PREDICTION OF WAVE LOADS AND MOTIONS OF FLOATING MARINE STRUCTURES BY THREE-DIMENSIONAL FLOW THEORY

DEBABRATA SEN

PREDICTION OF WAVE LOADS AND MOTIONS OF FLOATING MARINE STRUCTURES BY THREE-DIMENSIONAL FLOW THEORY

BY

© Debabrata Sen, B.Tech.(Hons)

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of Master of Engineering

Faculty of Engineering and Applied Science
Memorial University of Newfoundland

November, 1983

St. John's Newfoundland

ABSTRACT

The three-dimensional singularity distribution or boundary integral method has been demonstrated by many investigators to be the most versatile and reliable technique for the calculation of harmonic oscillation of a truly three-dimensional floating marine structure in potential flow field.

In the present work, a numerical scheme is presented and a computer program has been developed based on the three-dimensional singularity distribution theory. The program calculates the first order wave exciting forces and moments, hydrodynamic co-efficients and motion responses in six degrees of freedom of any floating marine structure of arbitrary geometry for different angles of heading. Calculations are performed for a floating rectangular box, a vertical circular cylinder and a 130,000 ton dwt tanker. The results are compared with available published results based on the same theoretical model. In general, a good agreement is found between the results.

To demonstrate the versatility and effectiveness of the program, calculations are also performed for a semi-submersible and the results are presented.

ACKNOWLEDGEMENT

The author wishes to express his sincere gratitude to Professor C.C. Hsiung for his enthusiastic supervision and valuable guidance during the entire course of this work. Sincere thanks are also due to Dr. G. R. Peters, Dean of Engineering and Applied Science and Dr. T. R. Chari, Associate Dean of Engineering and Applied Science for their encouragement.

The author is indebted to Dr. F. A. Aldrich, Dean of Graduate Studies for awarding a University Fellowship and also the financial support by NSERC Strategic Group Grant G0561 is acknowledged.

The author also wishes to extend his deep appreciation to Mr. C. C. Tse and Mr. J. M. Chuang, fellow graduate students in Ocean Engineering for their valuable suggestions and helpful discussions.

TABLE OF CONTENTS

LIST OF FIGURES

NOMENCLATURE

x_1, x_2, x_3	=	Co-ordinate system as shown in Figure 1
d	=	Water-depth.
ξ_k	=	Displacements ($k = 1, 2, \ldots 6$ refer to surge, sway, heave, roll, pitch and yaw respectively)
ζ_k	=	Complex motion amplitudes
ω	=	Circular frequency of wave
t	=	Time
i	=	$\sqrt{-1}$
ϕ	=	Complex velocity potential
ψ	=	Complex velocity potential, function of space co-ordinates only
ζ_0	=	Amplitude of incident wave
g	=	Acceleration due to gravity
n_k	=	Generalized direction cosines as defined in equation (2.8)
r_1, θ	=	Polar co-ordinates
$H(\theta)$	=	Unknown complex function
k	=	Wave number
ν	=	$\omega^2/g = k \tanh(kd)$
β	=	Direction of propagation of incident waves with respect to positive x_1 axis
λ	=	Wave length of incoming wave
σ	=	Complex source density function
G	=	Green's function
S	=	Body surface

a_1, a_2, a_3	=	Co-ordinates of a point on the surface of the body
R	=	$[(x_1 - a_1)^2 + (x_2 - a_2)^2 + (x_3 - a_3)^2]^{1/2}$
R_1	=	$[(x_1-a_1)^2 + (x_2-a_2)^2 + (x_3+ 2d +a_3)^2]^{1/2}$
r	=	$[(x_1 - a_1)^2 + (x_2 - a_2)^2]^{1/2}$
J_0	=	Bessel function of the first kind of order zero
Y_0	=	Bessel function of the second kind of order zero
K_0	=	Modified Bessel function of the second kind of order zero
PV	=	Cauchy principal value of the integral
μ	=	Dummy variable of the integral
μ_j	=	Real positive roots of the equation, $\mu_j \tan(\mu_j d) + \nu = 0$
n_1, n_2, n_3	=	Components of outward unit normal to the body surface S in x_1, x_2 and x_3 directions respectively
J_1	=	Bessel function of the first kind of order one
Y_1		Bessel function of the first kind of order one
K_1	=	Modified Bessel function of the second kind of order one
N	=	Total number of elements
ΔS_j	=	Area of j^{th} element
δ_{ij}	=	Kronecker delta function
$\bar{X}, \bar{Y}, \bar{Z}, \bar{\xi}, \bar{\eta}$	=	Local co-ordinate system as defined in Figure 2
$\bar{x}, \bar{y}, \bar{z}$	=	Local co-ordinate of a general point P in space

$\bar{\xi}_i, \bar{\eta}_i$	=	Local co-ordinates of the corner points of a plane quadrilateral element, i=1, 2,...4
b	=	Aspect ratio of a rectangular element
f_k	=	First order wave exciting forces and moments for k^{th} mode, k=1, 2, ...6
ρ	=	Mass density of water
a_{kj}	=	Added mass coefficients (j,k = 1,2,...6)
b_{kj}	=	Damping coefficients (j,k = 1,2,...6)
M_{kj}	=	Inertia matrix
c_{kj}	=	Hydrostatic restoring coefficients
$\|f_k\|$	=	Amplitudes of wave exciting forces and moments for k^{th} mode, k = 1, 2, ...6
m	=	Mass of the body
x_{3G}	=	x_3 co-ordinate of the centre of gravity of the body
I_{jk}	=	Moment of inertia of the body, as defined on page 26
A_{wp}	=	Area of waterplane
V	=	Immersed volume of the body
x_{3B}	=	x_3 co-ordinate of the centre of buoyancy of the body
X_k	=	Oscillatory hydrodyamic forces and moments for k^{th} mode, K = 1, 2, ...6
|A11|, |A22|, |A33|, |A44|, |A55|, |A66|	=	Non-dimensional added mass coefficients for surge, sway, heave, roll, pitch and yaw respectively
|B11|, |B22|, |B33|, |B44|, |B55|, |B66|	=	Non-dimensional damping coefficients for surge, sway, heave, roll, pitch and yaw respectively
|F1|, |F2|, |F3|, |F4|, |F5|, |F6|	=	Non-dimensional wave exciting force and moment amplitudes for surge, sway, heave, roll, pitch and yaw respectively

$\lvert \eta_i \rvert$	=	Non-dimensional amplitudes for i^{th} mode of motion
L	=	Characteristic dimension of the body
ω_n	=	Non-dimensional frequency of oscillation
r_{x_1}	=	Roll radius of gyration
r_{x_2}	=	Pitch radius of gyration
r_{x_3}	=	Yaw radius of gyration

CHAPTER 1

INTRODUCTION

It is essential to have knowledge of motion and hydrodynamic loads of floating marine structures, such as semisubmersible platforms, drilling ships in their early stage of design. Such structures, as a matter of course, require structural analysis in order to ensure safety, reliability and economic feasibility. Structural analyses require a correct prediction of dynamic wave loads, and an estimation of wave loads presupposes a knowledge of motion response in waves.

In potential flow theory, the flow is assumed to be inviscid, irrotational, incompressible and acyclic so that the flow field can be characterized by a single valued velocity potential. A further assumption is that the wave height and responses of the body are small compared to the wave length, water depth and typical body dimensions. Hence the free surface boundary conditions can be linearised with respect to wave height (which implies small amplitude oscillation of the body). This allows the use of Denis-Pierson hypothesis and consequently the usual spectral technique can be used to determine the force and motion responses in an irregular sea from the results obtained for regular waves. Standard frequency domain methods may be used to determine short and long term predictions. Another limitation of the potential flow theory arises due to the assumption of the fluid being ideal, which neglects the effects of viscosity. At high Reynolds number, viscous effects result in flow separation and wake formation for bodies such as slender circular cylinders. For ships, viscous damping effects are known to be important for roll motion. The exact nature of viscous effects are highly complex and depend on various factors such as the size and the shape of the body, amplitude of fluid motion relative to the size of the body, Reynolds number etc. The effects of viscosity will be more pronounced if equations such as Morrison's equations are used. Morrison's equations assume that the body is small relative to the incoming wave length such that the incident flow remains almost unaltered in the vicinity of the body. For large bodies such as ships and semi-submersibles, this assumption is not strictly valid due to diffraction effects. Furthermore, for such large bodies, separation of flow is usually not important [14]. As a result, linearised potential flow theory can be applied in the formulation and solution of the problem to obtain results within acceptable range of accuracy. This approach forms the basis of present day prediction methods for large marine objects.

The three-dimensional singularity distribution method is now believed to be the most versatile technique for calculating harmonic oscillatory motion in a potential flow field for a three-dimensional floating body of arbitrary geometry. Theoretical development for this method was first established by Kim [1], and was later extended and applied to various floating structures by Garrison [2] and Faltinsen [3]. Since then, the effectiveness and reliability of this method have been demonstrated by many investigators [4,5,6]. Conventional methods such as the 'strip' method for ships [7], Hooft's method for semisubmersibles [8] are based on two-dimensional approximations and are not adequate for predicting many of the hydrodynamic characteristics of such floating bodies to the required degree of accuracy. The popularity of these two-dimensional methods is due to the belief that they provide quick results at a much lower computing cost when compared to the three-dimensional singularity distribution method. However, in view of the large and fast computers available today, use of the three-dimensional singularity distribution technique should be made more popular considering its accuracy, reliability and versatility.

In this thesis work, a computer program has been developed based on the singularity distribution method or Green's function method for evaluating wave loads and motion response in six degrees of freedom for floating marine structures of arbitrary shape. Calculations are presented for a rectangular floating box, a vertical circular cylinder, a 130,000 ton dwt. tanker and a semisubmersible platform. Computations have been checked with available published results.

CHAPTER 2

THEORETICAL BACKGROUND

2.1 Formulation of the problem

Consider a rigid body oscillating sinusoidally about a state of rest in response to excitation by a long crested regular waves. An inertial, Cartesian and right-handed system of co-ordinate $Ox_1x_2x_3$ is defined with positive vertically upwards through the centre of gravity of the body and the origin in the plane of the undisturbed free surface. The waterdepth d is finite and constant, and the free surface is assumed to be infinite in all directions (Figure 1).

The problem posed here deals with the fluid motion and the forces induced by the small amplitude oscillation of the object in its six degrees of freedom as well as the fluid motion associated with the interaction of the object with a train of regular waves. The oscillatory motion of the object is described by,

$$\xi_k = \zeta_k e^{-i\omega t}, \quad k = 1, 2, \ldots 6 \tag{2.1}$$

Here, ζ_k is the complex amplitude of motion in the k^{th} mode and ω the circular frequency. The motion variables ξ_1, ξ_2 and ξ_3 denote the three translations along x_1, x_2 and x_3 axes (surge, sway and heave) while ξ_4, ξ_5 and ξ_6 represent angular oscillations about Ox_1, Ox_2 and Ox_3 axes (roll, pitch and yaw) respectively.

The fluid is assumed to be ideal and the flow irrotational, acyclic and harmonic. Therefore, the problem can be formulated in terms of potential flow theory. The flow field can be characterized by a first order complex velocity potential,

$$\phi\,(x_1, x_2, x_3;\ t) = \psi\,(x_1, x_2, x_3)\ e^{-i\omega t} \qquad (2.2)$$

The potential function ψ can be separated into contributions from all modes of motion and from the incident and diffraction wave fields,

$$\psi = i\omega\zeta_0\,(\psi_0 + \psi_7) - i\omega \sum_{k=1}^{6} \psi_k\,\zeta_k \qquad (2.3)$$

Here ψ_k denotes the normalised velocity potential associated with the motion induced by oscillations in the six degrees of freedom, ψ_0 denotes the velocity potential of the incident wave in the absence of the object and ψ_7 denotes the velocity potential of the scattered wave due to the presence of the rigid body. ζ_0 is the incident wave amplitude.

All the individual potentials must satisfy Laplace equation in the fluid domain,

$$\nabla^2\psi_k = 0, \quad k = 0,\ 1,\ 2,\ \ldots..7 \qquad (2.4)$$

It is now necessary to impose the boundary conditions for the geometry specified. These are,

(a) On the sea-floor

The kinematic boundary condition on the sea-floor is,

$$\frac{\partial \psi_k}{\partial x_3} = 0 \quad \text{on} \quad x_3 = -d, \quad k = 0,1,2,\ldots.7 \qquad (2.5)$$

(b) On the Free Surface

On the mean free surface, both kinematic and dynamic conditions are applied. This results in the following well known linearized free surface condition valid for small amplitude oscillations,

$$\frac{\partial \psi_k}{\partial x_3} - \frac{\omega^2}{g} \psi_k = 0 \quad \text{on } x_3 = 0, \; k = 0,1,2\ldots7 \qquad (2.6)$$

Here g = acceleration due to gravity

(c) On the Body Surface

Boundary conditions applied on the average position of the wetted body surface are of the following forms,

$$\frac{\partial \psi_k}{\partial n} = n_k, \quad k = 1,2,\ldots.6 \qquad (2.7a)$$

$$\frac{\partial \psi_7}{\partial n} = - \frac{\partial \psi_0}{\partial n} \qquad (2.7b)$$

Here, $\frac{\partial}{\partial n}$ is the normal derivative in the direction of the outward normal $\vec{n}$ to the body surface. n_1 through n_6 are the generalized direction cosines given by,

$$
\begin{aligned}
n_1 &= \cos\,(n, x_1) \\
n_2 &= \cos\,(n, x_2) \\
n_3 &= \cos\,(n, x_3) \\
n_4 &= x_2 n_3 - x_3 n_2 \\
n_5 &= x_3 n_1 - x_1 n_3 \\
n_6 &= x_1 n_2 - x_2 n_1
\end{aligned}
\tag{2.8}
$$

(d) On Far-field

In order to ensure that the velocity potential has the correct behaviour in the far field, the following radiation condition is imposed,

$$\phi_k(r_1, \theta, x_3) - H(\theta) r_2^{-1/2} \frac{\cosh[k(x_3+d)]}{\cosh(kd)} e^{ikr_2} \to 0 \text{ as } r_2 \to \infty \quad k = 1, 2, \ldots\ldots 7 \tag{2.9}$$

where,

r_1, θ	=	polar co-ordinates
r_2	=	$(x_1^2 + x_3^2)^{1/2}$
θ	=	$\tan^{-1}(x_2/x_1)$
$H(\theta)$	=	unknown complex function
k	=	wave number

2.2 Solution of potentials

Equations (2.4) through (2.9) complete the formulation of the hydrodynamic boundary value problem, to be solved for obtaining the unknown potential functions ψ_k, $k = 0, 1, 2, \ldots 7$.

From the linear wave theory, the incident wave potential ψ_0 is given by,

$$\psi_0 = \frac{1}{\nu} \frac{\cosh[k(x_3+d)]}{\cosh(kd)} e^{ik(x_1 \cos\beta + x_2 \sin\beta)} \tag{2.10}$$

where, β = angle of incidence of the incoming wave (β=0 means waves along positive x_1 direction)

k = wave number = $2\pi/\lambda$

λ = wave length

ν = ω^2/g

The wave number k is related to the wave frequency ω by means of the well known dispersion relation in linear wave theory,

$$\nu = \omega^2/g = k \tanh(kd) \tag{2.11}$$

The potential function ψ_k, $k = 1,2,\ldots 7$ can be represented by a continuous distribution of sources on the wetted body surface S,

$$\psi_k(x_1,x_2,x_3) = \frac{1}{4\pi} \iint_S \sigma_k(a_1,a_2,a_3) G(x_1,x_2,x_3;a_1,a_2,a_3) dS \tag{2.12}$$

where,

a_1,a_2,a_3 = a point on the body surface S

$\sigma_k(a_1,a_2,a_3)$ = unknown complex source strength function

$G(x_1,x_2,x_3;\, a_1,a_2,a_3)$ = the Green's function of a source, singular in (a_1,a_2,a_3)

The above representation has been obtained by Lamb [9] for an infinite fluid case. It is here extended to the case of a fluid of finite depth with free surface [10].

For equation (2.12) to be valid, this particular Green's function which is for a wave source at the body surface must satisfy Laplace equation, boundary conditions on the sea floor, at free surface and at the infinity. Wehausen and Laitone [11] have provided the expression for G appropriate to this particular boundary value problem in the following two forms,

(a) The Integral Form

$$G = \frac{1}{R} + \frac{1}{R_1} + PV \int_0^\infty \frac{2(\mu+\nu)e^{-\mu d}\cosh[\mu(a_3+d)]\cosh[\mu(x_3+d)]}{\mu\sinh(\mu d) - \nu\cosh(\mu d)} J_0(\mu r)d\mu$$

$$+ i\,\frac{2\pi(k^2 - \nu^2)}{k^2 d - \nu^2 d + \nu}\cosh[k(a_3+d)]\cosh[k(x_3+d)]J_0(kr) \qquad (2.13)$$

(b) The Series Form

$$G = \frac{2\pi(\nu^2-k^2)}{k^2 d-\nu^2 d+\nu}\cosh[k(a_3+d)]\cosh[k(x_3+d)][Y_0(kr)-iJ_0(kr)]$$

$$+4\sum_{j=1}^{\infty}\frac{(\mu_j^2 + \nu^2)}{(\mu_j^2 d + \nu^2 d - \nu)}\cos[\mu_j(x_3+d)]\cos[\mu_j(a_3+d)]K_0(\mu_j r) \qquad (2.14)$$

In the above equations,

$$R = [(x_1 - a_1)^2 + (x_2 - a_2)^2 + (x_3 - a_3)^2]^{1/2}$$

$$R_1 = [(x_1 - a_1)^2 + (x_2 - a_2)^2 + (x_3 + 2d + a_3)^2]^{1/2}$$

$$r = [(x_1 - a_1)^2 + (x_2 - a_2)^2]^{1/2}$$

J_0 = Bessel function of the first kind of order zero

Y_0 = Bessel function of the second kind of order zero

K_0 = Modified Bessel function of the second kind of order zero

PV = Cauchy principal value of the integral

The quantities μ_j are the positive solutions of the following equation,

$$\mu_j \tan(\mu_j d) + \nu = 0 \qquad (2.15)$$

The above equation follows from the derivation for G as given in [11] and is not to be linked with the dispersion relation (2.11).

The unknown source strength functions σ_k in equation (2.12) are to be determined such that the kinematic boundary conditions on the mean wetted surface (equations 2.7a, 2.7b) are fulfilled. This results in the following two dimensional Fredholm integral equation of the second kind,

$$-\frac{1}{2}\sigma_k(x_1,x_2,x_3) + \frac{1}{4\pi}\iint_S \sigma_k(a_1,a_2,a_3)\frac{\partial G}{\partial n}(x_1,x_2,x_3;\ a_1,a_2,a_3)dS$$
$$= n_k \quad \text{for } k = 1, 2, \ldots, 6 \qquad (2.16)$$
$$= -\frac{\partial \psi_0}{\partial n} \quad \text{for } k = 7$$

where $\frac{\partial G}{\partial n}$ represents the derivative of the Green's function in the direction of the outward normal vector n and can be expressed as,

$$\frac{\partial G}{\partial n} = \frac{\partial G}{\partial x_1} \cdot n_1 + \frac{\partial G}{\partial x_2} \cdot n_2 + \frac{\partial G}{\partial x_3} \cdot n_3 \tag{2.17}$$

where n_1, n_2, n_3 are the three components of the unit normal vector as defined in equation (2.8). $\frac{\partial G}{\partial x_1}$, $\frac{\partial G}{\partial x_2}$, $\frac{\partial G}{\partial x_3}$ can be obtained from straightforward differentation of G given in equations (2.13) and (2.14). The expressions for these derivatives are given below.

(a) Derivatives of G, series form

$$\frac{\partial G}{\partial x_1} = A \cosh[k(x_3+d)]\{-k\frac{(x_1-a_1)}{r} Y_1(kr) + ik\frac{(x_1-a_1)}{r} J_1(kr)\}$$
$$- \sum_{j=1}^{\infty} B(\mu_j) \cos[\mu_j(x_3+d)] \mu_j \frac{(x_1-a_1)}{r} K_1(\mu_j r) \tag{2.18a}$$

$$\frac{\partial G}{\partial x_2} = A \cosh[k(x_3+d)]\{-k\frac{(x_2-a_2)}{r} Y_1(kr) + ik\frac{(x_2-a_2)}{r} J_1(kr)\}$$
$$- \sum_{j=1}^{\infty} B(\mu_j) \cos[\mu_j(x_3+d)] \mu_j \frac{(x_2-a_2)}{r} K_1(\mu_j r) \tag{2.18b}$$

$$\frac{\partial G}{\partial x_3} = A\, k \sinh[k(x_3+d)]\{Y_0(kr) - i\, J_0(kr)\}$$
$$- \sum_{j=1}^{\infty} B(\mu_j)\, \mu_j \sin[\mu_j(x_3+d)]\, K_0(\mu_j r) \tag{2.18c}$$

where,

$$A = \frac{2\pi(\nu^2 - k^2)}{k^2 d - \nu^2 d + \nu} \cosh[k(a_3+d)]$$

$$B(\mu_j) = \frac{4(\mu_j^2 + \nu^2)}{\mu_j^2 d + \nu^2 d - \nu} \cos[\mu_j(a_3+d)]$$

J_1 = Bessel function of the first kind of order one

Y_1 = Bessel function of the second kind of order one

K_1 = Modified Bessel function of the second kind of order one

(b) Derivatives of G, Integral Form

$$\frac{\partial G}{\partial x_1} = -\frac{(x_1-a_1)}{R^3} - \frac{(x_1-a_1)}{R_1^3} - PV\int_0^\infty g_1(\mu)\cosh[\mu(x_3+d)]\mu(x_1-a_1)\cdot \frac{J_1(\mu r)}{r}d\mu - i\, D \cosh[k(x_3+d)]\ (x_1-a_1)\ k\frac{J_1(kr)}{r} \tag{2.19a}$$

$$\frac{\partial G}{\partial x_2} = -\frac{(x_2-a_2)}{R^3} - \frac{(x_2-a_2)}{R_1^3} - PV\int_0^\infty g_1(\mu)\cosh[\mu(x_3+d)]\mu(x_2-a_2)\cdot \frac{J_1(\mu r)}{r}d\mu - i\, D \cosh[k(x_3+d)]\ (x_2-a_2)\ k\frac{J_1(kr)}{r} \tag{2.19b}$$

$$\frac{\partial G}{\partial x_3} = -\frac{(x_3-a_3)}{R^3} - \frac{(x_3+2d+a_3)}{R_1^3} + PV\int_0^\infty g_1(\mu)\ \mu\ \sinh[\mu(x_3+d)]\cdot J_0(\mu r)d\mu + i\ D\ k\ \sinh[k(x_3+d)]J_0(kr) \tag{2.19c}$$

In the above,

$$g_1(\mu) = \frac{2(\mu+\nu)\ e^{-\mu d}\cosh[\mu(a_3+d)]}{\mu\sinh(\mu d) - \nu\cosh(\mu d)}$$

$$D = \frac{2\pi(k^2 - \nu^2)}{k^2 d - \nu^2 d + \nu} \cosh[k(a_3+d)]$$

When $r = 0$, $J_1(\mu r)/r$ term in (2.19a) and (2.19b) are replaced by 0.5μ.

