.LT.0.)GO TO 10
5     CONTINUE
C-----------------------------------------------------------
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C************************************************************
      WRITE(6,19)
19    FORMAT(/," CLOSURE NOT REACHED.RUN TERMINATED.")
      N1=N2-KK
      WRITE(6,75)
75    FORMAT(/,"    I     Y1     Y2")
      WRITE(6,85)(I,Y(I,1),Y(I,2),I=1,N2)
85    FORMAT(I5,2E20.8)
      STOP
10    CONTINUE
C          N2 IS THE NUMBER OF TIME STEPS ON UPPER CURVE(PASSING C)
C
      N1=N2-KK
      WRITE(6,9999)I,N2,KK
 9999 FORMAT(10H I,N2,KK   ,3I10)
C     COMPUTE COORDINATES OF CLOSURE  POINT C
      YC=(+Y(N2,2)*XI(1)+Y(N1,1)*XI(2))/(+XI(1)+XI(2))
      SC=(N2-1)*E-(Y(N2,2)-YC)/XI(2)  +SA
      RETURN
      END
      SUBROUTINE NEWTON
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
```

```
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      R=1.
      F=2.*PS/AL
      YA=PA/(F-1.)
      KAM=100
      DO 2 I=1,KAM
      IF(ABS(R).LT..0000001) GO TO 1
      PE=PA-PS+YA+PS*EXP(-2.*YA /AL)
      PP=1.-F*EXP(-2.*YA/AL)
      R=PE/PP
      YA=YA-R
2     CONTINUE
      WRITE(6,4)I
4     FORMAT(1H," FAILURE TO SOLVE EQU. IN # OF STEPS =",I4,/)
1     CONTINUE
      RETURN
      END
      FUNCTION G(WY)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      EXPZ(Z)=EXP(AMAX1(-500.,Z))
      G=-2.*PS*(EXPZ(-WY/AL))-WY  -PA
      RETURN
      END
      FUNCTION GG(WY)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      EXPZ(Z)=EXP(AMAX1(-500.,Z))
      GG=-1.+2.*PS*(EXPZ(-WY/AL))/AL
      RETURN
      END
      SUBROUTINE ZONE(Z,YX,SX,PRE,VEL,NT)
```

```
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      REAL L,M
      INTEGER Z
      YXR=-YX+T*SX
      YXT=YX+T*SX
      YCT=YC+T*SC
      YCR=-YC+T*SC
      YAT=YA+T*SA
      YAR=-YA+T*SA
      Z=8
      F=PA-YXR
      C=-PA-YXT
      IF(YXT.LT.0.) GO TO 1
      Z=1
      C=G(YXT)
      IF(YXR.LT.0.) GO TO 1
2     Z=2
      F=G(YXR)+2.*PA
      IF(YXT.LE.YAT) GO TO 1
      Z=2
      IF(YXR.GE.YCT) GO TO 1
      Z=3
      IF(YXR.LT.YCR.OR.YXT.LT.YCT) GO TO 44
      CALL SEARCH(-1.,1.,F,YX,SX)
      F=F+2.*PA
      GO TO 1
44    IF(SX.GE.SC.AND.YX.GT.YC) GO TO 444
      IF(SX.GE.SC.AND.YX.LE.YC) GO TO 6
      Z=4
      DO 20 JJ=1,2
      IF(JJ.EQ.0000001) XS(1)=2.*YX
      IF(JJ.EQ.1.AND.SX.LT.SB) GO TO 20
      N=1+(SX-SA+E*KK*(JJ-2))/E
      SN=SA+E*(KK*(2-JJ)+N-1)
      XS(JJ)=(Y(N+1,JJ)-Y(N,JJ))/E
      XS(JJ)=XS(JJ)*(SX-SN)+Y(N,JJ)
20    CONTINUE
      IF(YX.LE.XS(1)) GO TO 5
444   CALL SEARCH(-1.,-1.,F,YX,SX)
      GO TO 1
5     Z=5
      IF(YX.LT.XS(2)) GO TO 6
      C=-2.*GG(2.*YX)*(SX*SIN(TT)/AA-YX/AA**2)
      F=C+G(2.*YX)
      C=-C+G(2.*YX)
      GO TO 1
6     Z=6
      CALL SEARCH(1.,1.,C,YX,SX)
      IF(YXR.LT.YAT) GO TO 7
```

```
      CALL SEARCH(-1.,1.,F,YX,SX)
      INOW=MAX0(NNOW,INOW)
      F=F+2.*PA
      GO TO 1
7     Z=7
      F=G(YXR)+2.*PA
C      CONSTRUCT VELOCITY AND PRESSURE ARRAY
1     CONTINUE
      VEL=((F+C)/(2.*AA))+SX*SIN(TT)-HS*(NT-1)
      PRE=(F-C)/2.
