CHALMERS

Practical Source Code for Ship Motions Time Domain Numerical Analysis and Its Mobile Device Application

Master of Science Thesis in the Master Degree Programme, Naval Architecture

ZAYAR THEIN

Department of Shipping and Marine Technology
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2011
Report No. X-11/273

Practical Source Code for
Ship Motions Time Domain Numerical Analysis and Its Mobile Device Application

Zayar Thein

Department of Shipping and Marine Technology
CHALMERS UNIVERISTY OF TECHNOLOGY
Göteborg, Sweden 2011

Practical Source Code for Ship Motions Time Domain Numerical Analysis and Its Mobile Device Application

Zayar Thein

Report No X-11/273
Department of Shipping and Marine Technology
CHALMERS UNIVERSITY OF TECHNOLOGY
SE-412 96 Göteborg
Sweden

Abstract

This thesis included developing MATLAB program source code for the six degrees of rigid body freedom time domain motions of a ship travelling with a specified mean velocity in waves. The code is developed to simulate canonical forced and free motions. This study followed the impulse-response function time domain method in which the hydrodynamic coefficients are computed using the 2-D linear strip method in the frequency domain and converted into the time domain. The equations of motion are then solved simultaneously.

This study considers merchant ships except for ships with a buttock flow stern. Ikeda's prediction is used for roll damping computation, and a good solution is achieved for vessels other than buttock flow stern ships in roll motion. A range of wave periods and speeds are also considered. Steady-state results are compared to other published results. The numerical method agrees with classical ship motions and published results.

The present work explores ways to simulate ship motion in mobile devices. Two concepts are proposed: web-based simulation (WBS) and a stand-alone simulation. The latter approach is adopted. Two parts of the source codes are rewritten for the iOS platform: the motion related to memory forces and the calculation engine. The codes are tested in the iOS simulator, and the results are compared with the MATLAB solution. The world's first mobile device application in naval architecture achieves a good simulation result.

Acknowledgements

This work was made possible by the support of many people. First, the project could never have been completed without the remarkable support, effort, and motivation of Professor Carl-Erik Janson, my supervisor at Chalmers University of Technology, and Martin Schreuder, the examiner of my thesis. To them, I owe an enormous debt of gratitude.

The previous work of Benjamin Gros and Luis Felipe Sánchez Heres on the frequency domain part and Erik Ovegård's numerical simulation of parametric rolling served as an invaluable guide, which undoubtedly helped me to avoid wrong turns and poor decisions in my quest to solve many problems. Special thanks to Professor Lars Bergdahl, who stimulated my interest in seakeeping. My mentor, Professor Alaa Mansour inspired me to become a naval architect. I offer him my sincere gratitude.

I thank Lotta Olsson, the Studieadminstratör, for her kind assistance throughout my study at Chalmers. Thank you also to Dr. Ken Worthy for his tireless reviewing of my drafts. Finally, my family, friends, and co-workers at Herbert-ABS also supported me throughout the project.

Table of contents

Abbreviations

a_{ij}: Sectional added mass

a_w: Wave amplitude

a_{2n-1}: Mapping coefficient

A: Added mass matrix

A_s: Cross section area

b_{ij}: Sectional damping coefficient

B_{ij}: Inviscid damping matrix

B: Beam at waterline

B^F: Hull skin friction damping coefficient

B^E: Hull eddy shedding damping coefficient

B^w: Free surface wave damping coefficient

B^l: Lift force damping coefficient

C_{ij}: Stiffness coefficients matrix

C_b : Block coefficient

C_w : Wave celerity

d: Water depth

f_i^I : Sectional Froude-Kryov force for the i-th motion

f_i^D : Sectional Diffraction force for the i-th motion

F : Force vector

F_e : Exciting forces vector

F_n: Froude number

Fb : Freeboard

g: Earth's gravity

GMt, GMl: Transverse and longitudinal metacentric height

H_0: Half the beam draft ratio.