CHAPTER 3

NUMERICAL FORMULATION

3.1 Numerical solution of potential

In order to obtain the unknown source strength functions σ_k, it is now necessary to solve equation (2.16). The solution is obtained numerically, using a discretized solution scheme. The wetted body surface S is approximated by a sufficiently large number of plane quadrilateral surface panels or elements of area ΔS_j, $j = 1, N$ (N = total number of surface elements). Theoretically, the continuous formulation of equation (2.16) indicates that this equation is to be satisfied at all points on the wetted body surface. However, to obtain a practical numerical solution, this requirement is relaxed and the equation is satisfied only at N points which are termed as control points. The control points, in principle, can be chosen arbitrarily. Here the centroid of the elements are chosen as control points for the reason of convenience.

In the following numerical formulation, suffix k for the 6 modes of motion and diffraction potential has been omitted. It is to be understood that these equations apply to all k, k = 1, 2, ...7.

Due to discretization, equation (2.16) now gets transformed to a set of N linear equations,

$$\sum_{j=1}^{N} \alpha_{ij}\, \sigma_i = b_i, \qquad i = 1,\ 2,\ \ldots.N. \tag{3.1}$$

in which the coefficients α_{ij} and b_i are respectively given by,

$$\alpha_{ij} = -\delta_{ij} + \frac{1}{2\pi} \iint_{\Delta S_j} \frac{\partial G}{\partial n} (x_{1i}, x_{2i}, x_{3i};\ a_1, a_2, a_3) dS \tag{3.2}$$

and,

$$\begin{aligned} b_i &= 2n_i \qquad \text{for} \qquad k = 1,\ 2,\ \ldots\ldots 6 \\ &= -2\frac{\partial \psi_0}{\partial n} (x_{1i}, x_{2i}, x_{3i}) \ \text{for}\ k = 7 \end{aligned} \tag{3.3}$$

In the above, n_i for $k = 1,2,\ldots 6$ are the generalized direction cosines as defined in equation (2.8) for the control point i. $\frac{\partial \psi_0}{\partial n}$ can be obtained by straightforward differentiation of ψ_0 given in equation (2.10),

$$\begin{aligned} \frac{\partial \psi_0}{\partial n} = & \, i\{\frac{k}{\nu} \frac{\cosh[k(x_3+d)]}{\cosh(kd)} e^{ik(x_1\cos\beta\, +\, x_2\sin\beta)} [n_1\cos\beta + n_2\sin\beta]\} \\ & + n_3 \frac{k}{\nu} \frac{\sinh[k(x_3+d)]}{\cosh(kd)}\, e^{ik(x_1\cos\beta\, +\, x_2\sin\beta)} \end{aligned} \tag{3.4}$$

In equation (3.2), δ_{ij} is the Kronecker delta function, $\delta_{ij} = 0$ for $i \neq j$, $\delta_{ii} = 1$ and (x_{1i}, x_{2i}, x_{3i}) is the centroid or the control point of the i^{th} element. In physical terms, α_{ij} represents the velocity induced at the i^{th} control point in the direction normal to the surface by a source distribution of unit strength distributed uniformly over the j^{th} element. When $i = j$, $\delta_{ij} = 1$ and this term takes care of the velocity at the control point due to a uniform source distribution of that element, and the last term in equation (3.2) should be neglected.

To carry out the integration in the second term of equation (3.2) numerically, further assumption is necessary. This integrand oscillates approximately with the wave length λ which in practice is generally large, at least comparable to the characteristic dimension of the immersed surface. $\frac{\partial G}{\partial n}$ for $i \neq j$ thus vary slowly over ΔS_j and can be assumed to be constant over an element with the value equal to the value at the centroid. This yields the following approximation of α_{ij},

$$\alpha_{ij} = -\delta_{ij} + \frac{\Delta S_j}{2\pi} \frac{\partial G}{\partial n} (x_{1i}, x_{2i}, x_{3i}; a_{1j}, a_{2j}, a_{3j}) \qquad (3.5)$$

where (a_{1j}, a_{2j}, a_{3j}) is the j^{th} control point.

Thus, it is now possible to evaluate the matrix $[\alpha_{ij}]$ and the column vector $\{b_i\}$. The unknown source distribution function σ_j is now easily determined using a complex matrix inversion procedure.

By a similar method of discretization, equation

(2.12) can be written as,

$$\psi(x_{1i},x_{2i},x_{3i}) = \sum_{j=1}^{N} \beta_{ij}\ \sigma_j \tag{3.6}$$

where,

$$\beta_{ij} = \frac{1}{4\pi} \iint_{\Delta S_j} G(x_{1i},x_{2i},x_{3i};\ a_1,a_2,a_3)dS \tag{3.7}$$

To evaluate the above integration numerically, a similar assumption is made regarding the value of G over an element as was made for $\partial G/\partial n$, for the same reason. Thus, assuming G constant over the element with it's value same as at the centroid, the following approximation of β_{ij} is obtained,

$$\beta_{ij} = \frac{\Delta S_j}{4\pi} G(x_{1i},x_{2i},x_{3i};\ a_{1j},a_{2j},a_{3j}) \tag{3.8}$$

When i = j, this particular case must now be carefully considered, since in this case a singularity of the form $\frac{1}{R}$, R = 0 occurs in G. Clearly, the above approximation of β_{ij} can not be used for evaluating β_{ii}. The singular term in G is more dominant than the regular term in G for i = j, and hence this singular term alone is considered for the case i=j. Thus,

$$\beta_{ii} = \frac{1}{4\pi} \iint_{\Delta S_i} \frac{1}{R}\ dS \tag{3.9}$$

For evaluating the above integral, the formulation given by Faltinsen and Michelsen [3] is used. For a plane quadrilateral element, firstly the integral is written in terms of the local co-ordinates $\bar{X},\bar{Y},\bar{Z}$ and $\bar{\xi},\bar{\eta}$, where $\bar{X},\bar{Y}$ and $\bar{\xi},\bar{\eta}$ axes are in the plane of the quadrilateral element (Figure 2). This integral for a general point P in space having local co-ordinates $(\bar{x},\bar{y},\bar{z})$ is,

$$\iint_{\Delta S} \frac{1}{R}\, dS = \iint_{\Delta S} \frac{d\bar{\xi}\, d\bar{\eta}}{[(\bar{x}-\bar{\xi})^2 + (\bar{y}-\bar{\eta})^2 + \bar{z}^2]^{1/2}} \qquad (3.10)$$

This integration can be performed analytically yielding the following,

$$\begin{aligned}\iint_{\Delta S} \frac{1}{R}\, ds = &- \int_{\bar{\xi}_1}^{\bar{\xi}_2} d\bar{\xi}\, \ln\{\bar{y} - \bar{\eta}_{12} + [(\bar{y} - \bar{\eta}_{12})^2 + (\bar{x}-\bar{\xi})^2 + \bar{z}^2]^{1/2}\} \\ &- \int_{\bar{\xi}_2}^{\bar{\xi}_3} d\bar{\xi}\, \ln\{\bar{y} - \bar{\eta}_{23} + [(\bar{y}-\bar{\eta}_{23})^2 + (\bar{x}-\bar{\xi})^2 + \bar{z}^2]^{1/2}\} \\ &- \int_{\bar{\xi}_3}^{\bar{\xi}_4} d\bar{\xi}\, \ln\{\bar{y} - \bar{\eta}_{34} + [(\bar{y}-\bar{\eta}_{34})^2 + (\bar{x}-\bar{\xi})^2 + \bar{z}^2]^{1/2}\} \\ &- \int_{\bar{\xi}_4}^{\bar{\xi}_1} d\bar{\xi}\, \ln\{\bar{y} - \bar{\eta}_{41} + [(\bar{y}-\bar{\eta}_{41})^2 + (\bar{x}-\bar{\xi})^2 + \bar{z}^2]^{1/2}\}\end{aligned} \qquad (3.11)$$

where,

$$\bar{\eta}_{ij} = \bar{\eta}_i + \frac{\bar{\eta}_j - \bar{\eta}_i}{\bar{\xi}_j - \bar{\xi}_i}\,(\bar{\xi} - \bar{\xi}_i)$$

$\bar{\eta}_i$, $\bar{\xi}_i$ $\doteq$ $(\bar{\eta}, \bar{\xi})$ co-ordinates of the corner points of the element

All the integrals in equation (3.11) are performed numerically. Since this is evaluated only for β_{ii} in equation (3.9), point P in this case is at the centroid of the panel, thus $\bar{z} = 0$. A singularity in the integrands occur when $\bar{\xi} = \bar{x}$ and $\bar{y} - \bar{\eta}_{ij} < 0$. In such a case, integration about the immediate neighbourhood of the singular point $(\bar{x} - \varepsilon)$ and $(\bar{x} + \varepsilon)$ is avoided. For computer evaluation, ε has been successively reduced until the integral converges to a given limit.

For a rectangular element of aspect ratio b, an analytical expression has been derived by Garrison [10] when P is at the centroid of the element. This expression given below is used when the element is rectangular.

$$\iint_{\Delta S} \frac{1}{R}\, dS = 2\left(\frac{\Delta S}{b}\right)^{1/2}\{\ln[b+(b^2+1)^{1/2}] + b\ln[1+\frac{(b^2+1)^{1/2}}{b}]\} \tag{3.12}$$

After evaluating $[\beta_{ij}]$, the potential function $\psi(x_{1i}, x_{2i}, x_{3i})$ is easily determined from equation (3.6).

3.2 Numerical evaluation of Green's function

Although the two forms of the Green's function given in equation (2.13) and (2.14) are equivalent, one of the two forms may have preference for numerical computation, depending on the value of the variables. In general, the series form converges rapidly due to $K_0(\mu_j r)$ term. However, when $kr \to 0$, the Bessel function $K_0(\mu_j r) \to \infty$ and so the series form cannot be used for very small values of kr. Here the series form is used for $kr > 0.01$ and the more time consuming integral form is used for $kr < 0.01$.

Equation (2.15) has been solved using Newton-Raphson iteration method which converges fast. The evaluation of Green's function and it's derivatives through the series form is rather straightforward and no major numerical difficulties are encountered. A convergence criterior is used to terminate the series when required convergence is reached.

The integral form is evaluated after breaking down the infinite upper limit of the integral into two parts, 0 to 2k and 2k to ∞. The integral over the interval 0 to 2k can be further broken down and written in the following form,

$$\int_0^{2k} \frac{F(\mu)d\mu}{\mu\tanh(\mu d)-\nu} = \int_0^{2k} \frac{F(\mu)-F(k)}{\mu\tanh(\mu d)-\nu}d\mu + F(k)\int_0^{2k} \frac{1}{\mu\tanh(\mu d)-\nu}d\mu \quad (3.13a)$$

The first integral in the right hand side of the above equation is now finite at all points within the interval and can be numerically integrated. The second integral can be divided into the following intervals,

$$\int_0^{2k} \frac{d\mu}{\mu\tanh(\mu d)-\nu} = \int_0^{k-\varepsilon} \frac{d\mu}{\mu\tanh(\mu d)-\nu} + \int_{k-\varepsilon}^{k+\varepsilon} \frac{d\mu}{\mu\tanh(\mu d)-\nu} + \int_{k+\varepsilon}^{2k} \frac{d\mu}{\mu\tanh(\mu d)-\nu} \tag{3.13b}$$

All the integrals are evaluated using numerical integration procedure except for the integral within the limit $(k-\varepsilon)$ to $(k+\varepsilon)$, which contains a singularity of the form $1/(\mu-k)$. The integrand is expanded in the power of $(\mu-k)$ and only terms upto first order are considered,

$$\frac{1}{\mu\tanh(\mu d) - \nu} = \frac{C_{-1}}{(\mu-k)} + C_0 + C_1(\mu-k) + \ldots\ldots\ldots$$

Each term is now integrated giving the following result,

$$\int_{k-\varepsilon}^{k+\varepsilon} \frac{1}{\mu\tanh(\mu d)-\nu}\, d\mu = -\frac{\mathrm{sech}^2(kd)[1-kd\tanh(kd)]}{[\tanh(kd) + kd\,\mathrm{sech}^2(kd)]^2}(2\varepsilon)+0(\varepsilon^3) \tag{3.14}$$

For the purpose of computation, a value of $\varepsilon = 0.1k$ is chosen as suggested by Garrison [10].

To evaluate the integral within the interval 2k to ∞, trapezoidal rule is used and the integration is terminated when the contribution to the integral becomes sufficiently small. A convergence criterior is used for this purpose. When μ is large, the integrand decays as $\exp[\mu(x_3 + a_3)]$. To take advantage of this situation, a progressively larger stepsize is used for higher values of μ, thus saving valuable CPU time. A stepsize of 0.1μ or $0.3/r$, whichever is less is chosen. This is sufficient to represent the denominator $[\mu \sinh(\mu d) - \nu \cosh(\mu d)]$ and $J_0(\mu r)$ accurately [12].

3.3 Wave forces, moments and motion response

Once all the potentials ψ_k, k = 1, 2, ...7 are determined, the first order wave exciting forces and moments can be readily determined through a use of linearized Bernoulli's equation. They can be written as,

$$f_k = - \rho\omega^2 \zeta_0 e^{-i\omega t} \iint_S (\psi_0 + \psi_7) n_k dS, \quad k= 1,2,....6 \quad (3.15)$$

where, f_k = first order wave exciting forces/moments for k^{th} mode

ρ = mass density of water

The exciting forces and moments can also be expressed in terms of the incident and radiation potentials and their normal derivatives by means of the Haskind relation. In the method of computation presented in this thesis, the matrices $[\alpha_{ij}]$ and $[\beta_{ij}]$ containing $\partial G/\partial n$ and G terms respectively are to be calculated, and the inversion of $([\alpha_{ij}]$ is to be carried out in order to compute the radiation potentials. These are the most complex and time consuming parts of the calculation. Computation of the diffraction potential involves only a simple matrix multiplication. Thus it was felt more convenient to use the above expression for computing exciting forces and moments instead of using Haskind relation which requires calculation of the normal derivatives of the radiation potentials.

The oscillatory hydrodynamic forces ($k = 1,2,3,$) and moments ($k = 4,5,6,$) in the k^{th} mode are given by,

$$X_k = -\rho\omega^2 \sum_{j=1}^{6} \zeta_j \, e^{-i\omega t} \iint_S \psi_j \, n_k \, dS \tag{3.15a}$$

The added mass and damping coefficients are expressed in their usual forms,

$$\begin{aligned} a_{jk} &= -\rho \, \mathrm{Re} \iint_S \psi_k \, n_j \, dS \\ b_{jk} &= -\rho\omega \, \mathrm{Im} \iint_S \psi_k \, n_j \, dS \end{aligned} \tag{3.16}$$

where,

a_{jk} = added mass coefficient in j^{th} mode due to motion in k^{th} mode

b_{jk} = damping coefficient in j^{th} mode due to motion in k^{th} mode

Re = real part of the integral

Im = imaginary part of the integral

By applying Green's theorem to the expression for added mass and damping coefficients given above, it can be easily seen that the coefficients are symmetric; that is: $a_{kj} = a_{jk}$; $b_{kj} = b_{jk}$

The well known equations of motion are now used to determine the motion response to the first order excitation in frequency domain,

$$\sum_{j=1}^{6} [(M_{kj} + a_{kj})\ddot{\xi}_j + b_{kj}\dot{\xi}_j + c_{kj}\xi_j] = f_k, \quad k = 1,2,\ldots 6 \quad (3.17)$$

Substituting $\xi_j = \zeta_j e^{-i\omega t}$ and $f_k = |f_k| e^{-i\omega t}$, we get the following set of linear equations,

$$\sum_{j=1}^{6} [-\omega^2(M_{kj} + a_{kj}) - i\omega b_{kj} + c_{kj}] \zeta_j = |f_k| \quad (3.18)$$

where,

$|f_k|$ = amplitude of wave exciting forces/moments in the k^{th} mode

M_{kj} = inertia matrix

c_{kj} = hydrostatic restoring coefficient matrix

M_{kj} is given by,

$$M_{kj} = \begin{bmatrix} m & 0 & 0 & 0 & mx_{3G} & 0 \\ 0 & m & 0 & -mx_{3G} & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & -mx_{3G} & 0 & I_{44} & -I_{45} & -I_{46} \\ mx_{3G} & 0 & 0 & -I_{54} & I_{55} & -I_{56} \\ 0 & 0 & 0 & -I_{64} & -I_{65} & I_{66} \end{bmatrix} \quad (3.19)$$

where,

m = mass of the body

I_{jk} = moment of inertia with respect to the co-ordinate system $Ox_1 x_2 x_3$ shown in Figure 1

x_{3G} = x_3 co-ordinate of centre of gravity

The moment of inertia terms are defined as,

$$I_{jk} = I_{kj} = \int_m x_{j-3} x_{k-3} \, dm \, , \quad j = 4,5,6; \; k = 4,5,6 \qquad (3.20)$$

For a body symmetric about $x_1 x_3$ plane, $I_{45} = I_{54} = I_{56} = I_{65} = 0$.

The non-zero terms of the hydrostatic restoring matrix $[c_{jk}]$ for a general shape are,

$$c_{33} = \rho g \, A_{wp}$$

$$c_{43} = c_{34} = \rho g \iint_{A_{wp}} x_2 \, dS$$

$$c_{44} = \rho g V \, (x_{3B} - x_{3G}) + \rho g \iint_{A_{wp}} x_2^2 \, dS$$

$$c_{35} = c_{53} = -\rho g \iint_{A_{wp}} x_1 \, dS$$

$$c_{45} = c_{54} = -\rho g \iint_{A_{wp}} x_1 x_2 \, dS$$

$$c_{55} = \rho g V\,(x_{3B} - x_{3G}) + \rho g \iint_{A_{wp}} x_1^{\,2}\, dS \qquad (3.21)$$

In the above,

A_{wp} = area of the waterplane

V = immersed volume of the body

x_{3B} = x_3 co-ordinate of centre of buoyancy

If $x_1 x_3$ is a plane of symmetry for the body,

$c_{34} = c_{43} = c_{45} = c_{54} = 0.$

From equation (3.18), the complex motion amplitudes ζ_j for all six modes of motion are now easily determined using a complex matrix inversion procedure.

This completes the numerical formulation of the problem. The integrations in equations (3.15), (3.15a) and (3.16) are performed numerically, assuming the integrand to be constant over each element.

CHAPTER 4

COMPUTED RESULTS

A computer program has been written based on the theoretical and numerical formulation given above. The input information required are the geometry of the body, mass and various radii of gyrations (roll, pitch, yaw, roll-pitch, roll-yaw and yaw-pitch), vertical co-ordinate of the centre of gravity (x_{3G}), water depth (d), heading angles (β) and incoming wave lengths (λ). The program does not calculate the hydrostatic restoring coefficients which depend entirely on the geometry of the body and are rather straightforward to calculate. These are to be given as input data. Subdivision of the immersed body surface into plane quadrilateral elements is to be done by the user and the co-ordinates of the element vertices are to be given as input data. The program is in two parts. The first part calculates the element centroid, the components of the outward normal, the area of the element, and the integral as given by equation (3.11). The output of the first part of the program is the major input for the second which is the major aspect of the program. The final results obtained are the wave exciting forces, the moments and motion response in six degrees of freedom of a floating marine structure of an arbitrary shape. Listing of both the programs are given in Appendix A. The input data is in a free floating format form. Appendix B shows a typical input data for the second part of the program for the semisubmersible. The values are in a non-dimensional form and the non-dimensionalizing factors used are as follows.

a) Surge, sway and heave added mass co-efficients,

$(|A11|, |A22|, |A33|) = (a_{11}, a_{22}, a_{33})/\rho V$

b) Roll, pitch and yaw added mass co-efficients,

$(|A44|, |A55|, |A66|) = (a_{44}, a_{55}, a_{66})/\rho VL^2$

c) Surge, sway and heave damping co-efficients,

$(|B11|, |B22|, |B33|) = (b_{11}, b_{22}, b_{33})/\rho V\sqrt{(g/L)}$

d) Roll, pitch and yaw damping co-efficients,

$(|B44|, |B55|, |B66|) = (b_{44}, b_{55}, b_{66})/\rho VL^2\sqrt{(g/L)}$

e) Surge, sway and heave exciting force amplitudes,

$(|F1|, |F2|, |F3|) = (|f_1|, |f_2|, |f_3|)/\rho gV\zeta_0/L$

f) Roll, pitch and yaw exciting moment amplitudes,

$(|F4|, |F5|, |F6|) = (|f_4|, |f_5|, |f_6|)/\rho gV\zeta_0$

g) Surge, sway and heave motion amplitudes,

$(|\eta_1|, |\eta_2|, |\eta_3|) = (\zeta_1, \zeta_2, \zeta_3)/\zeta_0$

h) Roll, pitch and yaw motion amplitudes,

$(|\eta_4|, |\eta_5|, |\eta_6|) = (\zeta_4, \zeta_5, \zeta_6)/\zeta_0/L$

i) Non-dimensional frequency,

$$\omega_n = \omega\sqrt{(L/g)}$$

L in the above is the characteristic dimension of the body.

Computations are performed for various floating objects. Here the following results are presented.

A. Rectangular Box

Computations for a floating box of length 90 m, breadth 90 m and draft 20 m floating in water of depth 200 m are performed. The geometrical properties of the box are,

Centre of gravity co-ordinates (CG)	=	0, 0, 8.82 m
Roll radius of gyration, r_{x1}	=	37.32 m
Pitch radius of gyration, r_{x2}	=	37.30 m
Yaw radius of gyration, r_{x3}	=	40.08 m

Two sets of calculations are performed using a total of 48 and 108 elements to represent the box. L = 90 m is used for non-dimensionalization. The non-dimensional added mass and damping co-efficients, exciting force and moment amplitudes and phase angles, motion amplitudes and phase angles for heading angle $\beta = 0$ deg. are presented in Figures 3 through 14. This particular example is chosen to present the comparison of the results with those available in [3].

B. Vertical Circular Cylinder

Calculations are performed for a short vertical circular cylinder of radius a = 10 m and draft T = 0.5a = 5 m. The following geometrical properties are used for the purpose of computation, CG = 0, 0, 0 m (at the origin of the co-ordinate system)

$r_{x_1} = 0.5a = 5$ m

$r_{x_2} = 0.5a = 5$ m

$r_{x_3} = 0.707a = 7.07$ m

A total of 60 surface elements are used to idealize the body. Calculations are made for three different water depths, d = 10, 15 and 50 m. For non-dimensionalization, the diameter of the cylinder is taken as characteristic dimension of the body, which means $L = 2a = 20$ m is used. The results of computation are presented in Figures 15 through 22. This example is chosen since some of the results computed by Garrison based on the same theory are available in [10]. Figure 23 shows Garrison's computations for surge mode.

C. Tanker

Wave exciting forces, moments and motion response of a 130,000 tons dwt tanker moored in water of depth 500 ft (152.4 m) are computed. The geometry of the tanker is shown in Figure 24. Two different conditions of loading are considered, ballast and fully loaded. The geometrical properties of the tanker are given in Table 1.

A total of 196 elements for ballast condition and 208 elements for loaded condition are used. Computations are performed for three different heading angles, $\beta = 0$, 45 and 90 deg. Length between perpendiculars is used as characteristic dimension of the tanker. Calculations are also performed using two-dimensional strip theory for comparison. The results are presented in Figures 25 through 45.

Results of motion response for this tanker for both loaded and ballast conditions using DnV program are available in [13] and are shown in Figures 46 through 49 for the purpose of comparison.

D. Semisubmersible

Finally, to demonstrate the effectiveness and usefulness of the program, computations are performed for a semisubmersible. Figure 50 shows the sectional views [15]. The geometrical data of the semisubmersible are as follows,

Displacement	=	20869 tonnes
Length	=	90 m
Beam	=	75 m
Draft	=	18.5 m
Metacentric height, transverse	=	2.62 m
Metacentric height, longitudinal	=	2.67 m
r_{x_1}	=	30.22 m
r_{x_2}	=	28.88 m
r_{x_3}	=	36.92 m

A total of 244 elements are used to represent the semisubmersible. $L = 90$ m is used for non-dimensionalization. The computed results are presented in Figures 51 through 68.