      RETURN
      END
      SUBROUTINE SEARCH(B,C,HFG,YX,SX)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      REAL L,M
      RT=B*YX+T*SX
      J=2
      IF(C.LT.0)J=1
      DO 1 I=INOW,NPTS
      IF(C.GT.0..AND.B*YX+T*SX.LT.P(I)) GO TO 10
C
      IF(C.LT.0..AND.B*YX+T*SX.LT.M(I)) GO TO 10
1     CONTINUE
10    N=I-1
      X=(Y(N+1,J)-Y(N,J))/E
      SN=SA+(N-1+(2-J)*KK)*E
      XX=(B*C*YX-Y(N,J)+T*C*(SX-SN))/((T*C+X)*E)
      HFG=D(N,J)*(1.-XX)+D(N+1,J)*XX
      NNOW=N
      RETURN
      END
      SUBROUTINE ECHO(NT,MN)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      INTEGER Z
      ST(NT)=(NT-1)*HS
      YX=ZHIS(MN)*FH
      R = SQRT(RHIS(MN )**2+(DD-ZHIS(MN ))**2)
      SX=-YX/T+(HS*(NT-1)-(R-R11)*FH)/SIN(TT)
```

```
      SX = SX - R1*FH/SIN(TT)
      IF(NT.GE.2)GO TO 22
      WRITE(6,16)MN,YA,SA,YB,SB,YC,SC
 16   FORMAT(6H NODE=,I5,20H YA,SA,YB,SB,YC,SC    ,/6E10.3)
      YXX=YX/T
      WRITE(6,1)FH,HS,YX,SX,YXX
 1    FORMAT(" FH=",E10.3," HS=",E10.3," YX=",E10.3," SX=",E10.3
     *," YXT=",E10.3)
 22   CONTINUE
      CALL ZONE (Z,YX,SX,PRE,VEL,NT)
      VEL=VEL+(NT-1)*HS
      UEL=(-AMAX1(0.,PRE)+PA+YX)*SIN(TT)
      O(NT,1)=AMAX1(PRE,0.)
      O(NT,2)=UEL
      O(NT,3)=VEL
C     WRITE(6,8)NT,Z,PRE,UEL,VEL
 8    FORMAT(I5,I3,4E10.3)
      ST(NT)=1000.* ST(NT)*CC/GV
      O(NT,1)=(O(NT,1)-PA-YX)*GAM*CC**2/144.
      O(NT,2)=O(NT,2)*CC
      O(NT,3)=O(NT,3)*CC
      RETURN
      END
      SUBROUTINE RESTWR(MN)
      COMMON/CUE/W,DD,H,IBOT,NWET,ZSURF,SSS
      COMMON/DAT/PPS,XLE,PS,PA,L,TT,GV,CC,GAM
      COMMON/CONS/T,E,AA,AL,N1,N2,KK,XS(2)
      COMMON/DANGER/P(25000)
      COMMON/EOUS/M(25000)
      COMMON/CURVE/Y(25000,2),XI(2)
      COMMON/AHEAD/D(25000,2),NPTS,INOW,NNOW,ISKIP
      COMMON/POINTS/YA,SA,YB,SB,YC,SC
      COMMON/GRID/YXO,SXO,FH,HS,KTIM,R11,R1,SQ
      COMMON/CUMBER/YLABEL(3),YMIN(3),YMAX(3),TITLE(10)
      COMMON/KE/RHIS(1600),ZHIS(1600),MIN,RMIN,DT,PEDFAC
      COMMON/MIO/ST(15000),O(15000,3)
      DIMENSION TL(20)
      REAL L,M
      REAL MVC
      DATA TPC,APC,VTC,MVC,FOURPI/4970.,33.,3.21,12414.,12.5663706/
      HH(T)=AMAX1(0.,AMIN1(T/TP,1.-T/TP,.01))*100.