CHAPTER 5

DISCUSSIONS AND CONCLUDING REMARKS

To check the present computations, the results are compared with other available results based on the same three dimensional singularity distribution theory. In general, an excellent agreement is found between the results. The results for the rectangular box calculated by Faltinsen and Michelsen [3] using 68 surface elements are plotted in Figures 3 through 14. It can be easily seen that the results are in good agreement. The only significant differences are observed in heave exciting force amplitude at higher periods, and pitch exciting moment at lower periods.

Results for the vertical circular cylinder are also compared with the results calculated by Garrison [10] and again a good agreement is found. In Figure 23, present results are plotted against Garrison's results for surge mode.

To determine the effect of the number of surface elements, the rectangular box calculations are performed using both 48 and 108 elements. The observations are same as in [3]. For most of the cases, 48 panels are sufficient to ~~obtain reasonably accurate results. However, for some~~

rotational mode calculations (k = 4, 5, 6), there are some differences between the results using 48 and 108 elements. This is to be expected, since the rotational mode calculations are more sensitive to the correct representation of the geometry. They depend on $\vec{r} \times \vec{n}$ terms whereas the linear mode calculations depend on $\vec{n}$ terms ($\vec{r}$ is the position vector of any point on the body surface). A difference in heave damping coefficient is also noted.

Comparing the effect of the number of elements on the computed results of the heave and pitch motion responses (figures 13 and 14 respectively), it can be seen that the differences are more significant for the pitch motion and they extend over the entire frequency range. For the heave motion, the differences are significant over the resonant frequency region but are not so pronounced over the other range. The differences in results at peak period (resonant frequency) are about 1.5 times for heave motion while they differ by more than 3 times for pitch motion. It is to be noted that calculations in the region of the resonance frequency are sensitive to the number of elements and would require a careful evaluation.

To compare the results using three-dimensional singularity distribution method and two-dimensional strip theory, the tanker added mass and damping coefficients calculated by strip theory are plotted in Figures 25 to 40. The agreement between the results is not generally very good.

This is again in agreement to the observation made in [3]. An interesting observation is that the agreement between them improves towards lower time period or higher frequency range. This is expected, since strip theory is known to give better and more reliable results at higher frequencies.

The motion response of the tanker is compared with the results computed using DnV program based on the same singularity distribution theory (Figures 46 through 49). The results are in reasonably good agreeement and have comparable range of values. There does not appear to be a correlation for the pitch motion between the results computed here and the result of DnV. Similarly for the sway at beam sea conditions, there is some disagreement between the two results. The tanker geometry was obtained from the small scaled body plan given in [13] which was enlarged for the purpose of dividing the hull into surface elements. This could be a major input deficiency in comparing the results and is an aspect to be examined further.

The computed results of the semisubmersible could not be directly compared since there are no available data of the exactly same configuration, whether based on the same theory or any other theories or experimental results. However, it is possible to compare the nature and trend of the computed hydrodynamic coefficients, wave exciting forces and moments and motion responses with similar kind of structures. For example, in [8] some results of a Staflo drilling platform based on a different theory by Hooft are available. They show a similar trend, and the range of values of the various non-dimensional results are quite comparable.

The main disadvantage of the present three-dimensional singularity distribution method is the enormous volume of computation that is required. It is possible to achieve a reduction in the computation time if the object has one or more planes of symmetry. At present no such assumption about the geometrical symmetry is made, even though floating objects usually have at least one plane of symmetry. For a total of 48 surface elements, the CPU time is a little less than 2 minutes for one wave length in VAX 780/11 system. Most of the CPU time used is for forming the $[\alpha_{ij}]$ and $[\beta_{ij}]$ matrices given in equations (3.5) and (3.8), which contain $\partial G/\partial n$ and G terms respectively. The total number of elements used to describe the body has a very significant effect on the CPU time. For 196 elements, the CPU time for one frequency is about 35 minutes. It is thus necessary to use as few surface elements as possible to describe the surface sufficiently accurately, without losing the reliability of the calculated results. Many guidelines have been proposed by various investigators, mostly based on experience rather than rigid theoretical principle regarding the size and total number of elements [14]. To ensure that the body is divided into a sufficiently fine mesh, the element lengths should be less than $\frac{1}{8}$ th of the incoming wave length λ. This implies that for accuracy of computation at higher frequencies or lower incoming wave lengths, a larger number of elements should be used. Fortunately, for large floating marine structures such as semisubmersibles, the frequency range of interest is usually not as large and hence this problem does not become too restrictive. Also the neighbouring element sizes should not be widely different, which means that a large element should not be surrounded by comparatively very small elements. This results in computational inefficiencies as the precision offered by smaller elements is lost. Since only plane quadrilateral elements are used, a large number of elements should be used to describe the highly curved regions. It is also preferable to use as squarely shaped elements as possible. This means, for rectangular elements, aspect ratio closer to one is preferable. The evaluation of $\int \frac{1}{R}\, dS$ in equation (3.9) results in more numerical inaccuracies for thin long elements compared to a squarely one. It must be remembered that this integral forms the dominant diagonal elements in matrix $[\beta_{ij}]$. A detailed parametric study on the aspect ratio requirements of the elements for the same geometry is outside the scope of the work presented in the thesis.

One more point which should be noted here is the case of so called 'irregular' frequencies. At these frequencies, matrix $[\alpha_{ij}]$ in equation (3.5) becomes singular and thus the problem cannot be solved by using the integral formula in equation (2.12). So far there has been no theoretical method developed to determine such irregular frequencies for geometries of arbitrary shape. For certain regular geometrical shapes like vertical circular cylinder, these frequencies can be analytically determined [4]. Usually these frequencies correspond to wave lengths of the order of or less than the characteristic length of the body. So far any such problem of irregular frequencies has not been encountered in the present calculations. At this time, a physical explanation for this phenomenon is not obvious. This aspect of the singularity and its interpretation thereof in a subject for further research.

Finally, the computations performed and presented in this thesis show that the program developed calculates the first order wave exciting forces/moments and motion responses in six degrees of freedom correctly, comparing the results with other computations based on the same theoretical model. Also, the computations for semisubmersible demonstrate the versatility and usefulness of the program.

REFERENCES

1. Kim, W.D., "On the Harmonic Oscillation of a Rigid Body on a Free Surface", Journal of Fluid Mechanics, Vol. 21 part 3, 1965, pp. 427-451.

2. Garrison, C.J., "Hydrodynamics of Large Objects in the Sea Part II: Motion of Free-Floating Bodies", J. Hydronautics, Vol. 9, No. 2, 1975, pp. 5-12.

3. Faltinsen, O.M., and Michelsen, F.C., "Motions of Large Structures in Waves at Zero Froude Number", International Symposium on the Dynamics of Marine Vehicles and Structures in Waves, London, England, 1974, pp. 91-106.

4. Nojiri, N., "A Study of Hydrodynamic Pressures and Wave Loads on Three-Dimensional Floating Bodies", JHI Engineering Review, Vol. 14, No. 2, 1981, pp. 6-20.

5. Pinkster, J.A., and Van Oortmerssen, G., "Computation of the First and Second Order Wave Forces on Bodies Oscillating in Regular Waves", Proceedings of 2nd International Conference on Numerical Ship Hydrodynamics, University of California, Berkeley, U.S.A., 1977, pp. 136-156.

6. Hsiung, C.C., and Aboul-Azm, A.F., "Iceberg Drift Affected by Wave Action", Ocean Engineering, Vol. 9, No. 5, 1982.

7. Salvasen, N., Tuck, E.O., and Faltinsen, O.M., "Ship Motions and Sea Loads", Transactions of SNAME, Vol. 78, 1970, pp. 1-30.

8. Hooft, J.P., "A Mathematical Model of Determining Hydrodynamically Induced Forces on Semisubmersibles", Transactions of SNAME, Vol. 79, 1971, pp. 28-70.

9. Lamb, H., "Hydrodynamics", 6th Edition, Cambridge, University Press, 1932.

10. Garrison, C.J., "Hydrodynamic Loading of Large Offshore Structures: Three-Dimensional Source Distribution Methods", Chapter 3, Numerical Methods in Offshore Engineering, John Wiley & Sons, 1978, pp. 87-140.

11. Wehausen, J.V., and Laitone, E.E., "Surface Waves", Encyclopedia of Physics, Vol. 9, Springer-Verlag, Berlin, 1960, pp. 446-778.

12. Hogben, N., and Standing, R.G., "Wave Loads on Large Bodies", International Symposium on the Dynamics of Marine Vehicles and Structures in Waves, London, England, 1974, pp. 258-277.

13. Olsen, O.A., Braathen, A., Loken, A.E., Nyhus, K.A., and Torset, O.P., "Slow and High Frequency Motions and Loads of Articulated Single Point Mooring Systems for Large Tankers", Det Norske Veritas.

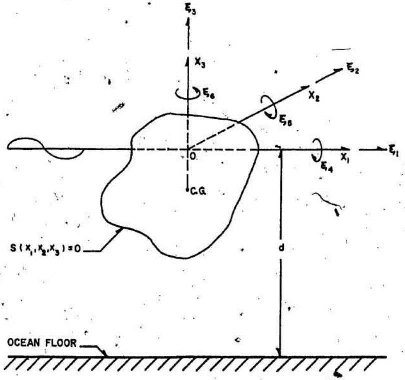
14. Sarpakaya, T., and Isaasson, M., "Mechanics of Wave Forces", Van Nostrand Co., 1981.

15. Arockiasamy, M., and Reddy, D.V., "Analysis of Certain Offshore Structures Vol. II: Analysis and Concepts", St. John's, Newfoundland, June 1982.

TABLE 1

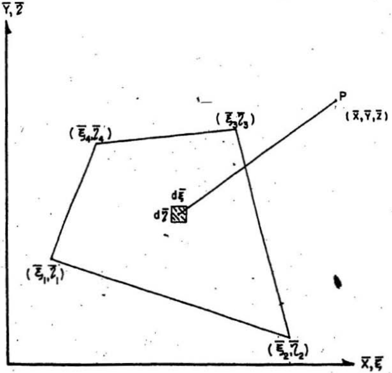
Main Particulars of 130,000 tons Dwt. Tanker

PARAMETER	UNIT	CONDITION	
		Ballast	Loaded
Length between perpendiculars	m	285.60	285.60
Beam	m	46.71	46.71
Depth	m	20.35	20.35
Draft, fore	m	4.84	13.82
Draft, aft	m	7.04	13.82
Draft, mean	m	5.94	13.82
Longitudinal centre of gravity (+ve means forward of midship)	m	+2.10	+6.46
Vertical Centre of gravity from baseline	m	9.73	11.03
Metacentric height, transverse	m	21.50	8.97
Pitch/yaw radius of gyration	m	71.40	71.40
Roll Radius of gyration	m	16.35	16.35



CO-ORDINATE SYSTEM AND GEOMETRICAL BOUNDARIES

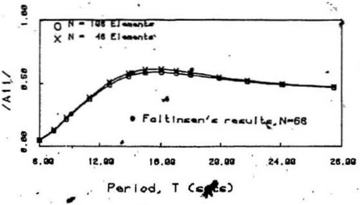
FIGURE - 1



LOCAL CO-ORDINATE SYSTEM OF A PLANE

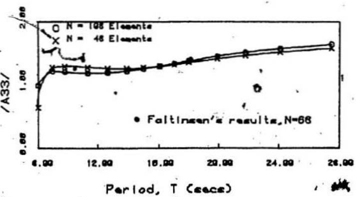
QUADRILATERAL ELEMENT

FIGURE - 2



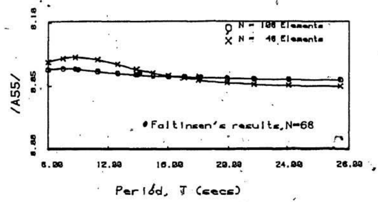
SURGE ADDED MASS COEFFICIENT FOR FLOATING BOX

FIGURE - 3



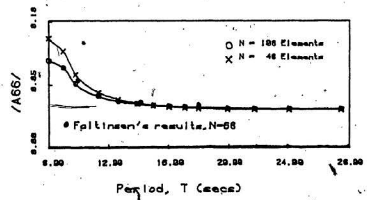
HEAVE ADDED MASS COEFFICIENT FOR FLOATING BOX

FIGURE - 4



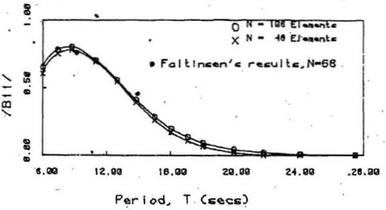
PITCH ADDED MASS COEFFICIENT FOR FLOATING BOX

FIGURE - 5



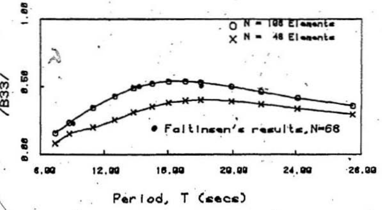
YAW ADDED MASS COEFFICIENT FOR FLOATING BOX

FIGURE - 6



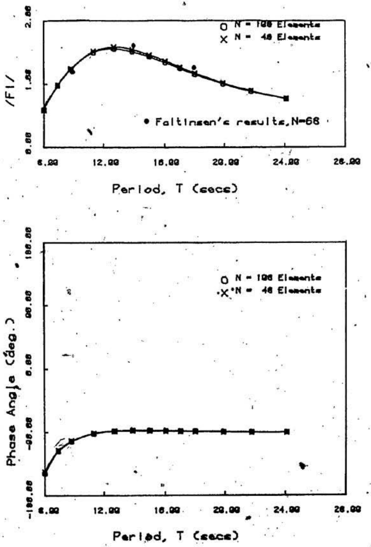
SURGE DAMPING COEFFICIENT FOR FLOATING BOX

FIGURE - 7



HEAVE DAMPING COEFFICIENT FOR FLOATING BOX

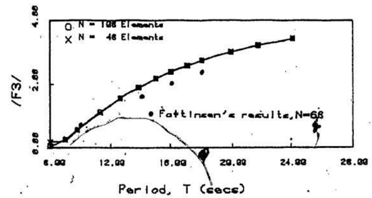
FIGURE - 8

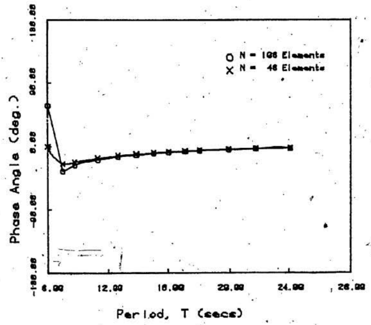


SURGE EXCITING FORCE ON FLOATING BOX

Amplitudes and Phases

FIGURE - 9

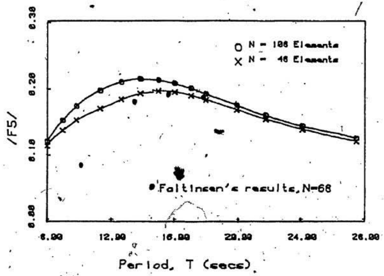


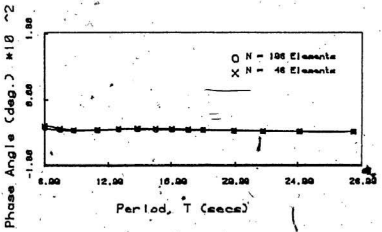


HEAVE EXCITING FORCE ON FLOATING BOX

Amplitudes and Phases

FIGURE - 10

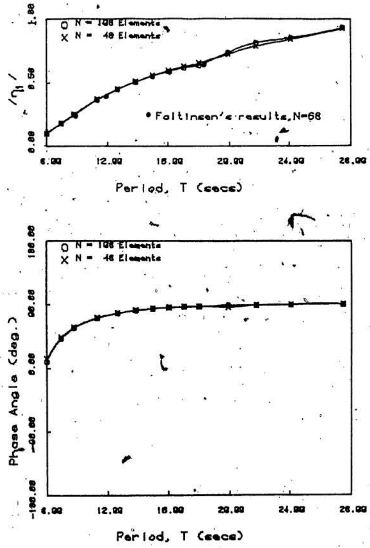




PITCH EXCITING MOMENT ON FLOATING BOX

Amplitudes and Phases

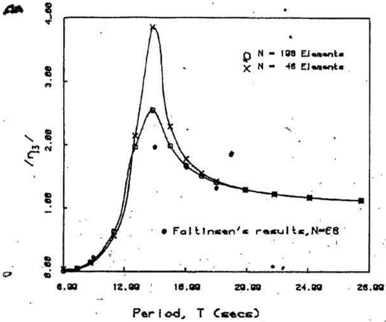
FIGURE - 11

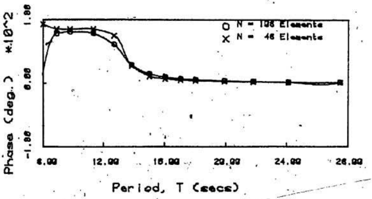


SURGE MOTION OF FLOATING BOX

Non-dimensional Amplitudes and Phases

FIGURE - 12

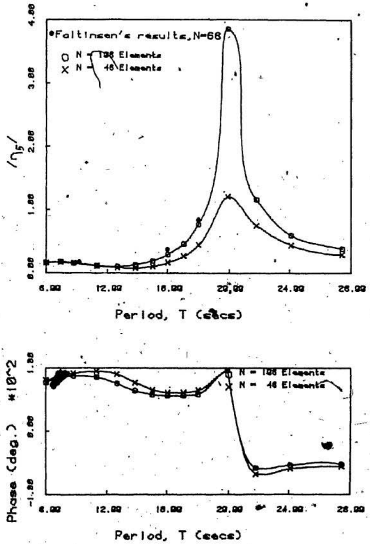




HEAVE MOTION OF FLOATING BOX

Non-dimensional Amplitudes and Phases

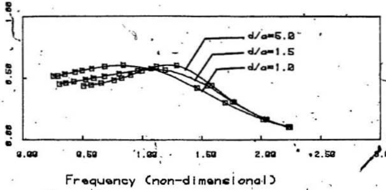
FIGURE - 13



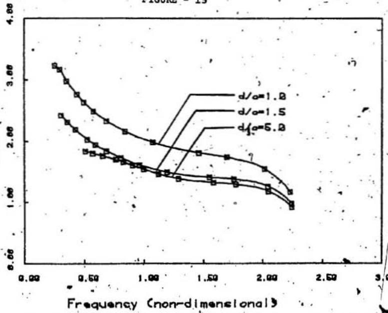
PITCH MOTION OF FLOATING BOX

Non-dimensional Amplitudes and Phases

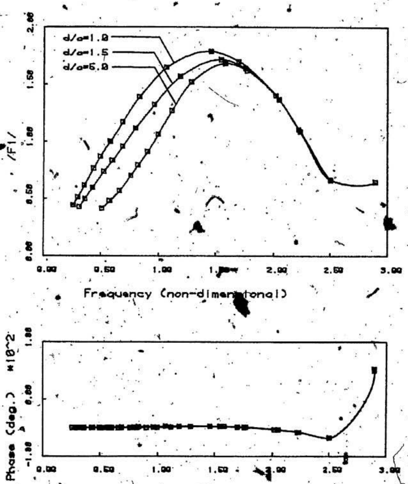
FIGURE - 14



SURGE ADDED MASS FOR VERTICAL CIRCULAR CYLINDER
FIGURE - 15

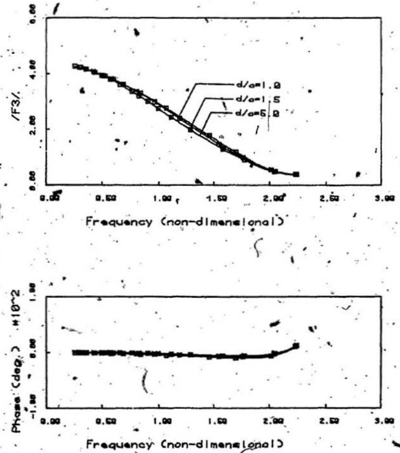


HEAVE ADDED MASS FOR VERTICAL CIRCULAR CYLINDER
FIGURE - 16



SURGE EXCITING FORCE ON VERTICAL CIRCULAR CYLINDER
Amplitudes and Phases

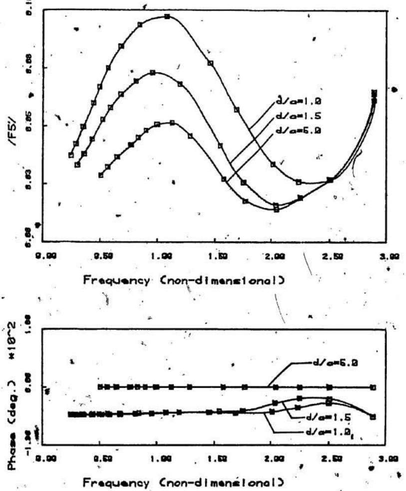
FIGURE - 17



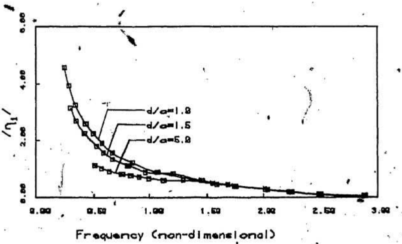
HEAVE EXCITING FORCE ON VERTICAL CIRCULAR CYLINDER

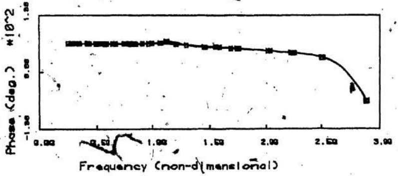
Amplitudes and Phases

FIGURE - 18



PITCH EXCITING MOMENT ON VERTICAL CIRCULAR CYLIN.
Amplitudes and Phases
FIGURE - 19

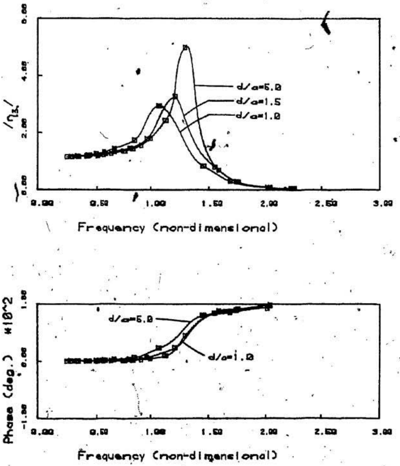




SURGE MOTION OF VERTICAL CIRCULAR CYLINDER

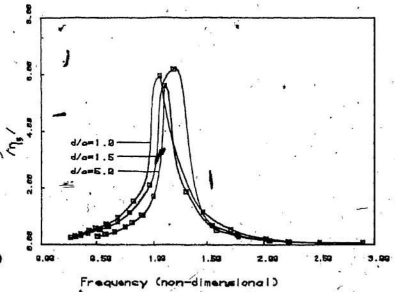
Non-dimensional Amplitudes and Phases

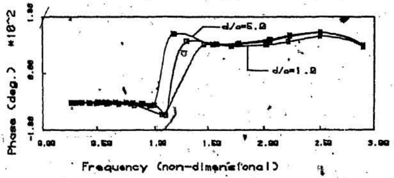
FIGURE - 20



HEAVE MOTION OF VERTICAL CIRCULAR CYLINDER
Non-dimensinal Amplitudes and Phases

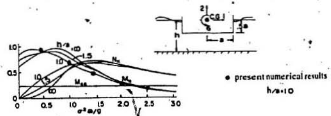
FIGURE - 21



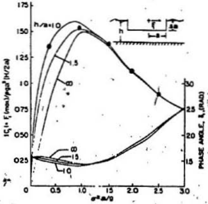


PITCH MOTION OF VERTICAL CIRCULAR CYLINDER
Non-dimensional Amplitudes and Phases

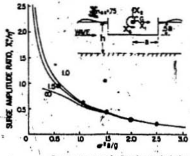
FIGURE - 22



Surge Added mass and damping coefficients for a floating vertical circular cylinder