      VDOTR(T)=VTC/TP*(VMAX*(COS(VTC*T/TP))*HH(T))
      TP=TPC*W**(1./3.)/(DD+APC)**(5./6.)
      VMAX=MVC*W/(DD+APC)
      KTEMP = KTIM
      KTIM = KTIM/ISKIP
      IF(MN.GT.1)GO TO 33
      NH=3*NWET
      DECODE(80,98,TITLE) TL
 98   FORMAT(20A4)
      WRITE(9) TL,KTIM,NH
      IFIVE=5
      IEIGHT=8
      TLABEL=8HT (MSEC)
      DO 15 I=1,KTIM
 15   ST(I) = ST(I)*ISKIP
       DT =ST(2) - ST(1)
      WRITE(9)IFIVE,KTIM,TLABEL,ST(1),ST(KTIM),(ST(I),I=1,KTIM)
C     WRITE(6,17)
 17   FORMAT(" 5,KTIM,TLABEL,ST(1),ST(KTIM)")
```

```
C     WRITE(6,7)IFIVE,KTIM,TLABEL,ST(1),ST(KTIM)
7     FORMAT(2I4,A8,2E10.3,/,)
      YLABEL(1)=8HPR (PSI)
      YLABEL(2)=8HHV(FT/S)
      YLABEL(3)=8HVV(FT/S)
33    CONTINUE
      SMALL=1.E-50*CC
      YMIN(1)=-SMALL*PEDFAC
      YMAX(1)=SMALL*PEDFAC
      YMIN(2)=-SMALL
      YMAX(2)=SMALL
      YMIN(3)=-SMALL
      YMAX(3)=SMALL
      BIGR=SQRT(RHIS(MN)**2+(DD-ZHIS(MN))**2)
      BIGR1=SQRT(RHIS(MN)**2+(DD+ZHIS(MN))**2)
      TZERO = (R11-R1)/CC*1000.
      DO 350 I=1,KTIM
      T=ST(I)+TZERO
      TAU=T-(BIGR/CC)*1000.
      TAU1=T-(BIGR1/CC)*1000.
      USUBV=1./FOURPI*((DD+ZHIS(MN))/(BIGR1**3)*VDOTR(TAU1)
     C                +(DD-ZHIS(MN))/(BIGR**3)*VDOTR(TAU))
      USUBH=RHIS(MN)/FOURPI*((1./(BIGR**3))*VDOTR(TAU)
     C                      -(1./(BIGR1**3))*VDOTR(TAU1))
      PPP=0
      UUU=0.
      VVV=0.
      DO 300 J=1,ISKIP
       PPP = PPP + O((I-1)*ISKIP+J,1)/ISKIP
       UUU = UUU + O((I-1)*ISKIP+J,2)/ISKIP
       VVV = VVV + O((I-1)*ISKIP+J,3)/ISKIP
 300  CONTINUE
      O(I,1) = PPP
      O(I,2)= -UUU+USUBH*1000.
      O(I,3)=VVV-USUBV*1000.
      YMAX(1)=AMAX1(YMAX(1),PPP)
      YMIN(1)=AMIN1(YMIN(1),PPP)
      YMAX(2)=AMAX1(YMAX(2),UUU)
      YMIN(2)=AMIN1(YMIN(2),UUU)
      YMIN(3)=AMIN1(YMIN(3),VVV)
      YMAX(3)=AMAX1(YMAX(3),VVV)
350   CONTINUE
      DO 400 I=1,3
      WRITE(9)IEIGHT,KTIM,YLABEL(I),RHIS(MN),ZHIS(MN)
     *,DT,YMIN(I),YMAX(I),(O(K,I),K=1,KTIM)
      WRITE(6,18)
18    FORMAT(" 8 ,KTIM,YLABEL,RHIS,ZHIS,DT,YMIN,YMAX")
      WRITE(6,6)IEIGHT,KTIM,YLABEL(I),RHIS(MN),ZHIS(MN)
     *,DT,YMIN(I),YMAX(I),(O(K,I),K=1,10)
6     FORMAT(2I4,A8,5E10.3,/,10E10.3)
400   CONTINUE
      KTIM = KTEMP
      RETURN
      END
```

SECTION 5

LIST OF REFERENCES

[1] M. L. Rehak , F. L. DiMaggio and I. S. Sandler, "Interactive approximations for a cavitating fluid around a floating structure," *Computers and Structures* V.21 No.21 (1985), pp.1159–1175.