Horizontal force coefficient for a floating vertical circular cylinder

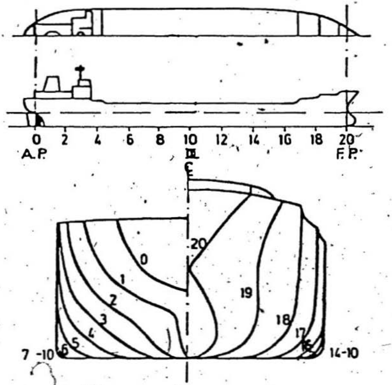


Surge response of a floating vertical circular cylinder

GARRISON'S RESULTS FOR VERTICAL CIRCULAR

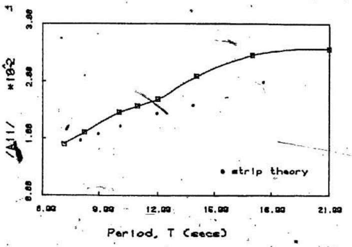
CYLINDER FOR SURGE MODE

FIGURE - 23

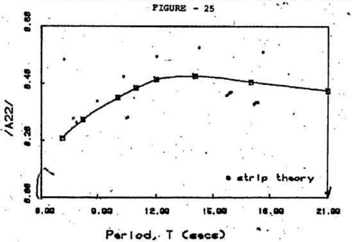


GEOMETRY OF THE TANKER

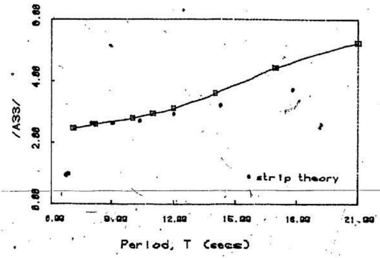
FIGURE - 24



SURGE ADDED MASS COEFF. FOR TANKER (BALLAST)
FIGURE - 25

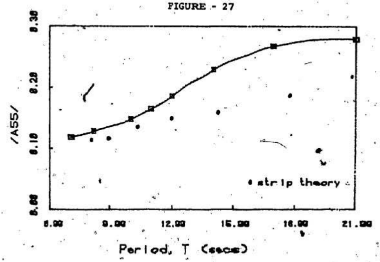


SWAY ADDED MASS COEFF. FOR TANKER (BALLAST)
FIGURE - 26



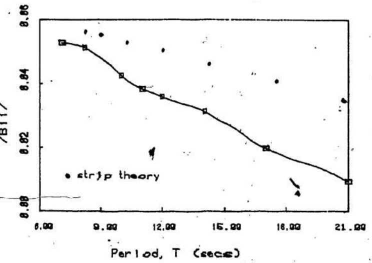
HEAVE ADDED MASS COEFF. FOR TANKER (BALLAST)

FIGURE - 27



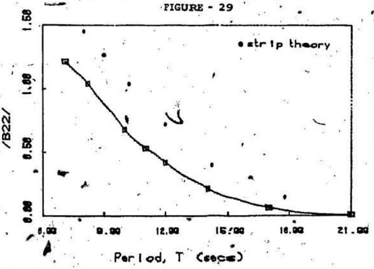
PITCH ADDED MASS COEFF. FOR TANKER (BALLAST)

FIGURE - 28



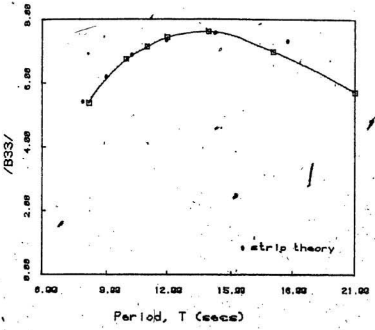
SURGE DAMPING COEFF. FOR TANKER (BALLAST)

FIGURE - 29



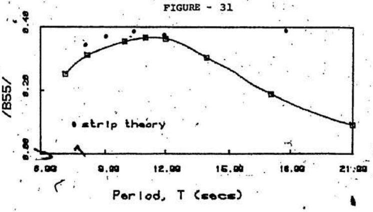
SWAY DAMPING COEFF. FOR TANKER (BALLAST)

FIGURE - 30



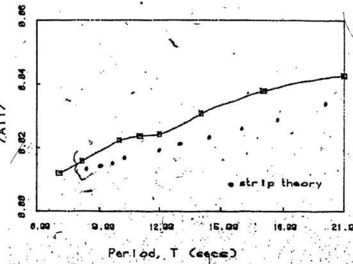
HEAVE DAMPING COEFF. FOR TANKER (BALLAST)

FIGURE - 31

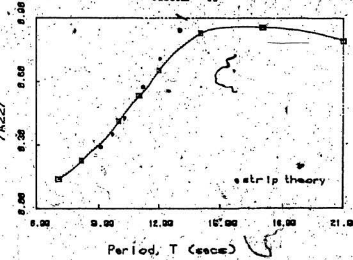


PITCH DAMPING COEFF. FOR TANKER (BALLAST)

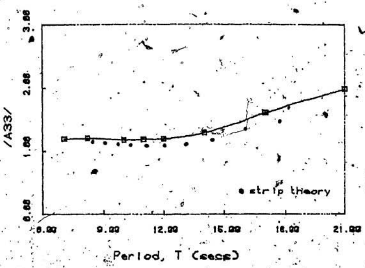
FIGURE - 32



SURGE ADDED MASS COEFF. FOR TANKER (LOADED)
FIGURE - 33

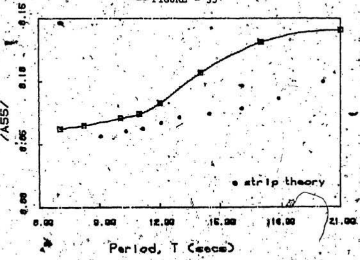


SWAY ADDED MASS COEFF. FOR TANKER (LOADED)
FIGURE - 34



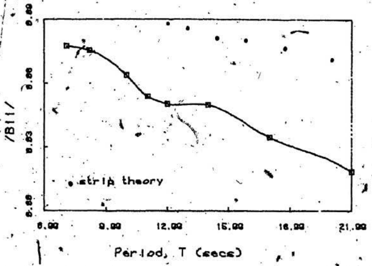
HEAVE ADDED MASS COEFF. FOR TANKER (LOADED)

FIGURE - 35



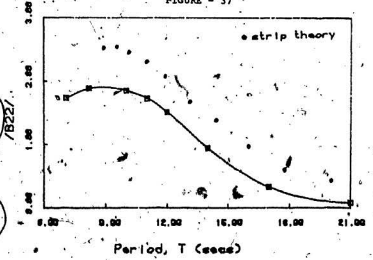
PITCH ADDED MASS COEFF. FOR TANKER (LOADED)

FIGURE - 36



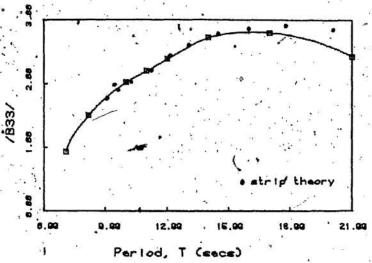
SURGE DAMPING COEFF. FOR TANKER (LOADED)

FIGURE - 37



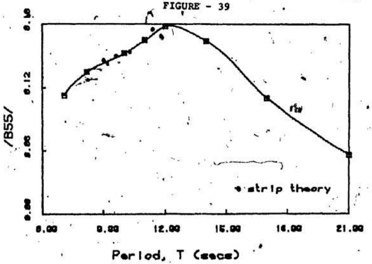
SWAY DAMPING COEFF. FOR TANKER (LOADED)

FIGURE - 38



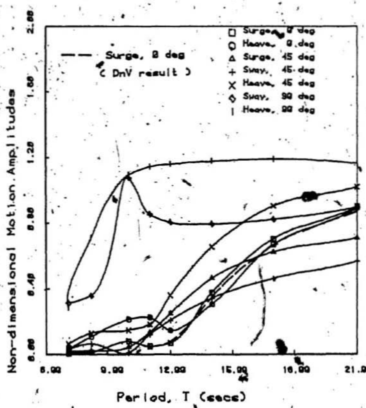
HEAVE DAMPING COEFF. FOR TANKER (LOADED)

FIGURE - 39



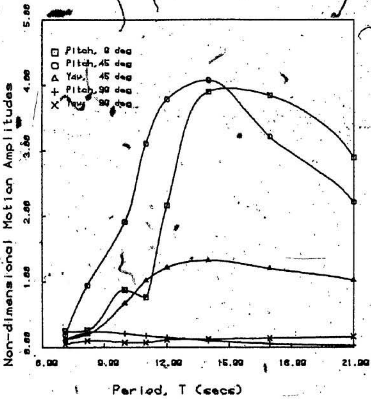
PITCH DAMPING COEFF. FOR TANKER (LOADED)

FIGURE - 40



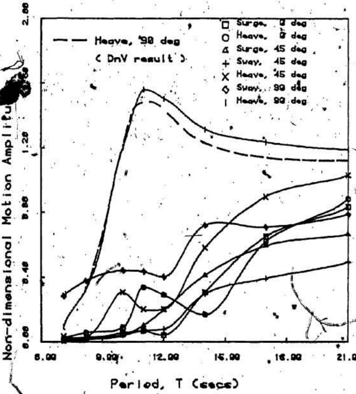
MOTION RESPONSE OF THE TANKER (BALLAST COND.)
(Surge, Sway and Heave)

FIGURE - 41



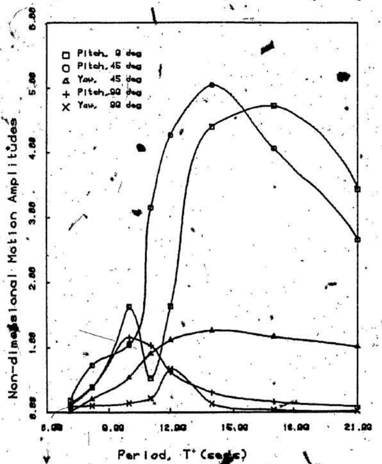
MOTION RESPONSE OF THE TANKER (BALLAST COND.)
(Pitch and Yaw)

FIGURE - 42



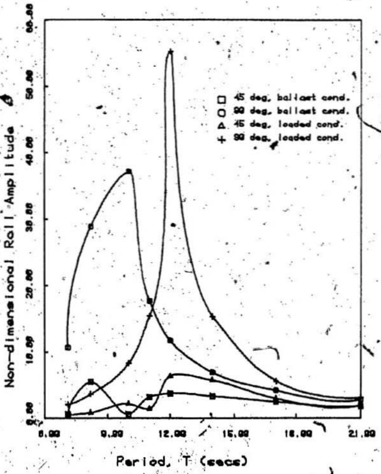
MOTION RESPONSE OF THE TANKER (LOADED COND.)
(Surge, Sway and Heave)

FIGURE - 43



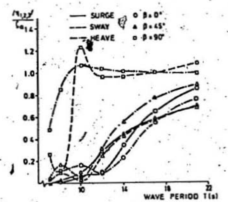
MOTION RESPONSE OF THE TANKER (LOADED COND.)
(Pitch and Yaw)

FIGURE - 44



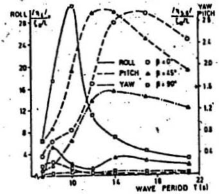
ROLL RESPONSE OF THE TANKER
(Ballast and Loaded Conditions)

FIGURE - 45



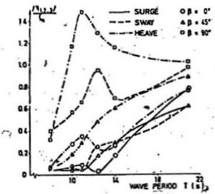
DnV RESULTS FOR MOTION RESPONSE OF THE TANKER, BALLAST (SURGE, SWAY AND HEAVE)

FIGURE - 46



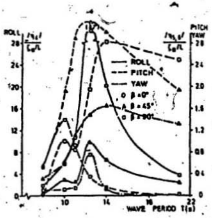
DnV RESULTS FOR MOTION RESPONSE OF THE TANKER, BALLAST (ROLL, PITCH AND YAW)

FIGURE - 47



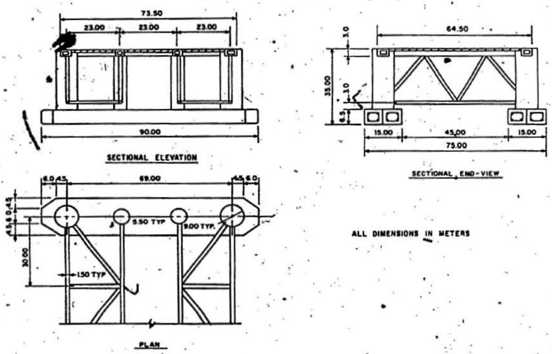
DnV RESULTS FOR MOTION RESPONSE OF THE TANKER, LOADED (SURGE, SWAY AND HEAVE)

FIGURE - 48



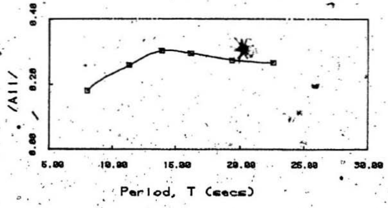
DnV RESULTS FOR MOTION RESPONSE OF THE TANKER, LOADED (ROLL, PITCH AND YAW)

FIGURE - 49



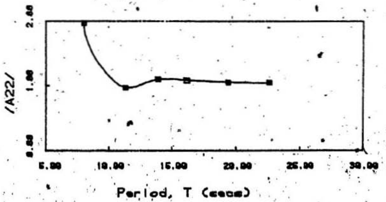
SECTIONAL VIEWS OF THE SEMISUBMERSIBLE

FIGURE - 50



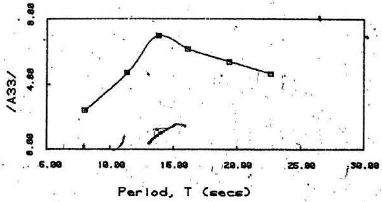
SURGE ADDED MASS COEFF. FOR SEMISUBMERSIBLE

FIGURE - 51



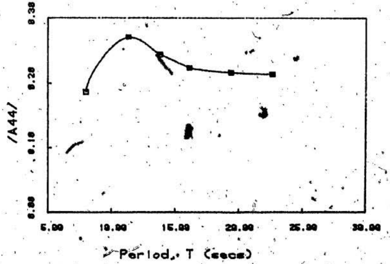
SWAY ADDED MASS COEFF. FOR SEMISUBMERSIBLE

FIGURE - 52



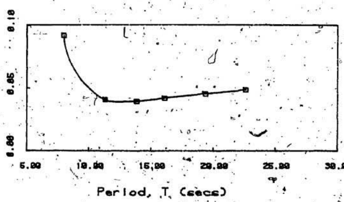
HEAVE ADDED MASS COEFF. FOR SEMISUBMERSIBLE

FIGURE - 53

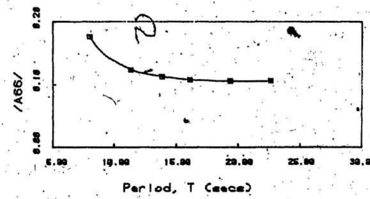


ROLL ADDED MASS COEFF. FOR SEMISUBMERSIBLE

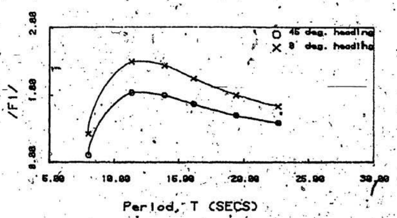
FIGURE - 54



PITCH ADDED MASS COEFF. FOR SEMISUBMERSIBLE
FIGURE - 55

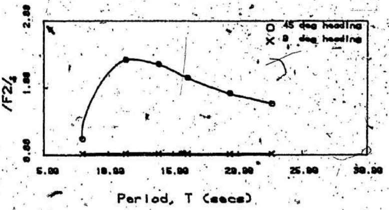


YAW ADDED MASS COEFF. FOR SEMISUBMERSIBLE
FIGURE - 56



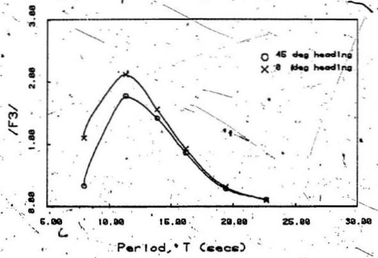
SURGE EXCITING FORCE ON SEMISUBMERSIBLE.

FIGURE - 57



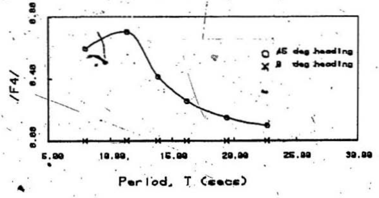
SWAY EXCITING FORCE ON SEMISUBMERSIBLE

FIGURE - 58



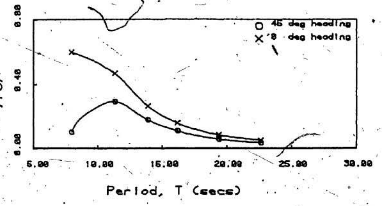
HEAVE EXCITING FORCE ON SEMISUBMERSIBLE

FIGURE - 59



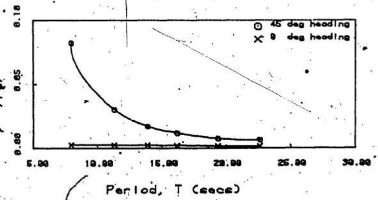
ROLL EXCITING MOMENT ON SEMISUBMERSIBLE

FIGURE - 60



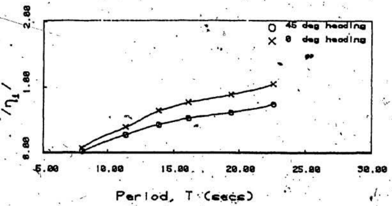
PITCH EXCITING MOMENT ON SEMISUBMERSIBLE

FIGURE - 61



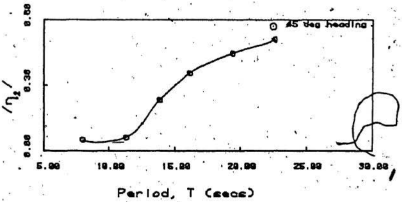
YAW EXCITING MOMENT ON SEMISUBMERSIBLE

FIGURE - 62



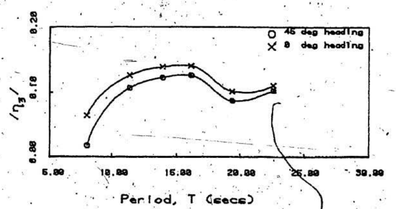
SURGE MOTION OF SEMISUBMERSIBLE

FIGURE - 63



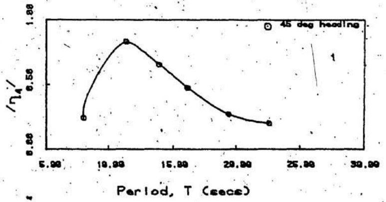
SWAY MOTION OF SEMISUBMERSIBLE

FIGURE - 64



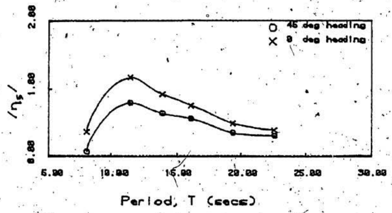
HEAVE MOTION OF SEMISUBMERSIBLE

FIGURE - 65



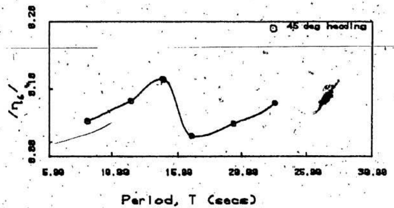
ROLL MOTION OF SEMISUBMERSIBLE

FIGURE - 66



PITCH MOTION OF SEMISUBMERSIBLE

FIGURE - 67



YAW MOTION OF SEMISUBMERSIBLE

FIGURE - 68

APPENDIX A

LISTING OF THE COMPUTER PROGRAM

```
C***********************************************************************
C                                                                      *
C                       P A R T   I                                    *
C                                                                      *
C***********************************************************************
C                                                                      *
C     THIS PRORAM CALCULATES THE AREA (SAREA), CO-ORDINATES OF THE     *
C     CENTROID (XCG,YCG,ZCG), THE THREE COMPONENTS OF THE OUTWARD      *
C     NORMAL AND THE SHAPE FACTORS OF THE BODY-SURFACE PANELS          *
C                                                                      *
C     THE SHAPE FACTOR (SHFACT) IS A FACTOR RELATING THE INTEGRATION   *
C     OF THE SINGULAR TERM IN GREEN'S FUNCTION                         *
C                                                                      *
C***********************************************************************
C
      DIMENSION BX1(128),BY1(128),BZ1(128),BX2(128),BY2(128),
     1 BZ2(128),BX3(128),BY3(128),BZ3(128),BX4(128),BY4(128),
     2 BZ4(128),SAREA(128),AAN1(128),AAN2(128),AAN3(128),
     3 XCG(128),YCG(128),ZCG(128),NXYZ(128),SHFACT(128)
C
C     READING INPUT DATA
C
C     INPUT DATA TO BE READ ARE THE CO-ORDINATES OF THE VERTICES
C     OF THE PANELS
C
      CALL ASSIGN(1,'HULL.DAT')
      READ(1,*) NP
      DO 275 I=1,NP
      READ(1,*) NXYZ(I)
      IF(NXYZ(I) .EQ. 3) GO TO 276
      READ(1,*) BX1(I),BY1(I),BZ1(I)
      READ(1,*) BX2(I),BY2(I),BZ2(I)
      READ(1,*) BX3(I),BY3(I),BZ3(I)
      READ(1,*) BX4(I),BY4(I),BZ4(I)
      GO TO 275
 276  READ(1,*) BX1(I),BY1(I),BZ1(I)
      READ(1,*) BX2(I),BY2(I),BZ2(I)
      READ(1,*) BX3(I),BY3(I),BZ3(I)
 275  CONTINUE
      TYPE *,NP
      DO 533 I=1,NP
      TYPE *,NXYZ(I)
      IF(NXYZ(I) .EQ. 3) GO TO 534
      TYPE *,BX1(I),BY1(I),BZ1(I)
      TYPE *,BX2(I),BY2(I),BZ2(I)
      TYPE *,BX3(I),BY3(I),BZ3(I)
      TYPE *,BX4(I),BY4(I),BZ4(I)
      GO TO 533
 534  TYPE *,BX1(I),BY1(I),BZ1(I)
      TYPE *,BX2(I),BY2(I),BZ2(I)
      TYPE *,BX3(I),BY3(I),BZ3(I)
 533  CONTINUE
C
C     CALCULATION FOR THE DIRECTION COSINES
C
      DO 269 I=1,NP
      X1=BX1(I)
      X2=BX2(I)
```

```
      X3=BX3(I)
      Y1=BY1(I)
      Y2=BY2(I)
      Y3=BY3(I)
      Z1=BZ1(I)
      Z2=BZ2(I)
      Z3=BZ3(I)
      NNXYZ=NXYZ(I)
      IF (NNXYZ .EQ. 3) GO TO 281
      X4=BX4(I)
      Y4=BY4(I)
      Z4=BZ4(I)
      GO TO 282
281   X4=X3
      Y4=Y3
      Z4=Z3
282   CONTINUE
      NN=60
      TYPE *,'GIVEN POINTS'
      TYPE *,X1,Y1,Z1
      TYPE *,X2,Y2,Z2
      TYPE *,X3,Y3,Z3
      TYPE *,X3,Y4,Z4
      XX1=0.0
      YY1=0.0
      ZZ1=0.0
      XX2=X2-X1
      YY2=Y2-Y1
      ZZ2=Z2-Z1
      XX3=X3-X1
      YY3=Y3-Y1
      ZZ3=Z3-Z1
      XX4=X4-X1
      YY4=Y4-Y1
      ZZ4=Z4-Z1
      A1=XX2-XX1
      B1=YY2-YY1
      C1=ZZ2-ZZ1
      ABC1=SQRT(A1*A1 + B1*B1 + C1*C1)
      XX=A1/ABC1
      XY=B1/ABC1
      XZ=C1/ABC1
      TYPE *,'XX=',XX,'XY=',XY,'XZ=',XZ
      A3=(YY2-YY1)*(ZZ4-ZZ1) - (YY4-YY1)*(ZZ2-ZZ1)
      B3=(XX4-XX1)*(ZZ2-ZZ1) - (XX2-XX1)*(ZZ4-ZZ1)
      C3=(XX2-XX1)*(YY4-YY1) - (XX4-XX1)*(YY2-YY1)
      ABC3=SQRT(A3*A3 + B3*B3 + C3*C3)
      ZX=A3/ABC3
      ZY=B3/ABC3
      ZZ=C3/ABC3
      TYPE *,'OUT NORMAL'
      TYPE *,'N1=',ZX,' N2=',ZY,' N3=',ZZ
      A2=B3*C1 - B1*C3
      B2=A1*C3 - A3*C1
      C2=A3*B1 - A1*B3
      ABC2=SQRT(A2*A2 + B2*B2 + C2*C2)
      YX=A2/ABC2
      YY=B2/ABC2
```

```
      YZ=C2/ABC2
      TYPE *,'YX=',YX,'YY=',YY,'YZ=',YZ
C
C     CALCULATION FOR THE PANEL AREA
C
      S1=SQRT((XX2-XX1)**2 + (YY2-YY1)**2 +(ZZ2-ZZ1)**2)
      S2=SQRT((XX3-XX2)**2 + (YY3-YY2)**2 + (ZZ3-ZZ2)**2)
      S=SQRT((XX3-XX1)**2 + (YY3-YY1)**2 + (ZZ3-ZZ1)**2)
      IF (NNXYZ .EQ. 3) GO TO 300
      S3=SQRT((XX4-XX3)**2 + (YY4-YY3)**2 + (ZZ4-ZZ3)**2)
      S4=SQRT((XX1-XX4)**2 + (YY1-YY4)**2 + (ZZ1-ZZ4)**2)
      AS1=(S1+S2+S)*.5
      AS2=(S3+S4+S)*.5
      AR1=SQRT(AS1*(AS1-S1)*(AS1-S2)*(AS1-S))
      AR2=SQRT(AS2*(AS2-S3)*(AS2-S4)*(AS2-S))
      AREA=AR1+AR2
      GO TO 305
  300 AS1=(S1+S2+S)*.5
      AREA=SQRT(AS1*(AS1-S1)*(AS1-S2)*(AS1-S))
  305 CONTINUE
      TYPE *,'AREA=',AREA
C
C     CALCULATION OF THE CENTROID OF THE PANELS
C
C
      IF (NNXYZ .EQ. 3) GO TO 290
      XXG1=(X1+X2+X3)/3.
      YYG1=(Y1+Y2+Y3)/3.
      ZZG1=(Z1+Z2+Z3)/3.
      XXG2=(X1+X3+X4)/3.
      YYG2=(Y1+Y3+Y4)/3.
      ZZG2=(Z1+Z3+Z4)/3.
      XG=(AR1*XXG1+AR2*XXG2)/AREA
      YG=(AR1*YYG1+AR2*YYG2)/AREA
      ZG=(AR1*ZZG1+AR2*ZZG2)/AREA
      GO TO 295
  290 XG=(X1+X2+X3)/3.
      YG=(Y1+Y2+Y3)/3.
      ZG=(Z1+Z2+Z3)/3.
  295 CONTINUE
      TYPE *,'XG=',XG,'YG=',YG,'ZG=',ZG
C
C     CALCULATION FOR THE SHAPE FACTORS
C
C     CHECK WHETHER THE PANELS ARE RECTANGULAR
C     ( FOR RECTANGULAR PANELS, AN ANALYTICAL EXPRESSION IS USED
C       TO CALCULATE THE SHAPE FACTORS )
C
      IF (NNXYZ .EQ. 3) GO TO 310
      DIAG1=SQRT((X1-X3)**2 + (Y1-Y3)**2 + (Z1-Z3)**2)
      DIAG2=SQRT((X2-X4)**2 + (Y2-Y4)**2 + (Z2-Z4)**2)
      TYPE *,'DIAG1=',DIAG1,'DIAG2=',DIAG2
      IF (ABS(DIAG1-DIAG2) .LT. .00001) GO TO 320
  310 CONTINUE
C     CHANGE ALL POINTS IN LOCAL COORDINATES
      X=XG-X1
      Y=YG-Y1
      Z=ZG-Z1
```

```
      E1=XX*XX1 + XY*YY1 + XZ*ZZ1
      F1=YX*XX1 + YY*YY1 + YZ*ZZ1
      G1=ZX*XX1 + ZY*YY1 + ZZ*ZZ1
      E2=XX*XX2 + XY*YY2 + XZ*ZZ2
      F2=YX*XX2 + YY*YY2 + YZ*ZZ2
      G2=ZX*XX2 + ZY*YY2 + ZZ*ZZ2
      E3=XX*XX3 + XY*YY3 + XZ*ZZ3
      F3=YX*XX3 + YY*YY3 + YZ*ZZ3
      G3=ZX*XX3 + ZY*YY3 + ZZ*ZZ3
      E4=XX*XX4 + XY*YY4 + XZ*ZZ4
      F4=YX*XX4 + YY*YY4 + YZ*ZZ4
      G4=ZX*XX4 + ZY*YY4 + ZZ*ZZ4
      XN=XX*X + XY*Y + XZ*Z
      YN=YX*X + YY*Y + YZ*Z
      ZN=ZX*X + ZY*Y + ZZ*Z
      TYPE *,'LOCAL COORDINATES'
      TYPE *,E1,F1,G1
      TYPE *,E2,F2,G2
      TYPE *,E3,F3,G3
      TYPE *,E4,F4,G4
      TYPE *,XN,YN,ZN
C     DETERMINE THE INTEGRATION
      ITER=1
      MN=NN
 101  CONTINUE
      INDEX=1
  50  CONTINUE
      GO TO(5,6,7,8),INDEX
   5  ET2=E2
      ET1=E1
      FT2=F2
      FT1=F1
      GO TO 10
   6  ET2=E3
      ET1=E2
      FT2=F3
      FT1=F2
      GO TO 10
   7  CONTINUE
      IF (NNXYZ .EQ. 3) GO TO 330
      ET2=E4
      ET1=E3
      FT2=F4
      FT1=F3
      GO TO 10
 330  ET2=E3
      ET1=E1
      FT2=F3
      FT1=F1
      GO TO 10
   8  ET2=E1
      ET1=E4
      FT2=F1
      FT1=F4
  10  CONTINUE
      SUM=0.0
      IK=1
      STEP=(ET2-ET1)/MN
```

```
      ET12=ABS(ET2-ET1)
      IF(ET12 .LE. .00001) GO TO 20
      IF (ABS(STEP) .LE. .00001) GO TO 75
      IF(STEP .LT. 0) IK=-1
      EZ1=ET1
65    IF(ABS(EZ1-XN) .LE. .0001) GO TO 60
      FN=FT1+(FT2-FT1)*(EZ1-ET1)/(ET2-ET1)
      EZ=EZ1
      CALL SUB(XN,YN,ZN,EZ,FN,FZ)
      FZ1=FZ
30    EZ2=EZ1+STEP

55    IF(ABS(EZ2-XN) .LE. .0001) GO TO 51
      IF(IK .EQ. 1) GO TO 15
      IF(EZ2 .LT. ET2) GO TO 20
      GO TO 25
15    IF(EZ2 .GT. ET2) GO TO 20
25    CONTINUE
      FN=FT1+(FT2-FT1)*(EZ2-ET1)/(ET2-ET1)
      EZ=EZ2
      CALL SUB(XN,YN,ZN,EZ,FN,FZ)
      FZ2=FZ
      SUM=SUM+.5*(FZ1+FZ2)*(EZ2-EZ1)
      EZ1=EZ2
      FZ1=FZ2
      GO TO 30
51    EZ2=EZ2+STEP
      GO TO 55
60    EZ1=EZ1+STEP
      GO TO 65
20    CONTINUE
      GO TO(70,71,72,73),INDEX
70    SUM1=SUM
      GO TO 75
71    SUM2=SUM
      GO TO 75
72    SUM3=SUM
      GO TO 75
73    SUM4=SUM
75    CONTINUE
      INDEX=INDEX+1
      IF(INDEX .GT. NNXYZ) GO TO 80
      GO TO 50
80    FSUM=SUM1+SUM2+SUM3+SUM4
      IF(ITER .EQ. 1) GO TO 100
      FSUM2=FSUM
      CONV=ABS((FSUM2-FSUM1)/FSUM2)
      IF(CONV .LE. .005) GO TO 102
      ITER=ITER+1
      FSUM1=FSUM2
      MN=MN*2
      TYPE *,'SUM2=',FSUM2,'SUM1=',FSUM1,'CONV=',CONV
      GO TO 101
100   FSUM1=FSUM
      TYPE *,'SUM1=',FSUM1
      ITER=ITER+1
      MN=MN*2
      GO TO 101
```

```
 102   FSUM=FSUM2
       SFACT=.5*SQRT(1./(AREA*3.14159265))*FSUM
       GO TO 400
 320   CONTINUE
C
C
       SIDEA=SQRT((X2-X1)**2 +(Y2-Y1)**2 + (Z2-Z1)**2)
       SIDEB=SQRT((X4-X1)**2 +(Y4-Y1)**2 + (Z4-Z1)**2)
       ASPECT=SIDEA/SIDEB
       IF (ASPECT .LT. 1) ASPECT=1./ASPECT
       BBB1=SQRT(ASPECT**2 + 1)
       BBB2=ALOG(ASPECT+BBB1)
       BBB3=ASPECT*ALOG((1+BBB1)/ASPECT)
       SFACT=(BBB2+BBB3)/(SQRT(3.14159265*ASPECT))
 400   CONTINUE
C
       XCG(I)=XG
       YCG(I)=YG
       ZCG(I)=ZG
       AAN1(I)=ZX
       AAN2(I)=ZY
       AAN3(I)=ZZ
       SAREA(I)=AREA
       SHFACT(I)=SFACT
C
 200   CONTINUE
C
       WRITE(2,1000)
 1000  FORMAT(5X,3HXCG,7X,3HYCG,7X,3HZCG,6X,2HNx,6X,2HNy,6X,2HNz,
     1 7X,4HAREA,4X,6HSHFACT//)
       DO 450 I=1,NP
       TYPE *,I,XCG(I),YCG(I),ZCG(I),AAN1(I),AAN2(I),AAN3(I),
     1 SAREA(I),SHFACT(I)
       WRITE(2,1001)I,XCG(I),YCG(I),ZCG(I),AAN1(I),AAN2(I),AAN3(I),
     1 SAREA(I),SHFACT(I)
 450   CONTINUE
 1001  FORMAT(I5,3F10.4,3F6.4,F10.4,F10.6)
       STOP
       END
       SUBROUTINE SUB(XN,YN,ZN,EZ,FN,FZ)
       T1=(YN-FN)**2 +(XN-EZ)**2 + ZN**2
       T2=SQRT(T1)
       T3=(YN-FN)+T2
       FZ=ALOG(T3)
       RETURN
       END
```

```
C*********************************************************************
C                                                                    *
C                          P A R T : 2                               *
C                                                                    *
C*********************************************************************
C
C     THIS PROGRAM CALCULATES THE WAVE LOADS AND MOTIONS OF FLOATING *
C     MARINE STRUCTURES USING THREE-DIMENSIONAL SINGULARITY DISTRIBUTION *
C     THEORY ( GREEN'S FUNCTION METHOD )                             *
C                                                                    *
C*********************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
      DIMENSION H(250)
      DIMENSION X1(250),X2(250),X3(250),AN1(250),AN2(250),AN3(250),
     1AMU(100),ADMAS(6,6),DAMP(6,6),TMAS(6,6),REST(6,6) ,S(250)
      DIMENSION ASPECT(250),SHFACT(250),WVLEN(30),ALFA(8),WVSTP(4),
     1ADM11(30),ADM22(30),ADM33(30),ADM44(30),ADM55(30),ADM66(30),
     2DMP11(30),DMP22(30),DMP33(30),DMP44(30),DMP55(30),DMP66(30),
     3XAMP1(30),XAMP2(30),XAMP3(30),XAMP4(30),XAMP5(30),XAMP6(30),
     4XPH1(30),XPH2(30),XPH3(30),XPH4(30),XPH5(30),XPH6(30),
     5ZMAM1(30),ZMAM2(30),ZMAM3(30),ZMAM4(30),ZMAM5(30),ZMAM6(30),
     6ZPH1(30),ZPH2(30),ZPH3(30),ZPH4(30),ZPH5(30),ZPH6(30),
     7FAMP1(30),FAMP2(30),FAMP3(30),FAMP4(30),FAMP5(30),FAMP6(30),
     8FPH1(30),FPH2(30),FPH3(30),FPH4(30),FPH5(30),FPH6(30)
      DIMENSION WE(30),TIME(30),WMP(30)
      COMPLEX*16 B(250,7),PP1,PP2,D,TEMP,DE,F(250,7),SUM,
     1G(250,250),PHI(250,7),POTIN(250),TT,FEXT(6),AA(6,6),CC(6),ZAMP(6),
     2SSUM,FHYD(6)
      COMPLEX*16 FXAM1(8,30),FXAM2(8,30),FXAM3(8,30),FXAM4(8,30),
     1FXAM5(8,30),FXAM6(8,30),ZMOT1(8,30),ZMOT2(8,30),ZMOT3(8,30),
     2ZMOT4(8,30),ZMOT5(8,30),ZMOT6(8,30),FHAM1(8,30),FHAM2(8,30),
     3FHAM3(8,30),FHAM4(8,30),FHAM5(8,30),FHAM6(8,30)
      COMPLEX*16 A(250,250),C(250)
C
      SQRT(X)=DSQRT(X)
      EXP(X)=DEXP(X)
      COS(X)=DCOS(X)
      SIN(X)=DSIN(X)
      ABS(X)=DABS(X)
      ALOG(X)=DLOG(X)
      ATAN(X)=DATAN(X)
C
C     READING INPUT DATA
C
      READ(5,*) IAXIS
      READ(5,*) NP
      DO 2011 I=1,NP
      READ(5,*) X1(I),X2(I),X3(I),AN1(I),AN2(I),AN3(I),S(I),SHFACT(I)
 2011 CONTINUE
      IF(IAXIS .NE. 1) GO TO 3779
      DO 3780 I=1,NP
      X1(NP+I)=X1(I)
      X2(NP+I)=-X2(I)
      X3(NP+I)=X3(I)
      AN1(NP+I)=AN1(I)
      AN2(NP+I)=-AN2(I)
```

```
      AN3(NP+I)=AN3(I)
      S(NP+I)=S(I)
      SHFACT(NP+I)=SHFACT(I)
3780  CONTINUE
      NP=2*NP
3779  CONTINUE
      READ(5,*) SMAS,TI44,TI46,TI55,TI64,TI66,C33,C35,C44,C55
      READ(5,*) NROOT,NWVLN,NHEAD,NWSTP
      READ(5,*) (WVLEN(I),I=1,NWVLN)
      READ(5,*) (ALFA(I),I=1,NHEAD)
      READ(5,*) (WVSTP(I),I=1,NWSTP)
2015  FORMAT(4I5)
2016  FORMAT(8F10.4)
      READ(5,*) DF,CH,GRAV,RHO,VOLM,ALLN
2017  FORMAT(6F12.4)
      WRITE(6,2010)NP
      DO 2021 I=1,NP
      WRITE(6,2012)X1(I),X2(I),X3(I),AN1(I),AN2(I),AN3(I),S(I),SHFACT(I)
2021  CONTINUE
      WRITE(6,2085) SMAS,TI44,TI46,TI55,TI64,TI66,C33,C35,C44,C55
      WRITE(6,2015) NROOT,NWVLN,NHEAD,NWSTP
      WRITE(6,2016) (WVLEN(I),I=1,NWVLN)
      WRITE(6,2016) (ALFA(I),I=1,NHEAD)
      WRITE(6,2016) (WVSTP(I),I=1,NWSTP)
      WRITE(6,2017)DF,CH,GRAV,RHO,VOLM,ALLN
2010  FORMAT(I5)
2012  FORMAT(7F10.4,F10.6)
2013  FORMAT(10F8.2)
2014  FORMAT(I5,7F10.4)
2085  FORMAT(5E16.8/5E16.8)
C
C     FORMATION OF MASS AND RESTORING CO-EFFICIENT MATRIX
C
      DO 22 I=1,6
      DO 22 J=1,6
      TMAS(I,J)=0.0
      REST(I,J)=0.0
   22 CONTINUE
      ZZCC=CH-DF
      ZMCC=SMAS*ZZCC
      TMAS(1,5)=ZMCC
      TMAS(2,4)=-ZMCC
      TMAS(4,2)=-ZMCC
      TMAS(5,1)=ZMCC
      TMAS(1,1)=SMAS
      TMAS(2,2)=SMAS
      TMAS(3,3)=SMAS
      TMAS(4,4)=TI44
      TMAS(4,6)=-TI46
      TMAS(5,5)=TI55
      TMAS(6,4)=-TI64
      TMAS(6,6)=TI66
      REST(3,3)=C33
      REST(3,5)=C35
      REST(5,3)=C35
      REST(4,4)=C44
      REST(5,5)=C55
      DO 16 I=1,NP
      DO 16 J=1,NP
      IF (I .EQ. J) GO TO 16
```

```
      RR=(X1(I)-X1(J))**2 + (X2(I)-X2(J))**2 + (X3(I)-X3(J))**2
      IF (RR .GT. .00001) GO TO 16
      WRITE(6,2019) I,J
   16 CONTINUE
 2019 FORMAT(2I10)
C
C     MAIN PROGRAM
C
      DO 1501 NN1=1,NWSTP
      DO 1503 NN3=1,NWVLN
      WVLN=WVLEN(NN3)
      WAMP=WVLN/WVSTP(NN1)
      WMP(NN3)=WAMP
C
C
      FACT1=.001
      FACT2=.1
      CONV1=.001
      CONV2=.0001
C
      AK=6.2831853/WVLN
      AK1=AK*DP
      AK2=EXP(AK1)
      AK3=EXP(-AK1)
      ANU=AK*(AK2-AK3)/(AK2+AK3)
      FR=SQRT(ANU*GRAV)
C
      CALL ROOT(NROOT,DP,ANU,AMU)
 2091 FORMAT(8E16.8)
C
C     CALCULATION FOR GREEN FUNCTION
C
      DO 10 I=1,NP
      DO 20 J=1,NP
      IF (I .EQ. J) GO TO 36
      XX1=X1(I)
      XX2=X2(I)
      XX3=X3(I)
      AA1=X1(J)
      AA2=X2(J)
      AA3=X3(J)
      AAN1=AN1(I)
      AAN2=AN2(I)
      AAN3=AN3(I)
      XA1=(X1(I)-X1(J))**2
      XA2=(X2(I)-X2(J))**2
      R=SQRT(XA1+XA2)
      SR1=SQRT(XA1+XA2+(X3(I)-X3(J))**2)
      SR2=SQRT(XA1+XA2+(X3(I)+2.*CH+X3(J))**2)
      AKR=AK*R
      IF (AKR .LE. .01) GO TO 30
C
C     EVALUATION OF GREEN FUNCTION USING SERIES FORM
C
      CALL GREEN1(NROOT,AMU,ANU,AK,DP,CH,R,XX1,XX2,XX3,AA1,AA2,AA3,AAN1,
     1AAN2,AAN3,GR,GIM,DGR,DGIM)
   35 CONTINUE
C
C
      G(I,J)=DCMPLX(GR,GIM)
```

```
      A(I,J)=DCMPLX(DGR,DGIM)
      G(I,J)=G(I,J)*S(J)/12.566371
      A(I,J)=A(I,J)*S(J)/6.2831853
      GO TO 400
   30 CONTINUE
C
C     EVALUATION OF GREEN FUNCTION BY INTEGRAL FORM
C
      CALL GREEN2(ANU,AK,DF,CH,R,XX1,XX2,XX3,AA1,AA2,AA3,SR1,SR2,AAN1,
     1AAN2,AAN3,FACT1,FACT2,CONV1,CONV2,GR,GIM,DGR,DGIM)
      GO TO 35
   36 G(I,J)=(0.0,0.0)
      A(I,J)=(-1.0,0.0)
      GGG1=0.5*SQRT(S(J)/3.14159265)
      GGG2=0.0
      G(I,J)=DCMPLX(GGG1,GGG2)
      G(I,J)=G(I,J)*SHFACT(I)
  400 CONTINUE
   20 CONTINUE
   10 CONTINUE
C
C     FORMATION OF VECTOR (B)
C
      K=1
      DUMM2=0.0
      DO 90 I=1,NP
      DUMM1=AN1(I)
   90 B(I,K)=DCMPLX(DUMM1,DUMM2)
      K=2
      DO 71 I=1,NP
      DUMM1=AN2(I)
   71 B(I,K)=DCMPLX(DUMM1,DUMM2)
      K=3
      DO 72 I=1,NP
      DUMM1=AN3(I)
   72 B(I,K)=DCMPLX(DUMM1,DUMM2)
      K=4
      DO 73 I=1,NP
      DUMM1=X2(I)*AN3(I) - X3(I)*AN2(I)
   73 B(I,K)=DCMPLX(DUMM1,DUMM2)
      K=5
      DO 74 I=1,NP
      DUMM1=X3(I)*AN1(I) - X1(I)*AN3(I)
   74 B(I,K)=DCMPLX(DUMM1,DUMM2)
      K=6
      DO 97 I=1,NP
      DUMM1=X1(I)*AN2(I) - X2(I)*AN1(I)
   97 B(I,K)=DCMPLX(DUMM1,DUMM2)
C
      DO 130 K=1,6
      DO 130 I=1,NP
      B(I,K)=2.*B(I,K)
  130 CONTINUE
C
C
C     INVERSION OF MATRIX (A)
C
      NN=NP
      N=NP
      CALL INVERT(A,NN,N,M,C,DET)
```

```
 2060 FORMAT(4(2E16.8))
C
C     DETERMING SOURCE STRENGTHS
C
      DO 52 K=1,6
      DO 53 J=1,NP
      SUM=(0.0,0.0)
      DO 54 I=1,NP
      SUM=SUM + A(J,I)*B(I,K)
   54 CONTINUE
      F(J,K)=SUM
   53 CONTINUE
   52 CONTINUE
C
C     CALCULATION OF POTENTIAL
C
      DO 65 K=1,6
      DO 66 I=1,NP
      SUM=(0.0,0.0)
      DO 67 J=I,NP
      SUM=SUM + G(I,J)*F(J,K)
   67 CONTINUE
      PHI(I,K)=SUM
   66 CONTINUE
   65 CONTINUE
C
C     CALCULATION FOR ADDED MASS AND DAMPING CO-EFFICIENTS
C
      DO 150 J=I,6
      DO 151 K=I,6
      SUM=(0.0,0.0)
      DO 152 I=1,NP
      GO TO (153,154,155,156,157,158),K
  153 DN=AN1(I)
      GO TO 160
  154 DN=AN2(I)
      GO TO 160
  155 DN=AN3(I)
      GO TO 160
  156 DN=X2(I)*AN3(I) - X3(I)*AN2(I)
      GO TO 160
  157 DN=X3(I)*AN1(I) - X1(I)*AN3(I)
      GO TO 160
  158 DN=X1(I)*AN2(I) - X2(I)*AN1(I)
  160 CONTINUE
      SUM=SUM + PHI(I,J)*S(I)*DN
  152 CONTINUE
      AX1=DREAL(SUM)
      AX3=DIMAG(SUM)
      ADMAS(K,J)=-RHO*AX1
      DAMP(K,J)=-RHO*FR*AX3
 2023 FORMAT(2I5,2E20.8)
  151 CONTINUE
  150 CONTINUE
C
C     ADM11 ETC ARE ADDED MASS, DMP11 ETC. ARE DAMP. COEFF.
C
      ADM11(NN3)=ADMAS(1,1)
      ADM22(NN3)=ADMAS(2,2)
      ADM33(NN3)=ADMAS(3,3)
```

```
      ADM44(NN3)=ADMAS(4,4)
      ADM55(NN3)=ADMAS(5,5)
      ADM66(NN3)=ADMAS(6,6)
      DMP11(NN3)=DAMP(1,1)
      DMP22(NN3)=DAMP(2,2)
      DMP33(NN3)=DAMP(3,3)
      DMP44(NN3)=DAMP(4,4)
      DMP55(NN3)=DAMP(5,5)
      DMP66(NN3)=DAMP(6,6)
C
      DO 1502 NN2=1,NHEAD