DISTRIBUTION LIST

DEPARTMENT OF DEFENSE

DEF RSCH & ENGRG
ATTN: STRAT & SPACE SYS(OS)

DEFENSE INTELLIGENCE AGENCY
ATTN: DB-6E2 C WIEHLE
ATTN: RTS-2A (TECH LIB)
ATTN: RTS-2B

DEFENSE NUCLEAR AGENCY
ATTN: SPSS
ATTN: STSP
4 CYS ATTN: STTI-CA

DEFENSE TECHNICAL INFORMATION CENTER
12 CYS ATTN: DD

DEPARTMENT OF THE ARMY

U S ARMY ENGR WATERWAYS EXPER STATION
ATTN: TECHNICAL LIBRARY

DEPARTMENT OF THE NAVY

DAVID TAYLOR NAVAL SHIP R & D CTR
ATTN: CODE 11
ATTN: CODE 172
ATTN: CODE 173
ATTN: CODE 174
ATTN: CODE 1740.1
ATTN: CODE 1740.4
ATTN: CODE 1740.5
ATTN: CODE 1740.6
ATTN: CODE 1770.1
ATTN: CODE 1844
ATTN: CODE 2740
ATTN: TECH INFO CTR CODE 522.1

NAVAL POSTGRADUATE SCHOOL
ATTN: CODE 1424 LIBRARY
ATTN: PROF Y S SHIN, CODE 96SG

NAVAL RESEARCH LABORATORY
ATTN: CODE 2627 (TECH LIB)
ATTN: CODE 5100
ATTN: CODE 6380

NAVAL SEA SYSTEMS COMMAND
ATTN: SEA-033
ATTN: SEA-08
ATTN: SEA-323
ATTN: SEA-55X1

NAVAL SURFACE WEAPONS CENTER
ATTN: CODE H21
ATTN: CODE R14
ATTN: CODE R15

NAVAL SURFACE WEAPONS CENTER
ATTN: TECH LIBRARY & INFO SVCS BR

NAVAL UNDERWATER SYS CENTER
ATTN: 8092

NAVAL UNDERWATER SYSTEMS CTR
ATTN: NUSC-NEW LONDON, CT TECH LIBRARY
ATTN: NUSC-NPT, RI TECH LIBRARY

NAVAL WEAPONS CENTER
ATTN: CODE 266 C AUSTIN
ATTN: CODE 3263 J BOWEN
ATTN: CODE 343 (FKA6A2) (TECH SVCS)

NEW LONDON LABORATORY
ATTN: CODE 4492 J KALINOWSKI
ATTN: CODE 4494 J PATEL

DEPARTMENT OF DEFENSE CONTRACTORS

APPLIED RESEARCH ASSOCIATES, INC
ATTN: D PIEPENBURG

BDM CORP
ATTN: A LAVAGNINO
ATTN: A VITELLO
ATTN: CORPORATE LIB

CALIFORNIA RESEARCH & TECHNOLOGY, INC
ATTN: F SAUER

COLUMBIA UNIVERSITY
ATTN: F DIMAGGIO

KAMAN SCIENCES CORP
ATTN: E CONRAD

KAMAN TEMPO
ATTN: DASIAC

KAMAN TEMPO
ATTN: DASIAC

PACIFIC-SIERRA RESEARCH CORP
ATTN: H BRODE, CHAIRMAN SAGE

PACIFICA TECHNOLOGY
ATTN: R BJORK

R & D ASSOCIATES
ATTN: C KNOWLES

WEIDLINGER ASSOC., CONSULTING ENGRG
ATTN: T DEEVY

WEIDLINGER ASSOC, CONSULTING ENGRG
ATTN: J MCCORMICK
ATTN: M BARON
2 CYS ATTN: M REHAK

2 CYS ATTN: R KAGEL
2 CYS ATTN: R SMILOWITZ

END

7-87

DTIC