      ALPHA=ALFA(NN2)*3.1415924/180.0
C     CALCULATION OF DEFFRACTION POTENTIAL
C
      K=7
      DO 76 I=1,NP
      P1=AK*(X3(I) + CH)
      P2=AK*DP
      P3=EXP(P1)
      P4=EXP(-P1)
      P5=(COS(ALPHA)*AN1(I) + SIN(ALPHA)*AN2(I))*(P3+P4)*.5
      P6=(P3-P4)*AN3(I)*.5
      PP1=DCMPLX(P6,P5)
      P7=(X1(I)*COS(ALPHA) + X2(I)*SIN(ALPHA))*AK
      P8=COS(P7)
      P9=SIN(P7)
      PP2=DCMPLX(P8,P9)
      PO=2.*AK/(ANU*(EXP(P2)+EXP(-P2)))
      B(I,K)=-PO*PP1*PP2
   76 CONTINUE
C
      K=7
      DO 83 J=1,NP
      SUM=(0.0,0.0)
      DO 84 I=1,NP
      SUM=SUM + A(J,I)*B(I,K)
   84 CONTINUE
      F(J,K)=SUM
   83 CONTINUE
C
C
      K=7
      DO 86 I=1,NP
      SUM=(0.0,0.0)
      DO 87 J=1,NP
      SUM=SUM + G(I,J)*F(J,K)
   87 CONTINUE
      PHI(I,K)=SUM
   86 CONTINUE
C     CALCULATION OF INCIDENT WAVE POTENTIAL
C
      DO 350 I=1,NP
      P1=AK*(X3(I)+CH)
      P2=AK*DP
      P3=AK*(X1(I)*COS(ALPHA) + X2(I)*SIN(ALPHA))
      P4=(EXP(P1) + EXP(-P1))*.5
      P5=(EXP(P2) + EXP(-P2))*.5
      P6=P4/(P5*ANU)
      P7=COS(P3)
      P8=SIN(P3)
```

```
      TT=DCMPLX(P7,P8)
      POTIN(I)=P6*TT
  350 CONTINUE
C
C     CALCULATION OF EXCITING FORCE COMPLEX AMPLITUDE
C
      DO 360 K=1,6
      SUM=(0.0,0.0)
      DO 370 I=1,NP
      GO TO (371,372,373,374,375,376),K
  371 EN=AN1(I)
      GO TO 380
  372 EN=AN2(I)
      GO TO 380
  373 EN=AN3(I)
      GO TO 380
  374 EN=X2(I)*AN3(I) - X3(I)*AN2(I)
      GO TO 380
  375 EN=X3(I)*AN1(I) - X1(I)*AN3(I)
      GO TO 380
  376 EN=X1(I)*AN2(I) - X2(I)*AN1(I)
  380 CONTINUE
      SUM=SUM + (PHI(I,7) + POTIN(I))*EN*S(I)
  370 CONTINUE
      FEXT(K)=-RHO*FR*FR*WAMP*SUM
  360 CONTINUE
C
C     FXAM1 ETC ARE COMPLEX AMPLITUDE OF EXCITING FORCE
C
      FXAM1(NN2,NN3)=FEXT(1)
      FXAM2(NN2,NN3)=FEXT(2)
      FXAM3(NN2,NN3)=FEXT(3)
      FXAM4(NN2,NN3)=FEXT(4)
      FXAM5(NN2,NN3)=FEXT(5)
C     CALCULATION FOR MOTIONS , (SOLUTION OF EQUATIONS OF MOTION )
      FXAM6(NN2,NN3)=FEXT(6)
C
C
      DO 403 K=1,6
      DO 410 J=1,6
      T1=-FR*FR*(TMAS(K,J) + ADMAS(K,J))
      T2=-FR*DAMP(K,J)
      T3=REST(K,J)
      T4=T1+T3
      AA(K,J)=DCMPLX(T4,T2)
  410 CONTINUE
  403 CONTINUE
C
C     INVERSION OF COMPLEX MATRIX (A )
C
      N=6
      NN=6
      DE=(1.0E0,0.0E0)
  600 DO 610 I=1,NN
      M(I)=-I
  610 CONTINUE
      DO 620 I=I,NN
      X=0.0E0
      DO 630 L=I,NN
      IF(M(L) .GT. 0) GO TO 630
```

```
      DO 640 K=1,NN
      IF (M(K) .GT. 0) GO TO 640
      D=AA(L,K)
      Y=ABS(DREAL(D)) + ABS(DIMAG(D))
      IF (X .GT. Y) GO TO 640
      LD=L
      KD=K
      X=Y
  640 CONTINUE
  630 CONTINUE
      D=AA(LD,KD)
      DE=D
      L=-M(LD)
      M(LD)=M(KD)
      M(KD)=L
      DO 660 J=1,NN
      CC(J)=AA(LD,J)
      AA(LD,J)=AA(KD,J)
  660 AA(KD,J)=CC(J)
      DO 670 K=1,NN
      AA(K,KD)=AA(K,KD)/D
  670 CONTINUE
      DO 700 J=1,NN
      IF (J .EQ. KD) GO TO 700
      DO 710 K=1,NN
      AA(K,J)=AA(K,J)-CC(J)*AA(K,KD)
  710 CONTINUE
  700 CONTINUE
      CC(KD)=(-1.0E0,0.0E0)
      DO 780 K=1,NN
      AA(KD,K)=-CC(K)/D
  780 CONTINUE
  620 CONTINUE
      DO 840 I=1,NN
      L=0
  820 L=L+1
      IF (M(L) .NE. I) GO TO 820
      M(L)=M(I)
      M(I)=I
      DO 840 K=1,NN
      TEMP=AA(K,L)
      AA(K,L)=AA(K,I)
  840 AA(K,I)=TEMP
      DET=CDABS(DE)
  900 CONTINUE
C
      DO 1001 I=1,6
      SUM=(0.0,0.0)
      DO 1002 J=1,6
 1002 SUM=SUM+AA(I,J)*FEXT(J)
      ZAMP(I)=SUM
 1001 CONTINUE
C
C     ZMOT1 ETC. ARE COMPLEX MOTION AMPLITUDE
C
      ZMOT1(NN2,NN3)=ZAMP(1)
      ZMOT2(NN2,NN3)=ZAMP(2)
      ZMOT3(NN2,NN3)=ZAMP(3)
      ZMOT4(NN2,NN3)=ZAMP(4)
      ZMOT5(NN2,NN3)=ZAMP(5)
```

```
      ZMOT6(NN2,NN3)=ZAMP(6)
C
C     CALCULATION OF OSCILLATOTY HYDRODYNAMIC FORCE COMPLEX AMPLITUDE
C
      DO 525 K=1,6
      SSUM=(0.0,0.0)
      DO 526 J=1,6
      SUM=(0.0,0.0)
      DO 527 I=1,NP
      GO TO (528,529,530,531,532,533),K
  528 CN=AN1(I)
      GO TO 535
  529 CN=AN2(I)
      GO TO 535
  530 CN=AN3(I)
      GO TO 535
  531 CN=X2(I)*AN3(I) - X3(I)*AN2(I)
      GO TO 535
  532 CN=X3(I)*AN1(I) - X1(I)*AN3(I)
      GO TO 535
  533 CN=X1(I)*AN2(I) - X2(I)*AN1(I)
  535 CONTINUE
      SUM=SUM+PHI(I,J)*CN*S(I)
  527 CONTINUE
      SSUM=SSUM+SUM*ZAMP(J)
  526 CONTINUE
      FHYD(K)=-RHO*FR*FR*SSUM
  525 CONTINUE
C
C
C     FHAM1 ETC. ARE COMPLEX AMPLITUDE OF HYDO. FORCE
C
      FHAM1(NN2,NN3)=FHYD(1)
      FHAM2(NN2,NN3)=FHYD(2)
      FHAM3(NN2,NN3)=FHYD(3)
      FHAM4(NN2,NN3)=FHYD(4)
      FHAM5(NN2,NN3)=FHYD(5)
      FHAM6(NN2,NN3)=FHYD(6)
C
C     STORE FREQUENCY(NON-DIMENSIONAL) AND PERIOD
C
      WE(NN3)=FR*SQRT(ALLN/GRAV)
      TIME(NN3)=6.2831853/FR
C
 1502 CONTINUE
C
C     CLOSE LOOP FOR FREQUECCY
C
 1503 CONTINUE
C
C     DETERMING COMPLEX AMPLITUDE OF EXT FORCE AND PHASE
C
C     XAMP1 ETC. ARE AMPLITUDE OF EXCITING FORCE
C     XPH1 ETC. ARE PHASE
      DO 1573 J=1,NHEAD
C
      ALPHA1=ALFA(J)
      DO 1506 I=1,NWVLN
      XAMP1(I)=CDABS(FXAM1(J,I))
      XAMP2(I)=CDABS(FXAM2(J,I))
```

```
      XAMP3(I)=CDABS(FXAM3(J,I))
      XAMP4(I)=CDABS(FXAM4(J,I))
      XAMP5(I)=CDABS(FXAM5(J,I))
      XAMP6(I)=CDABS(FXAM6(J,I))
      IF(ABS(DREAL(FXAM1(J,I))) .LE. 0.0000001) GO TO 7261
      QQ1=DIMAG(FXAM1(J,I))
      QQ2=DREAL(FXAM1(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      XPH1(I)=QQ3
      GO TO 7262
 7261 XPH1(I)=90.0
      IF(DIMAG(FXAM1(J,I)) .LT. 0.0) XPH1(I)=-90.0
 7262 IF(ABS(DREAL(FXAM2(J,I))) .LE. 0.0000001) GO TO 7263
      QQ1=DIMAG(FXAM2(J,I))
      QQ2=DREAL(FXAM2(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      XPH2(I)=QQ3
      GO TO 7264
 7263 XPH2(I)=90.0
      IF(DIMAG(FXAM2(J,I)) .LT. 0.0) XPH2(I)=-90.0
 7264 IF(ABS(DREAL(FXAM3(J,I))) .LE. 0.0000001) GO TO 7265
      QQ1=DIMAG(FXAM3(J,I))
      QQ2=DREAL(FXAM3(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      XPH3(I)=QQ3
      GO TO 7266
 7265 XPH3(I)=90.0
      IF(DIMAG(FXAM3(J,I)) .LT. 0.0) XPH3(I)=-90.0
 7266 IF(ABS(DREAL(FXAM4(J,I))) .LE. 0.0000001) GO TO 7267
      QQ1=DIMAG(FXAM4(J,I))
      QQ2=DREAL(FXAM4(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      XPH4(I)=QQ3
      GO TO 7268
 7267 XPH4(I)=90.0
      IF(DIMAG(FXAM4(J,I)) .LT. 0.0) XPH4(I)=-90.0
 7268 IF(ABS(DREAL(FXAM5(J,I))) .LE. 0.0000001) GO TO 7269
      QQ1=DIMAG(FXAM5(J,I))
      QQ2=DREAL(FXAM5(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      XPH5(I)=QQ3
      GO TO 7270
 7269 XPH5(I)=90.0
      IF(DIMAG(FXAM5(J,I)) .LT. 0.0) XPH5(I)=-90.0
 7270 IF(ABS(DREAL(FXAM6(J,I))) .LE. 0.0000001) GO TO 7271
      QQ1=DIMAG(FXAM6(J,I))
      QQ2=DREAL(FXAM6(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      XPH6(I)=QQ3
      GO TO 1506
 7271 XPH6(I)=90.0
      IF(DIMAG(FXAM6(J,I)) .LT. 0.0) XPH6(I)=-90.0
C
C     CHANGE PHASE FROM RADIANS TO DEGREES
C
 1506 CONTINUE
C
C     ZMAM1 AND ZPH1 ETC. ARE MOTION AMPLITUDE AND PHASES
C
      DO 1507 I=1,NWVLN
```

```
      ZMAM1(I)=CDABS(ZMOT1(J,I))
      ZMAM2(I)=CDABS(ZMOT2(J,I))
      ZMAM3(I)=CDABS(ZMOT3(J,I))
      ZMAM4(I)=CDABS(ZMOT4(J,I))
      ZMAM5(I)=CDABS(ZMOT5(J,I))
      ZMAM6(I)=CDABS(ZMOT6(J,I))
      IF(ABS(DREAL(ZMOT1(J,I))) .LE. 0.0000001) GO TO 7241
      QQ1=DIMAG(ZMOT1(J,I))
      QQ2=DREAL(ZMOT1(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      ZPH1(I)=QQ3
      GO TO 7242
 7241 ZPH1(I)=90.0
      IF(DIMAG(ZMOT1(J,I)) .LT. 0.0) ZPH1(I)=-90.0
 7242 IF(ABS(DREAL(ZMOT2(J,I))) .LE. 0.0000001) GO TO 7243
      QQ1=DIMAG(ZMOT2(J,I))
      QQ2=DREAL(ZMOT2(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      ZPH2(I)=QQ3
      GO TO 7244
 7243 ZPH2(I)=90.0
      IF(DIMAG(ZMOT2(J,I)) .LT. 0.0) ZPH2(I)=-90.0
 7244 IF(ABS(DREAL(ZMOT3(J,I))) .LE. 0.0000001) GO TO 7245
      QQ1=DIMAG(ZMOT3(J,I))
      QQ2=DREAL(ZMOT3(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      ZPH3(I)=QQ3
      GO TO 7246
 7245 ZPH3(I)=90.0
      IF(DIMAG(ZMOT3(J,I)) .LT. 0.0) ZPH3(I)=-90.0
 7246 IF(ABS(DREAL(ZMOT4(J,I))) .LE. 0.0000001) GO TO 7247
      QQ1=DIMAG(ZMOT4(J,I))
      QQ2=DREAL(ZMOT4(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      ZPH4(I)=QQ3
      GO TO 7248
 7247 ZPH4(I)=90.0
      IF(DIMAG(ZMOT4(J,I)) .LT. 0.0) ZPH4(I)=-90.0
 7248 IF(ABS(DREAL(ZMOT5(J,I))) .LE. 0.0000001) GO TO 7249
      QQ1=DIMAG(ZMOT5(J,I))
      QQ2=DREAL(ZMOT5(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      ZPH5(I)=QQ3
      GO TO 7250
 7249 ZPH5(I)=90.0
      IF(DIMAG(ZMOT5(J,I)) .LT. 0.0) ZPH5(I)=-90.0
 7250 IF(ABS(DREAL(ZMOT6(J,I))) .LE. 0.0000001) GO TO 7251
      QQ1=DIMAG(ZMOT6(J,I))
      QQ2=DREAL(ZMOT6(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      ZPH6(I)=QQ3
      GO TO 1507
 7251 ZPH6(I)=90.0
      IF(DIMAG(ZMOT6(J,I)) .LT. 0.0) ZPH6(I)=-90.0
C     CHANGE PHASE FROM RADIANS TO DEGREES
C
 1507 CONTINUE
C
C     FAMP1 AND FPH1 ETC ARE HYDRO FORCE AMPLT. AND PHASES
C
```

```
      DO 1508 I=I,NWVLN
      FAMP1(I)=CDABS(FHAM1(J,I))
      FAMP2(I)=CDABS(FHAM2(J,I))
      FAMP3(I)=CDABS(FHAM3(J,I))
      FAMP4(I)=CDABS(FHAM4(J,I))
      FAMP5(I)=CDABS(FHAM5(J,I))
      FAMP6(I)=CDABS(FHAM6(J,I))
      IF(ABS(DREAL(FHAM1(J,I))) .LE. 0.0000001) GO TO 7221
      QQ1=DIMAG(FHAM1(J,I))
      QQ2=DREAL(FHAM1(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      FPH1(I)=QQ3
      GO TO 7222
 7221 FPH1(I)=90.0
      IF(DIMAG(FHAM1(J,I)) .LT. 0.0) FPH1(I)=-90.0
 7222 IF(ABS(DREAL(FHAM2(J,I))) .LE. 0.0000001) GO TO 7223
      QQ1=DIMAG(FHAM2(J,I))
      QQ2=DREAL(FHAM2(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      FPH2(I)=QQ3
      GO TO 7224
 7223 FPH2(I)=90.0
      IF(DIMAG(FHAM2(J,I)) .LT. 0.0) FPH2(I)=-90.0
 7224 IF(ABS(DREAL(FHAM3(J,I))) .LE. 0.0000001) GO TO 7225
      QQ1=DIMAG(FHAM3(J,I))
      QQ2=DREAL(FHAM3(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      FPH3(I)=QQ3
      GO TO 7226
 7225 FPH3(I)=90.0
      IF(DIMAG(FHAM3(J,I)) .LT. 0.0) FPH3(I)=-90.0
 7226 IF(ABS(DREAL(FHAM4(J,I))) .LE. 0.0000001) GO TO 7227
      QQ1=DIMAG(FHAM4(J,I))
      QQ2=DREAL(FHAM4(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      FPH4(I)=QQ3
      GO TO 7228
 7227 FPH4(I)=90.0
      IF(DIMAG(FHAM4(J,I)) .LT. 0.0) FPH4(I)=-90.0
 7228 IF(ABS(DREAL(FHAM5(J,I))) .LE. 0.0000001) GO TO 7229
      QQ1=DIMAG(FHAM5(J,I))
      QQ2=DREAL(FHAM5(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      FPH5(I)=QQ3
      GO TO 7230
 7229 FPH5(I)=90.0
      IF(DIMAG(FHAM5(J,I)) .LT. 0.0) FPH5(I)=-90.0
 7230 IF(ABS(DREAL(FHAM6(J,I))) .LE. 0.0000001) GO TO 7231
      QQ1=DIMAG(FHAM6(J,I))
      QQ2=DREAL(FHAM6(J,I))
      CALL SUB5(QQ1,QQ2,QQ3)
      FPH6(I)=QQ3
      GO TO 1508
 7231 FPH6(I)=90.0
      IF(DIMAG(FHAM6(J,I)) .LT. 0.0) FPH6(I)=-90.0
C
C     CHANGE PHASE FROM RADIANS TO DEGREES
C
 1508 CONTINUE
C     NON-DIMENSIONALISATION
```

```
C
      W1=RHO*VOLN
      W2=W1*ALLN*ALLN
      W3=W1*SQRT(GRAV/ALLN)
      W4=W3*ALLN*ALLN
C
      DO 1522 I=I,NWVLN
      WAMQ=WMP(I)
      W5=RHO*GRAV*VOLN*WAMQ/ALLN
      W6=RHO*GRAV*VOLN*WAMQ
      ADM11(I)=ADM11(I)/W1
      ADM22(I)=ADM22(I)/W1
      ADM33(I)=ADM33(I)/W1
      ADM44(I)=ADM44(I)/W2
      ADM55(I)=ADM55(I)/W2
      ADM66(I)=ADM66(I)/W2
      DMP11(I)=DMP11(I)/W3
      DMP22(I)=DMP22(I)/W3
      DMP33(I)=DMP33(I)/W3
      DMP44(I)=DMP44(I)/W4
      DMP55(I)=DMP55(I)/W4
      DMP66(I)=DMP66(I)/W4
      XAMP1(I)=XAMP1(I)/W5
      XAMP2(I)=XAMP2(I)/W5
      XAMP3(I)=XAMP3(I)/W5
      XAMP4(I)=XAMP4(I)/W6
      XAMP5(I)=XAMP5(I)/W6
      XAMP6(I)=XAMP6(I)/W6
      ZHAM1(I)=ZHAM1(I)/WAMQ
      ZHAM2(I)=ZHAM2(I)/WAMQ
      ZHAM3(I)=ZHAM3(I)/WAMQ
      ZHAM4(I)=ZHAM4(I)*ALLN/WAMQ
      ZHAM5(I)=ZHAM5(I)*ALLN/WAMQ
      ZHAM6(I)=ZHAM6(I)*ALLN/WAMQ
      FAMP1(I)=FAMP1(I)/W5
      FAMP2(I)=FAMP2(I)/W5
      FAMP3(I)=FAMP3(I)/W5
      FAMP4(I)=FAMP4(I)/W6
      FAMP5(I)=FAMP5(I)/W6
      FAMP6(I)=FAMP6(I)/W6
C
 1522 CONTINUE
C
C     PRINTING
C
      IF(J .GT. 1) GO TO 1579
      WRITE(6,2070)
      DO 1511 I=I,NWVLN
 1511 WRITE(6,2080) WE(I),TIME(I),ADM11(I),ADM22(I),ADM33(I),ADM44(I),AD
     1M55(I),ADM66(I)
      WRITE(6,2071)
      DO 1512 I=1,NWVLN
 1512 WRITE(6,2080) WE(I),TIME(I),DMP11(I),DMP22(I),DMP33(I),DMP44(I),DM
     1P55(I),DMP66(I)
 1579 CONTINUE
      WRITE(6,2072) ALPHAI
      DO 1513 I=I,NWVLN
 1513 WRITE(6,2081) WE(I),TIME(I),XAMP1(I),XPH1(I),XAMP2(I),XPH2(I),XAMP
     13(I),XPH3(I),XAMP4(I),XPH4(I),XAMP5(I),XPH5(I),XAMP6(I),XPH6(I)
      WRITE(6,2073) ALPHAI
```

```
      DO 1514 I=1,NWVLN
 1514 WRITE(6,2081) WE(I),TIME(I),ZMAM1(I),ZPH1(I),ZMAM2(I),ZPH2(I),ZMAM
     13(I),ZPH3(I),ZMAM4(I),ZPH4(I),ZMAM5(I),ZPH5(I),ZMAM6(I),ZPH6(I)
      WRITE(6,2074) ALPHA1
      DO 1515 I=1,NWVLN
 1515 WRITE(6,2081) WE(I),TIME(I),FAMP1(I),FPH1(I),FAMP2(I),FPH2(I),FAMP
     13(I),FPH3(I),FAMP4(I),FPH4(I),FAMP5(I),FPH5(I),FAMP6(I),FPH6(I)
 2080 FORMAT(4X,F6.3,5X,F5.2,6(5X,E10.4))
 2081 FORMAT(F6.3,1X,F5.2,6(1X,E10.4,1X,F6.1))
 2070 FORMAT(1H1,35X,47H- NON-DIMENSIONAL  ADDED  MASS  CO-EFFICIENTS -/
     1/60X,///7X,2HWE,7X,4HTIME,8X,6HA(1,1),
     29X,6HA(2,2),9X,6HA(3,3),9X,6HA(4,4),9X,6HA(5,5),9X,6HA(6,6)/)
 2071 FORMAT(1H1,35X,43H- NON-DIMENSIONAL  DAMPING  CO-EFFICIENTS -/
     1/60X,///7X,2HWE,7X,4HTIME,8X,6HB(1,1),
     29X,6HB(2,2),9X,6HB(3,3),9X,6HB(4,4),9X,6HB(5,5),9X,6HB(6,6)/)
 2072 FORMAT(1H1,35X,50H- NON-DIMENSIONAL  EXCITING  FORCE  AND  MOMENTS
     1 -//60X,10HHEADING = F7.2,8H DEGREES///3X,2HWE,3X,4HTIME,4X,11HSUR
     2GE FORCE,7X,10HSWAY FORCE,8X,11HHEAVE FORCE,7X,11HROLL MOMENT,7X,
     312HPITCH MOMENT,7X,10HYAW MOMENT/13X,11HAMPL. RATIO,1X,5HPHASE,1X,
     411HAMPL..RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL.
     5RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X
     6,5HPHASE/25X,5H(DEG),13X,5H(DEG),13X,5H(DEG),13X,5H(DEG),13X,5H(DE
     7G),13X,5H(DEG)/)
 2073 FORMAT(1H1,35X,50H - NON-DIMENSIONAL MOTION AMPLITUDES AND PHASES
     1 -//60X,10HHEADING = F7.2,8H DEGREES///3X,2HWE,3X,4HTIME,4X,11HSUR
     2GE MOTN.,7X,10HSWAY MOTN.,8X,11HHEAVE MOTN.,7X,11HROLL MOTION,7X,
     312HPITCH MOTION,7X,10HYAW MOTION/13X,11HAMPL. RATIO,1X,5HPHASE,1X,
     411HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL.
     5RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X
     6,5HPHASE/25X,5H(DEG),13X,5H(DEG),13X,5H(DEG),13X,5H(DEG),13X,5H(DE
     7G),13X,5H(DEG)/)
 2074 FORMAT(1H1,35X,50H- NON DIMENSIONAL OSCILLATORY HYDRODYNAMIC FORCE
     1 -//60X,10HHEADING = F7.2,8H DEGREES///3X,2HWE,3X,4HTIME,4X,11HSUR
     2GE FORCE,7X,10HSWAY FORCE,8X,11HHEAVE FORCE,7X,11HROLL MOMENT,7X,
     312HPITCH MOMENT,7X,10HYAW MOMENT/13X,11HAMPL. RATIO,1X,5HPHASE,1X,
     411HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL.
     5RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X,5HPHASE,1X,11HAMPL. RATIO,1X
     6,5HPHASE/25X,5H(DEG),13X,5H(DEG),13X,5H(DEG),13X,5H(DEG),13X,5H(DE
     7G),13X,5H(DEG)/)
C     CLOSE LOOP FOR HEADING
C
 1573 CONTINUE
C
C     CLOSE LOOP FOR WAVE STEEPNESS
C
 1501 CONTINUE
C
      STOP
      END
      SUBROUTINE ROOT(NROOT,DP,ANU,AMU)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AMU(100)
C
      TAN(X)=DTAN(X)
      ABS(X)=DABS(X)
      COS(X)=DCOS(X)
      DO 50 K=1,NROOT
      MM=1
      XX1=(K-.5)*3.14159265
      XLIMIT=XX1/DP
```

```
      XINT=3.14159265/80.0
      XXN=XX1+XINT
    5 X1=XXN/DP
      F1=X1*TAN(XXN)+ANU
      IF (F1) 6,7,8
    6 XXN=XXN+XINT
      GO TO 5
    8 XX=X1*DP
      DF1=TAN(XX) + XX/(COS(XX)**2)
      X2=X1-F1/DF1
      IF (X2 .GT. XLIMIT) GO TO 9
      XINT=XINT/2.
      XXNEW=XXN-XINT
   13 XNEW=XXNEW/DP
      FNEW=XNEW*TAN(XXNEW) + ANU
      IF (FNEW) 10,11,12
   10 XINT=XINT/2.
      XXNEW=XXNEW+XINT
      GO TO 13
   12 X1=XNEW
      F1=FNEW
      XXN=XXNEW
      GO TO 8
    9 F2=X2*TAN(X2*DP) + ANU
      MM=MM+1
      IF(MM .GT. 100) GO TO 14
      IF (ABS(X1-X2)-.000001) 14,14,15
   15 X1=X2
      F1=F2
      GO TO 8
    7 XROOT=X1
      FN=F1
      GO TO 16
   11 XROOT=XNEW
      FN=FNEW
      GO TO 16
   14 XROOT=X2
      FN=F2
   16 CONTINUE
      AMU(K)=XROOT
   50 CONTINUE
      RETURN
      END
C
C     SUBROUTINE FOR EVALUATION OF GREEN FUNCTION AND ITS DERIVATIVES BY THE SE
C    *IES FORM
C
      SUBROUTINE GREEN1(NROOT,AMU,ANU,AK,DP,CH,K,XX1,XX2,XX3,AA1,AA2,
     1AA3,AAN1,AAN2,AAN3,GR,GIM,DGR,DGIM)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AMU(100)
C
      ABS(X)=DABS(X)
      EXP(X)=DEXP(X)
      COS(X)=DCOS(X)
      SIN(X)=DSIN(X)
C
      C=CH
      XXAA1=XX1-AA1
      XXAA2=XX2-AA2
```

```
      T1=AK*(AA3+C)
      T2=(EXP(T1) + EXP(-T1))/2.
      T3=6.2831853*(ANU*ANU-AK*AK)/(AK*AK*DP-ANU*ANU*DP+ANU)
      A=T3*T2
      X=AK*R
      N=0
      D=.000001
      CALL BESJ(X,N,BJ,D,IER)
      BJ0=BJ
      CALL BESY(X,N,BY,IER)
      BY0=BY
      T4=AK*(XX3+C)
      T5=(EXP(T4) + EXP(-T4))/2.
      P1=A*T5*BY0
      P2=-A*T5*BJ0
      N=1
      CALL BESJ(X,N,BJ,D,IER)
      BJ1=BJ
 2047 FORMAT(3E20.8)
      CALL BESY(X,N,BY,IER)
      BY1=BY
      P3=-A*T5*XXAA1*BY1*AK/R
      P4=A*T5*XXAA1*AK*BJ1/R
      P5=-A*T5*BY1*AK*XXAA2/R
      P6=A*T5*BJ1*AK*XXAA2/R
      T6=(EXP(T4)-EXP(-T4))/2.
      P7=A*AK*T6*BY0
      P8=-A*AK*T6*BJ0
      SUM1=0.
      SUM3=0.
      SUM5=0.
      SUM7=0.
      K1=0
      K3=0
      K5=0
      K7=0
      I=1
    9 B=4*(AMU(I)**2 + ANU**2)/(DP*AMU(I)**2 + DP*ANU**2 - ANU)*
     1COS(AMU(I)*(AA3+C))
      X=AMU(I)*R
      N=0
      CALL BESK(X,N,BK,IER)
      BK0=BK
      N=1
      CALL BESK(X,N,BK,IER)
      BK1=BK
      BB=B*COS(AMU(I)*(XX3+C))
      S1=BB*BK0
      IF (K1 .EQ. 1) GO TO 2
      SS1=ABS(S1)
      SSUM1=ABS(SUM1)
      IF (SS1 .LE. (.000001*SSUM1)) K1=1
      INDEX=1
      SUM1=SUM1+S1
    2 S3=BB*AMU(I)*XXAA1*BK1/R
      IF (K3 .EQ. 1) GO TO 3
      SS3=ABS(S3)
      SSUM3=ABS(SUM3)
      IF (SS3 .LE. (.000001*SSUM3)) K3=1
      INDEX=2
```

```
      SUM3=SUM3+S3
    3 S5=BB*AMU(I)*XXAA2*BK1/R
      IF (K5 .EQ. 1) GO TO 4
      SS5=ABS(S5)
      SSUM5=ABS(SUM5)
      IF (SS5 .LE. (.000001*SSUM5)) K5=1
      INDEX=3
      SUM5=SUM5+S5
    4 S7=B*AMU(I)*SIN(AMU(I)*(XX3+C))*BK0
      IF (K7 .EQ. 1) GO TO 5
      SS7=ABS(S7)
      SSUM7=ABS(SUM7)
      IF (SS7 .LE. (.000001*SSUM7)) K7=1
      INDEX=4
      SUM7=SUM7+S7
    5 IF (K1 .NE. 1) GO TO 6
      IF (K3 .NE. 1) GO TO 6
      IF (K5 .NE. 1) GO TO 6
      IF (K7 .NE. 1) GO TO 6
    7 SUM3=-SUM3
      SUM5=-SUM5
      SUM7=-SUM7
      GO TO 8
    6 I=I+1
      IF (I .GT. NROOT) GO TO 7
      GO TO 9
    8 GR=P1+SUM1
      GIM=P2
      B3=P3+SUM3
      B5=P5+SUM5
      B7=P7+SUM7
      DGR=B3*AAN1 + B5*AAN2 + B7*AAN3
      DGIM=P4*AAN1 + P6*AAN2 + P8*AAN3
      RETURN
      END
      SUBROUTINE GREEN2(AMU,AK,DP,CH,R,XX1,XX2,XX3,AA1,AA2,AA3,SR1,SR2,
     1AAN1,AAN2,AAN3,FACT1,FACT2,CONV1,CONV2,GR,GIM,DGR,DGIM)
C
      IMPLICIT REAL*8 (A-H,O-Z)
C     EVALUATION OF GREEN FUNCTION USING THE INTEGRAL FORM
C
C
      EXP(X)=DEXP(X)
      ABS(X)=DABS(X)
      C=CH
      XXAA1=XX1-AA1
      XXAA2=XX2-AA2
      XXAA3=XX3+2.*C+AA3
C
      EPS1=FACT1*AK
      Z=AK
      CALL SUB1(Z,C,DP,AMU,XX1,XX2,XX3,AA1,AA2,AA3,R,F1MU,F2MU,F3MU,
     1F4MU)
      F1K=F1MU
      F2K=F2MU
      F3K=F3MU
      F4K=F4MU
C
      INDEX=1
      RLIMIT=AK-EPS1
```

```
      SLEN=AK-EPS1
      ZINIT=0.0
      ITER=1
   35 CONTINUE
      NITER=0
      SINT=.4/DP
      NUM=SLEN/SINT
      SNUM=NUM+1
   12 CONTINUE
      STEP=SLEN/SNUM
      NUMORD=SNUM+1
      SUM1=0.0
      SUM2=0.0
      SUM3=0.0
      SUM4=0.0
      NN=1
      Z=ZINIT
    5 CONTINUE
C
      CALL SUB1(Z,C,DP,ANU,XX1,XX2,XX3,AA1,AA2,AA3,K,F1MU,F2MU,F3MU,
     1F4MU)
C
      F1Z=F1MU
      F2Z=F2MU
      F3Z=F3MU
      F4Z=F4MU
C
      P1=Z*DP
      P2=EXP(P1)
      P3=EXP(-P1)
      P4=(P2-P3)/(P2+P3)
      UU=Z*P4-ANU
C
      OR1=(F1Z-F1K)/UU
      OR2=(F2Z-F2K)/UU
      OR3=(F3Z-F3K)/UU
      OR4=(F4Z-F4K)/UU
C
      IF(NN .NE. 1) GO TO 6
      ORS1=OR1
      ORS2=OR2
      ORS3=OR3
      ORS4=OR4
      Z=Z+STEP
      NN=NN+1
      GO TO 5
    6 CONTINUE
      ORE1=OR1
      ORE2=OR2
      ORE3=OR3
      ORE4=OR4
C
      SSUM1=.5*STEP*(ORS1+ORE1)
      SSUM2=.5*STEP*(ORS2+ORE2)
      SSUM3=.5*STEP*(ORS3+ORE3)
      SSUM4=.5*STEP*(ORS4+ORE4)
      SUM1=SUM1+SSUM1
      SUM2=SUM2+SSUM2
      SUM3=SUM3+SSUM3
      SUM4=SUM4+SSUM4
```

```
      Z=Z+STEP
      NN=NN+1
      IF(NN .GT. NUMORD) GO TO 8
      ORS1=ORE1
      ORS2=ORE2
      ORS3=ORE3
      ORS4=ORE4
      GO TO 5
C
    8 CONTINUE
C
      IF (ITER .NE. 1) GO TO 13
      SSA1=SUM1
      SSA2=SUM2
      SSA3=SUM3
      SSA4=SUM4
   20 CONTINUE
      ITER=2
      SNUM=2.*SNUM
      GO TO 12
   13 CONTINUE
      NITER=NITER+1
      IF (NITER .GT. 6) GO TO 909
      SSB1=SUM1
      SSB2=SUM2
      SSB3=SUM3
      SSB4=SUM4
 3001 FORMAT(I5,4E20.8)
C
      IF (ABS(SSB1) .LE. .0000001) GO TO 901
      SCON1=ABS((SSB1-SSA1)/SSB1)
      GO TO 902
  901 SCON1=0.0
  902 CONTINUE
      IF (ABS(SSB2) .LE. .0000001) GO TO 903
      SCON2=ABS((SSB2-SSA2)/SSB2)
      GO TO 904
  903 SCON2=0.0
  904 CONTINUE
      IF (ABS(SSB3) .LE. .0000001) GO TO 905
      SCON3=ABS((SSB3-SSA3)/SSB3)
      GO TO 906
  905 SCON3=0.0
  906 CONTINUE
      IF (ABS(SSB4) .LE. 0.0000001) GO TO 907
      SCON4=ABS((SSB4-SSA4)/SSB4)
      GO TO 908
  907 SCON4=0.0
  908 CONTINUE
       IF (SCON1 .GT. CONV1) GO TO 15
       IF (SCON2 .GT. CONV1) GO TO 15
       IF (SCON3 .GT. CONV1) GO TO 15
       IF (SCON4 .GT. CONV1) GO TO 15
C
  909 CONTINUE
      IF (INDEX .NE. 1) GO TO 27
C
      SS11=SSB1
      SS21=SSB2
      SS31=SSB3
```

```
      SS41=SSB4
      GO TO 25
   15 CONTINUE
C
      SSA1=SSB1
      SSA2=SSB2
      SSA3=SSB3
      SSA4=SSB4
      GO TO 20
   25 CONTINUE
C
      INDEX=2
      ZINIT=AK+EPS1
      ZLIMIT=2.*AK
      ITER=1
      GO TO 35
   27 CONTINUE
C
      SS12=SSB1
      SS22=SSB2
      SS32=SSB3
      SS42=SSB4
C
      EPS2=FACT2*AK
C
      INDEX=1
      ZLIMIT=AK-EPS2
      SLEN=AK-EPS2
      ZINIT=0.0
      ITER=1
   75 CONTINUE
      NITER=0
      SINT=0.4/DP
      NUM=SLEN/SINT
      SNUM=NUM+1
   52 CONTINUE
      STEP=SLEN/SNUM
      NUMORD=SNUM+1
C
      SUM1=0.0
      SUM2=0.0
      SUM3=0.0
      SUM4=0.0
      NN=1
      Z=ZINIT
C
   45 CONTINUE
      P1=Z*DP
      P2=EXP(P1)
      P3=EXP(-P1)
      P4=(P2-P3)/(P2+P3)
      UU=Z*P4-ANU
      OR1=1.0/UU
      IF (NN .NE. 1) GO TO 40
      ORS1=OR1
      NN=NN+1
      Z=Z+STEP
      GO TO 45
   40 CONTINUE
      ORE1=OR1
```

```
      SSUM1=0.5*STEP*(ORE1+ORS1)
      SUM1=SUM1+SSUM1
      Z=Z+STEP
      NN=NN+1
      IF (NN .GT. NUMORD) GO TO 48
      ORS1=ORE1
      GO TO 45
   48 CONTINUE
      IF (ITER .NE. 1) GO TO 53
      SSA1=SUM1
   60 CONTINUE
      ITER=2
      SNUM=2.*SNUM
      GO TO 52
   53 CONTINUE
      NITER=NITER+1
      IF (NITER .GT. 6) GO TO 54
      SSB1=SUM1
C
      SCON1=ABS((SSB1-SSA1)/SSB1)
      IF (SCON1 .GT. CONV1) GO TO 55
   54 CONTINUE
      IF (INDEX .NE. 1) GO TO 67
      SSA3=SSB1
      GO TO 65
   55 CONTINUE
      SSA1=SSB1
      GO TO 60
   65 CONTINUE
      INDEX=2
      ZINIT=AK+EPS2
      ITER=1
      ZLIMIT=2.*AK
      GO TO 75
   67 CONTINUE
      SSA4=SSB1
C
      SS13=F1K*SSA3
      SS23=F2K*SSA3
      SS33=F3K*SSA3
      SS43=F4K*SSA3
C
      SS14=F1K*SSA4
      SS24=F2K*SSA4
      SS34=F3K*SSA4
      SS44=F4K*SSA4
 2055 FORMAT(2E20.8)
C
      P1=AK*DP
      P2=EXP(P1)
      P3=EXP(-P1)
      P4=(P2+P3)*.5
      P5=1.0/P4
      P6=(P2-P3)/(P2+P3)
      P7=P5**2
      P8=P7*(1-P1*P6)*(2.0*EPS2)
      P9=(P6+P1*P7)**2
      SSC5=-P8/P9
      SS15=F1K*SSC5
```

```
      SS25=F2K*SSC5
      SS35=F3K*SSC5
      SS45=F4K*SSC5
C
      STMIN=0.4/DP
      STMAX=100000.
      IF (R .GT. .000001) STMAX=0.3/R
C
      SUM1=0.0
      SUM2=0.0
      SUM3=0.0
      SUM4=0.0
      NN=1
      STEP=STMIN
      Z=2.*AK
   81 CONTINUE
      CALL SUB2(Z,C,DP,ANU,XX1,XX2,XX3,AA1,AA2,AA3,R,FN1,FN2,FN3,FN4)
      F1Z=FN1
      F2Z=FN2
      F3Z=FN3
      F4Z=FN4
      IF (NN .NE. 1) GO TO 83
      ORS1=F1Z
      ORS2=F2Z
      ORS3=F3Z
      ORS4=F4Z
      NN=NN+1
      Z=Z+STEP
      GO TO 81
   83 ORE1=F1Z
      ORE2=F2Z
      ORE3=F3Z
      ORE4=F4Z
C
      SSUM1=0.5*STEP*(ORS1 + ORE1)
      SSUM2=0.5*STEP*(ORS2 + ORE2)
      SSUM3=0.5*STEP*(ORS3 + ORE3)
      SSUM4=0.5*STEP*(ORS4 + ORE4)
C
      SUM1=SUM1 + SSUM1
      SUM2=SUM2 + SSUM2
      SUM3=SUM3 + SSUM3
      SUM4=SUM4 + SSUM4
C
      IF (ABS(SUM1) .LE. 0.000001) GO TO 910
      SS1=ABS(SSUM1/SUM1)
      GO TO 911
  910 SS1=0.0
  911 CONTINUE
      IF (ABS(SUM2) .LE. 0.000001) GO TO 912
      SS2=ABS(SSUM2/SUM2)
      GO TO 913
  912 SS2=0.0
  913 CONTINUE
      IF (ABS(SUM3) .LE. 0.000001) GO TO 914
      SS3=ABS(SSUM3/SUM3)
      GO TO 915
  914 SS3=0.0
  915 CONTINUE
      IF (ABS(SUM4) .LE. 0.000001) GO TO 916
```

```
      SS4=ABS(SSUM4/SUM4)
      GO TO 917
  916 SS4=0.0
  917 CONTINUE
      IF (SS1 .GT. CONV2) GO TO 888
      IF (SS2 .GT. CONV2) GO TO 888
      IF (SS3 .GT. CONV2) GO TO 888
      IF (SS4 .GT. CONV2) GO TO 888
C
      GO TO 85
  888 CONTINUE
C
      NN=NN+1
      ORS1=ORE1
      ORS2=ORE2
      ORS3=ORE3
      ORS4=ORE4
C
      STEP=0.1*Z
      IF (STEP .LE. STMIN) STEP=STMIN
      IF (STEP .GT. STMAX) STEP=STMAX
      Z=Z+STEP
      ZZDP=Z*DP
      IF (ZZDP .GT. 80) GO TO 85
      GO TO 81
   85 CONTINUE
C
      SS16=SUM1
      SS26=SUM2
      SS36=SUM3
      SS46=SUM4
C
      SS17=(1.0/SR1) + (1.0/SR2)
      EE=SR1**3
      FF=SR2**3
      SS27=-XXAA1/EE - XXAA1/FF
      SS37=-XXAA2/EE - XXAA2/FF
      XXAA4=XX3-AA3
      SS47=-XXAA4/EE - XXAA3/FF
 2054 FORMAT(4E20.8)
C
      Q1=SS11 + SS12 + SS13 + SS14 + SS15 + SS16 + SS17
      Q2=SS21 + SS22 + SS23 + SS24 + SS25 + SS26 + SS27
      Q3=SS31 + SS32 + SS33 + SS34 + SS35 + SS36 + SS37
      Q4=SS41 + SS42 + SS43 + SS44 + SS45 + SS46 + SS47
C
      GR=Q1
      DGR=Q2*AAN1 + Q3*AAN2 + Q4*AAN3
C
      V1=AK*(AA3+C)
      V2=(EXP(V1) + EXP(-V1))*0.5
      V3=AK*AK - ANU*ANU
      V4=AK*AK*DP - ANU*ANU*DP + ANU
      TT=6.2831853*V3*V2/V4
      V5=AK*(XX3+C)
      V6=EXP(V5)
      V7=EXP(-V5)
      X=AK*X
      N=0
      D=0.000001
```

```
      IF (X .LE. 0.000001) GO TO 101
      CALL BESJ(X,N,BJ,D,IER)
      BJ0=BJ
      N=2
      CALL BESJ(X,N,BJ,D,IER)
      BJ2=BJ
      GO TO 102
  101 BJ0=1.0
      BJ2=0.0
  102 CONTINUE
      T1=TT*(V6+V7)*0.5*BJ0
      T2=-0.25*TT*(V6+V7)*AK*AK*XXAA1*(BJ0+BJ2)
      T3=-0.25*TT*(V6+V7)*AK*AK*XXAA2*(BJ0+BJ2)
      T4=TT*AK*(V6-V7)*0.5*BJ0
C
      GIM=T1
      DGIM=T2*AAN1 + T3*AAN2 + T4*AAN3
C
 3000 FORMAT(4E20.8)
      RETURN
      END
      SUBROUTINE SUB1(Z,C,DP,ANU,XX1,XX2,XX3,AA1,AA2,AA3,R,F1MU,F2MU,
     1F3MU,F4MU)
      IMPLICIT REAL*8 (A-H,O-Z)
      EXP(X)=DEXP(X)
      XXAA1=XX1-AA1
      XXAA2=XX2-AA2
C
      P1=Z*DP
      P2=Z*(AA3+C)
      P3=Z*(XX3+C)
C
      X=Z*R
      N=0
      D=0.000001
      IF (X .LE. 0.000001) GO TO 5
      CALL BESJ(X,N,BJ,D,IER)
      BJ0=BJ
      GO TO 6
    5 BJ0=1.0
    6 CONTINUE
      N=1
      IF (X .LE. 0.000001) GO TO 8
      CALL BESJ(X,N,BJ,D,IER)
      BJ1=BJ
      GO TO 9
    8 BJ1=0.0
    9 CONTINUE
C
      T1=2.*(Z+ANU)
      T2=EXP(-P1)
      T3=(EXP(P2) + EXP(-P2))*0.5
      T4=(EXP(P1) + T2)*0.5
      A=T1*T2*T3/T4
      P4=EXP(P3)
      P5=EXP(-P3)
      T5=(P4+P5)*0.5
      T6=(P4-P5)*0.5
      F1MU=A*T5*BJ0
      F4MU=Z*A*T6*BJO
```

```
      IF (X .LE. 0.000001) GO TO 10
      F2MU=-A*T5*Z*XXAA1*BJ1/R
      F3MU=-A*T5*Z*XXAA2*BJ1/R
      GO TO 15
   10 F2MU=-A*T5*Z*Z*XXAA1*0.5
      F3MU=-A*T5*Z*Z*XXAA2*0.5
   15 CONTINUE
C
      RETURN
      END
      SUBROUTINE SUB2(Z,C,DP,ANU,XX1,XX2,XX3,AA1,AA2,AA3,R,FN1,FN2,FN3,
     1FN4)
      IMPLICIT REAL*8 (A-H,O-Z)
C
      EXP(X)=DEXP(X)
      XXAA1=XX1-AA1
      XXAA2=XX2-AA2
      P1=Z*DP
      P2=Z*(AA3+C)
      P3=Z*(XX3+C)
C
      N=0
      D=0.000001
      X=Z*R
      IF (X .LE. 0.000001) GO TO 2
      CALL BESJ(X,N,BJ,D,IER)
      BJ0=BJ
      GO TO 3
    2 BJ0=1.0
    3 CONTINUE
      N=1
      IF (X .LE. 0.000001) GO TO 4
      CALL BESJ(X,N,BJ,D,IER)
      BJ1=BJ
      GO TO 5
    4 BJ1=0.0
    5 CONTINUE
C
      P4=EXP(-P1)
      P5=(EXP(P2) + EXP(-P2))*0.5
      T1=EXP(P1)
      T2=P4
      P6=(T1-T2)*0.5
      P7=(T1+T2)*0.5
      A=(2.*(Z+ANU)*P4*P5)/(Z*P6-ANU*P7)
      T3=EXP(P3)
      T4=EXP(-P3)
      T5=(T3+T4)*0.5
      T6=(T3-T4)*0.5
C
      FN1=A*T5*BJ0
      FN4=Z*A*T6*BJ0
      IF (X .LE. 0.000001) GO TO 7
      FN2=-A*T5*Z*XXAA1*BJ1/R
      FN3=-A*T5*Z*XXAA2*BJ1/R
      GO TO 10
    7 CONTINUE
      FN2=-A*T5*Z*Z*XXAA1*0.5
      FN3=-A*T5*Z*Z*XXAA2*0.5
   10 CONTINUE
```

```
C
      RETURN
      END
      SUBROUTINE INVERT(A,NN,N,M,C,DET)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION M(250)
      COMPLEX*16 A(250,250),C(250),D,TEMP,DE
      ABS(X)=DABS(X)
      DE=(1.0E0,0.0E0)
      IF (NN-1) 300,350,100
  100 DO 110 I=1,NN
      M(I)=-I
  110 CONTINUE
      DO 200 I=1,NN
      X=0.0E0
      DO 130 L=1,NN
      IF (M(L) .GT. 0) GO TO 130
      DO 120 K=1,NN
      IF (M(K) .GT. 0) GO TO 120
      D=A(L,K)
      Y=ABS(DREAL(D)) + ABS(DIMAG(D))
      IF (X .GT. Y) GO TO 120
      LD=L
      KD=K
      X=Y
  120 CONTINUE
  130 CONTINUE
      D=A(LD,KD)
      DE=D
      L=-M(LD)
      M(LD)=M(KD)
      M(KD)=L
      DO 140 J=1,NN
      C(J)=A(LD,J)
      A(LD,J)=A(KD,J)
  140 A(KD,J)=C(J)
      DO 150 K=1,NN
      A(K,KD)=A(K,KD)/D
  150 CONTINUE
      DO 170 J=1,NN
      IF (J .EQ. KD) GO TO 170
      DO 160 K=1,NN
      A(K,J)=A(K,J)-C(J)*A(K,KD)
  160 CONTINUE
  170 CONTINUE
      C(KD)=(-1.0E0,0.0E0)
      DO 180 K=1,NN
      A(KD,K)=-C(K)/D
  180 CONTINUE
  200 CONTINUE
      DO 240 I=1,NN
      L=0
  220 L=L+1
      IF (M(L) .NE. I) GO TO 220
      M(L)=M(I)
      M(I)=I
      DO 240 K=1,NN
      TEMP=A(K,L)
      A(K,L)=A(K,I)
  240 A(K,I)=TEMP
```

```
      DET=CDABS(DE)
  300 RETURN
  350 A(I,1)=1.0E0/A(I,1)
      DET=CDABS(A(I,1))
      GO TO 300
      END
      SUBROUTINE BESJ(X,N,BJ,D,IER)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*4 X44
      ABS(X)=DABS(X)
      FLOAT(I)=DFLOAT(I)
      ALOG(X)=DLOG(X)
C
      X44=X
      BJ=0.0
      IF(N)10,20,20
   10 IER=1
      RETURN
   20 IF(X)30,30,31
   30 IER=2
      RETURN
   31 IF(X-15.)32,32,34
   32 NTEST=20.+10.*X-X**2/3
      GO TO 36
   34 NTEST=90.+X/2.
   36 IF(N-NTEST)40,38,38
   38 IER=4
      RETURN
   40 IER=0
      N1=N+1
      BPREV=0.0
C
C     COMPUTE STARTING VALUE OF M
C
      IF(X-5.)50,60,60
   50 MA=X+6.
      GO TO 70
   60 MA=1.4*X+60./X
   70 MB=N+IFIX(X44)/4+2
      MZERO=MAX0(MA,MB)
C
C     SET UPPER LIMIT OF M
C
      MMAX=NTEST
  100 DO 190 M=MZERO,MMAX,3
C
C     SET F(M),F(M-1)
C
      FM1=1.0E-28
      FM=0.0
      ALPHA=0.0
      IF(M-(M/2)*2)120,110,120
  110 JT=-1
      GO TO 130
  120 JT=1
  130 M2=M-2
      DO 160 K=1,M2
      MK=M-K
      BMK=2.*FLOAT(MK)*FM1/X-FM
      FM=FM1
```

```
      FM1=BMK
      IF(MK-N-1)150,140,150
  140 BJ=BMK
  150 JT=-JT
      S=1+JT
  160 ALPHA=ALPHA+BMK*S
      BMK=2.*FM1/X-FM
      IF(N)180,170,180
  170 BJ=BMK
  180 ALPHA=ALPHA+BMK
      BJ=BJ/ALPHA
      IF(ABS(BJ-BPREV)-ABS(D*BJ))200,200,190
  190 BPREV=BJ
      IER=3
  200 RETURN
      END
      SUBROUTINE BESK(X,N,BK,IER)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION T(12)
      EXP(X)=DEXP(X)
      SQRT(X)=DSQRT(X)
      FLOAT(I)=DFLOAT(I)
      ALOG(X)=DLOG(X)
      BK=0.0
      IF(N)10,11,11
   10 IER=1
      RETURN
   11 IF(X)12,12,20
   12 IER=2
      RETURN
   20 IF(X-170.0)22,22,21
   21 IER=3
      RETURN
   22 IER=0
      IF(X-1.)36,36,25
   25 A=EXP(-X)
      B=1./X
      C=SQRT(B)
      T(1)=B
      DO 26 L=2,12
   26 T(L)=T(L-1)*B
      IF(N-1)27,29,27
C
C     COMPUTE KO USING POLYNOMIAL APPROXIMATION
C
   27 GO=A*(1.25331414-.15666418*T(1)+.088111278*T(2)-.091390954*T(3)
     2+.13445962*T(4)-.22998503*T(5)+.37924097*T(6)-.52472773*T(7)
     3+.55753684*T(8)-.42626329*T(9)+.21845181*T(10)-.066809767*T(11)
     4+.009189383*T(12))*C
      IF(N)20,28,29
   28 BK=GO
      RETURN
C
C     COMPUTE K1 USING PLYNOMIAL APPROXIMATION
C
   29 G1=A*(1.2533141+.46999270*T(1)-.14685830*T(2)+.12804266*T(3)
     2-.17364316*T(4)+.28476181*T(5)-.45943421*T(6)+.62833807*T(7)
     3-.66322954*T(8)+.50502386*T(9)-.25813038*T(10)+.078800012*T(11)
     4-.010824177*T(12))*C
      IF(N-1)20,30,31
```

```
   30 BK=G1
      RETURN
C
C     FROM K0,K1 COMPUTE KN USING RECURRENCE RELATION
C
   31 DO 35 J=2,N
      GJ=2.*(FLOAT(J)-1.)*G1/X+G0
      IF(GJ-1.0E38)33,33,32
   32 IER=4
      GO TO 34
   33 G0=G1
   35 G1=GJ
   34 BK=GJ
      RETURN
   36 B=X/2.
      A=.57721566+ALOG(B)
      C=B*B
      IF(N-1)37,43,37
C
C     COMPUTE K0 USING SERIES EXPANSION
C
   37 G0=-A
      X2J=1.
      FACT=1.
      HJ=0.0
      DO 40 J=1,6
      RJ=1./FLOAT(J)
      X2J=X2J*C
      FACT=FACT*RJ*RJ
      HJ=HJ+RJ
   40 G0=G0+X2J*FACT*(HJ-A)
      IF(N)43,42,43
   42 BK=G0
      RETURN
C
C     COMPUTE K1 USING SERIES EXPANSION
C
   43 X2J=B
      FACT=1.
      HJ=1.
      G1=1./X+X2J*(.5+A-HJ)
      DO 50 J=2,8
      X2J=X2J*C
      RJ=1./FLOAT(J)
      FACT=FACT*RJ*RJ
      HJ=HJ+RJ
   50 G1=G1+X2J*FACT*(.5+(A-HJ)*FLOAT(J))
      IF(N-1)31,52,31
   52 BK=G1
      RETURN
      END
      SUBROUTINE BESY(X,N,BY,IER)
      IMPLICIT REAL*8 (A-H,O-Z)
      SQRT(X)=DSQRT(X)
      SIN(X)=DSIN(X)
      COS(X)=DCOS(X)
      FLOAT(I)=DFLOAT(I)
      ALOG(X)=DLOG(X)
      ABS(X)=DABS(X)
C
```

```
C       CHECK FOR ERRORS IN N AND X
C
        IF(N)180,10,10
   10   IER=0
        IF(X)190,190,20
C
C       BRANCH IF X LESS THAN OR EQUAL 4
C
   20   IF(X-4.0)40,40,30
C
C       COMPUTE Y0 AND Y1 FOR X GREATER THAN 4
C
   30   T1=4.0/X
        T2=T1*T1
        P0=((((-.0000037043*T2+.0000173565)*T2-.0000487613)*T2
      1   +.00017343)*T2-.001753062)*T2+.3989423
        Q0=((((.0000032312*T2-.0000142078)*T2+.0000342468)*T2
      1   -.0000869791)*T2+.0004564324)*T2-.01246694
        P1=((((.0000042414*T2-.0000200920)*T2+.0000580759)*T2
      1   -.000223203)*T2+.002921826)*T2+.3989423
        Q1=((((-.0000036594*T2+.00001622)*T2-.0000398708)*T2
      1   +.0001064741)*T2-.0006390400)*T2+.03740084
        A=2.0/SQRT(X)
        B=A*T1
        C=X-.7853982
        Y0=A*P0*SIN(C)+B*Q0*COS(C)
        Y1=-A*P1*COS(C)+B*Q1*SIN(C)
        GO TO 90
C
C       COMPUTE Y0 AND Y1 FOR X LESS THAN OR EQUAL TO 4
C
   40   XX=X/2.
        X2=XX*XX
        T=ALOG(XX)+.5772157
        SUM=0.0
        TERM=T
        Y0=T
        DO 70 L=1,15
        IF(L-1)50,60,50
   50   SUM=SUM+1./FLOAT(L-1)
   60   FL=L
        TS=T-SUM
        TERM=(TERM*(-X2)/FL**2)*(1.-1./(FL*TS))
   70   Y0=Y0+TERM
        TERM=  XX*(T-.5)
        SUM=0.0
        Y1=TERM
        DO 80 L=2,16
        SUM=SUM+1./FLOAT(L-1)
        FL=L
        FL1=FL-1.
        TS=T-SUM
        TERM=(TERM*(-X2)/(FL1*FL))*((TS-.5/FL)/(TS+.5/FL1))
   80   Y1=Y1+TERM
        PI2=.6366198
        Y0=PI2*Y0
        Y1=-PI2/X+PI2*Y1
C
C
C
```

```
   90 IF(N-1)100,100,130
C
C
  100 IF(N)110,120,110
  110 BY=Y1
      GO TO 170
  120 BY=Y0
      GO TO 170
C
C     PERFORM RECURRENCE OPERATIONS TO FIND YN(X)
C
  130 YA=Y0
      YB=Y1
      K=1
  140 T=FLOAT(2*K)/X
      YC=T*YB-YA
      IF(ABS(YC)-1.0E38)145,145,141
  141 IER=3
      RETURN
  145 K=K+1
      IF(K-N)150,160,150
  150 YA=YB
      YB=YC
      GO TO 140
  160 BY=YC
  170 RETURN
  180 IER=1
      RETURN
  190 IER=2
      RETURN
      END
      SUBROUTINE SUBS(QQ1,QQ2,QQ3)
      IMPLICIT REAL*8 (A-H,O-Z)
      IF(QQ1 .GT. 0 .AND. QQ2 .LE. 0) GO TO 2
      IF(QQ1 .LE. 0 .AND. QQ2 .LE. 0) GO TO 3
      QQ3=DATAN(QQ1/QQ2)
      GO TO 5
    2 QQ3=3.141592654-DATAN(-QQ1/QQ2)
      GO TO 5
    3 QQ3=-3.141592654+DATAN(QQ1/QQ2)
    5 QQ3=QQ3*57.29578
      RETURN
      END
```

APPENDIX B

TYPICAL INPUT DATA

```
1
122
-42,-33.75,-15.245,-.6,-.6,0,46.75,.9930
-36,-37.5,-15.245,0,-1,0,39,.9940
-30,-37.5,-15.245,0,-1,0,39,.9940
-24,-37.5,-15.245,0,-1,0,39,.9940
-18,-37.5,-15.245,0,-1,0,39,.9940
-12,-37.5,-15.245,0,-1,0,39,.9940
 -6,-37.5,-15.245,0,-1,0,39,.9940
  0,-37.5,-15.245,0,-1,0,39,.9940
  6,-37.5,-15.245,0,-1,0,39,.9940
 12,-37.5,-15.245,0,-1,0,39,.9940
 18,-37.5,-15.245,0,-1,0,39,.9940
 24,-37.5,-15.245,0,-1,0,39,.9940
 30,-37.5,-15.245,0,-1,0,39,.9940
 36,-37.5,-15.245,0,-1,0,39,.9940
 42,-33.75,-15.245,0.6,-.6,0,46.75,.9930
-42,-26.75,-15.245,-.6,.6,0,46.75,.9930
-36,-22.5,-15.245,0,1,0,39,.9940
-30,-22.5,-15.245,0,1,0,39,.9940
-24,-22.5,-15.245,0,1,0,39,.9940
-18,-22.5,-15.245,0,1,0,39,.9940
-12,-22.5,-15.245,0,1,0,39,.9940
 -6,-22.5,-15.245,0,1,0,39,.9940
  0,-22.5,-15.245,0,1,0,39,.9940
  6,-22.5,-15.245,0,1,0,39,.9940
 12,-22.5,-15.245,0,1,0,39,.9940
 18,-22.5,-15.245,0,1,0,39,.9940
 24,-22.5,-15.245,0,1,0,39,.9940
 30,-22.5,-15.245,0,1,0,39,.9940
 36,-22.5,-15.245,0,1,0,39,.9940
 42,-26.25,-15.245,.6,.6,0,46.75,.9930
-45,-30,-15.245,-1,0,0,39,.9940 *
 45,-30,-15.245, 1,0,0,39,.9940
-41.5714,-32.7857,-18.495,0,0,-1,31.5,.975667
-36,-33.75,-18.495,0,0,-1,45,.9906
-30,-33.75,-18.495,0,0,-1,45,.9906
-24,-33.75,-18.495,0,0,-1,45,.9906
-18,-33.75,-18.495,0,0,-1,45,.9906
-12,-33.75,-18.495,0,0,-1,45,.9906
 -6,-33.75,-18.495,0,0,-1,45,.9906
  0,-33.75,-18.495,0,0,-1,45,.9906
  6,-33.75,-18.495,0,0,-1,45,.9906
 12,-33.75,-18.495,0,0,-1,45,.9906
 18,-33.75,-18.495,0,0,-1,45,.9906
 24,-33.75,-18.495,0,0,-1,45,.9906
 30,-33.75,-18.495,0,0,-1,45,.9906
 36,-33.75,-18.495,0,0,-1,45,.9906
41.5714,-27.2143,-18.495,0,0,-1,31.5,.975617
-41.5714,-27.2143,-18.495,0,0,-1,31.50,.975667
-36,-26.25,-18.495,0,0,-1,45,.9906
-30,-26.25,-18.495,0,0,-1,45,.9906
-24,-26.25,-18.495,0,0,-1,45,.9906
-18,-26.25,-18.495,0,0,-1,45,.9906
-12,-26.25,-18.495,0,0,-1,45,.9906
 -6,-26.25,-18.495,0,0,-1,45,.9906
  0,-26.25,-18.495,0,0,-1,45,.9906
  6,-26.25,-18.495,0,0,-1,45,.9906
```

```
12,-26.25,-18.495,0,0,-1,45,.9986
18,-26.25,-18.495,0,0,-1,45,.9986
24,-26.25,-18.495,0,0,-1,45,.9986
30,-26.25,-18.495,0,0,-1,45,.9986
36,-26.25,-18.495,0,0,-1,45,.9986
41.5714,-27.2143,-18.495,0,0,-1,31.50,.975817
-42.3706,-31.1263,-11.995,0,0,1,12.9690,.921625
-40.2483,-33.9857,-11.995,0,0,1,25.7160,.971628
-36.3241,-35.9894,-11.995,0,0,1,11.6550,.982686
-31.5089,-35.8331,-11.995,0,0,1,20.9625,.967001
-26.6676,-32.9917,-11.995,0,0,1,20.9730,.954075
-42.3706,-28.8737,-11.995,0,0,1,12.9690,.921625
-40.2483,-26.0143,-11.995,0,0,1,25.7160,.971628
-36.3241,-24.0106,-11.995,0,0,1,11.6550,.982686
-31.5089,-24.1669,-11.995,0,0,1,20.9625,.967001
-28.6676,-27.0083,-11.995,0,0,1,20.9730,.954075
-23.8125,-33.7500,-11.995,0,0,1,47.8125,.992500
-17.4375,-33.7500,-11.995,0,0,1,47.8125,.992500
-23.8125,-26.2500,-11.995,0,0,1,47.8125,.992500
-17.4375,-26.2500,-11.995,0,0,1,47.8125,.992500
-11.50,-35.5,-11.995,0,0,1,22.0,.9870
-11.50,-23.5,-11.995,0,0,1,22.0,.9870
-13.1716,-32.2892,-11.995,0,0,1,5.8438,.961873
-9.8284,-32.2892,-11.995,0,0,1,5.8438,.965062
-13.1716,-27.7108,-11.995,0,0,1,5.8438,.961873
-9.8284,-27.7108,-11.995,0,0,1,5.8438,.965062
-4.3750,-33.7500,-11.995,0,0,1,65.635,.992700
-4.3750,-26.2500,-11.995,0,0,1,65.625,.992700
4.375,-26.25,-11.995,0,0,1,65.625,.992700
4.375,-33.75,-11.995,0,0,1,65.625,.9927
9.8284,-27.7108,-11.995,0,0,1,5.8438,.965062
13.1716,-27.7108,-11.995,0,0,1,5.8438,.961873
9.8284,-32.2892,-11.995,0,0,1,5.8438,.965062
13.1716,-32.2892,-11.995,0,0,1,5.8438,.961873
[illegible]50,-23.5,-11.995,0,0,1,22.0,.9870
11.50,-35.5,-11.995,0,0,1,22.0,.9870
17.4375,-26.25,-11.995,0,0,1,47.8125,.9925
23.8125,-26.25,-11.995,0,0,1,47.8125,.9925
17.4375,-33.75,-11.995,0,0,1,47.8125,.9925
23.8125,-33.75,-11.995,0,0,1,47.8125,.9925
28.6676,-27.0083,-11.995,0,0,1,20.9730,.954075
31.5089,-24.1669,-11.995,0,0,1,20.9625,.967001
36.3241,-24.0106,-11.995,0,0,1,11.6550,.982686
40.2483,-26.0143,-11.995,0,0,1,25.7160,.971628
42.3706,-28.8737,-11.995,0,0,1,12.9690,.921625
28.6676,-32.9917,-11.995,0,0,1,20.9730,.954075
31.5089,-35.8331,-11.995,0,0,1,20.9625,.967001
36.3241,-35.9894,-11.995,0,0,1,11.6550,.982686
40.2483,-33.9857,-11.995,0,0,1,25.7160,.971628
42.3706,-31.1263,-11.995,0,0,1,12.9690,.921625
-38.50,-30.0,-5.9975,-1,0,0,95.96,.9824
-30.50,-30.0,-5.9975, 1,0,0,95.96,.9824
-34.50,-34.0,-5.9975,0,-1,0,95.96,.9824
-34.50,-26.0,-5.9975,0, 1,0,95.96,.9824
-14.0,-30.0,-5.9975,-1,0,0,59.975,.9406
 -9.0,-30.0,-5.9975, 1,0,0,59.975,.9406
-11.5,-32.5,-5.9975,0,-1,0,59.975,.9406
-11.5,-27.5,-5.9975,0, 1,0,59.975,.9406
```

```
 11.5,-27.5,-5.9975,0, 1,0,59.975,.9406
 11.5,-32.5,-5.9975,0,-1,0,59.975,.9406
  9.0,-30.0,-5.9975,-1,0,0,59.975,.9406
 14.0,-30.0,-5.9975, 1,0,0,59.975,.9406
 34.5,-26.0,-5.9975,0, 1,0,95.96,.9824
 34.5,-34.0,-5.9975,0,-1,0,95.96,.9824
 30.50,-30.0,-5.9975,-1,0,0,95.96,.9844
 38.50,-30.0,-5.9975, 1,0,0,95.96,.9844
20869,19054603,0,17411925,0
28449914,3510,0,535546,545894
60,6,2,1
100,200,300,400,550,700
0,45
100
150,140.000,9.8,1.025,20300,90
```