I_4: Roll moment of inertia

I_5: Pitch moment of inertia

j: Index of six degrees of freedom

k: Wave number

K: Retardation Function

k_{xx}: Roll radius of gyration

k_{yy}: Pitch radius of gyration

l: Euclidian length

LCG: Longitudinal centre of gravity

LOA: Length over all

LWL: Length of waterline

$\hat{n}$: Unit vector

N_ω : Number of angular frequency steps

N_τ : Number of time steps

M_s: Conformal mapping scale factor

OG: Vertical distance between the water line and the center of gravity, OG=D-KG

P: Pressure

$\hat{r}$: Position Vector

R_e: Effective bilge radius

U: Vessel speed

VCG, KG: Vertical centre of gravity

VCB: Vertical centre of buoyancy

v_o: Threshold velocity

V: Displaced Volume

∇ : Displacement

ξ: Wave elevation

ω: Angular frequency

Ω: Angular frequency Maximum value in lieu of infinity in computation

$\Delta\tau$: Time step

Γ: Maximum time limit in lieu of infinity in computation

ω_e: Encounter frequency

β: Heading angle

ε: Phase angle

λ: Wave length

η_i: i-th wave motion

η_c: Complex amplitude of the wave motion

v: Kinematic viscosity of the water

ϕ: Velocity potential

σ_s: Sectional area coefficient

ρ: Density of water

1. Introduction

1.1. Overview

The study of wave-induced loads and motions of ships is important both in ship design and operational studies. To assess these loads and motions, model test, full-scale trials, or numerical calculations may be used. Although full-scale trials provide the most realistic results, they may not work well for testing extreme weather conditions, which are infrequent. Model tests also create difficulties in scaling the test results when viscous hydrodynamic forces are considered. [4] On the other hand, the growing computational capacity of modern computers and the ease of carrying out simulated tests allow numerical computations to play an increasingly important role. More advanced and easy to use computer aided engineering software products are needed for naval architecture work, including analysis of ship motions.

Based on different approaches to solutions, the numerical technique in ship motion analysis can be categorized as frequency domain and time domain methods. The latter approach is still under development at industries and institutions, including at the Shipping and Marine Technology Department of Chalmers University of Technology. For research and educational purposes, computational source codes are written in MATLAB, a high-level technical computing language, at Charmers University. On the other hand, the current shift taking place in the computer industry from personal computers to mobile devices demands touch-screen based computer programs and mobile device applications. Unlike other computations, the numerical time domain method requires longer computing time and more computer resources.

The purpose of this study is to develop a MATLAB source code based on the time domain method, which can be used to analyze ship motion and to investigate whether the method can be implemented in current mobile devices.

Section 2 of this document formulates the mathematical descriptions for the equations of motion in the time domain as well as the dynamic balance of inertial, hydrostatic (i.e. gravitational), and hydrodynamic forces acting upon a ship's hull. It also describes the numerical implementation of equations of motion and methods to analyze errors generated by numerical calculations.

Section 3 presents the MATLAB source code development as well as an application for mobile devices.

Section 4 illustrates results for two hull forms with zero speed. The conventional ship S-175 and a box-shaped hull are examined. Forced periodic motion results are compared to available published results.

Section 5 discusses the contributions of this work and its practical applications. It highlights the advantages and disadvantages of this program as well as its limitations. It also highlights important issues for future research.

1.2. Project Description

In addition to time domain numerical computation, the ship motion simulation source code that the present work seeks to develop requires knowledge of the seakeeping properties of

the vessel under study. Moreover, the program must be capable of reading ship geometry files so that the study can be done on different ship models. Furthermore, the program should include a graphical user interface for easy entering and editing of data and viewing of results. These requirements are listed in Table 1 as a program development sequence.

Table 1: Program Development Sequence

Sequence	Description
Entering Data	To be able to upload ship geometry files, enter computational data, and select user options, a MATLAB graphical user interface will be developed.
Analyzing the Ship Model	Time domain computation requires ship model data. The program will calculate ship hull model data and hydrostatic properties from the corresponding geometry file.
Computing Coefficients	Hydrodynamics coefficients are required in computing motion equations and are based on ship geometry, wave frequency, and ship speed. These coefficients will be computed before solving the motion equation numerically.
Computing Forces	Linear and non-linear forces required by the time-domain method will be solved numerically.
Solving Motion Equation	Numerical method will be developed and the motion equations will be solved numerically in all rigid body degrees of freedom.
Viewing the Results	The same MATLAB user interface will be developed so that the user will be able to easily access the result of the time domain numerical computation.
Migrating to a Mobile	The MATLAB source code will be rewritten in Xcodes for the Apple Mobile iOS platform.

To simplify reading the geometry file and solving the motion equation, the program will be limited to analyzing motion on ships. Offshore structures, such as semi-submersible, tension leg platforms, and Jack-up drilling rig, will not be handled by the program. In addition, the hull should be a mono hull, and no roll tank will be considered.

1.3. Ship Motions

Although a ship operates on a two-dimensional sea surface, due to wave induced loads, six rigid body degrees of freedom—surge, sway, heave, roll, pitch, and yaw—about a reference frame (oxyz), fixed to the steady motion of the ship, are considered for the motion of a ship, as shown in figure 1. The vessel floats freely and travels at constant forward speed *U* and rotation, β, with respect to an inertial frame which will be described in section 3. These six

oscillation motions at a given time, ship geometry, and speed, and wave definition are determined in a ship motion study.

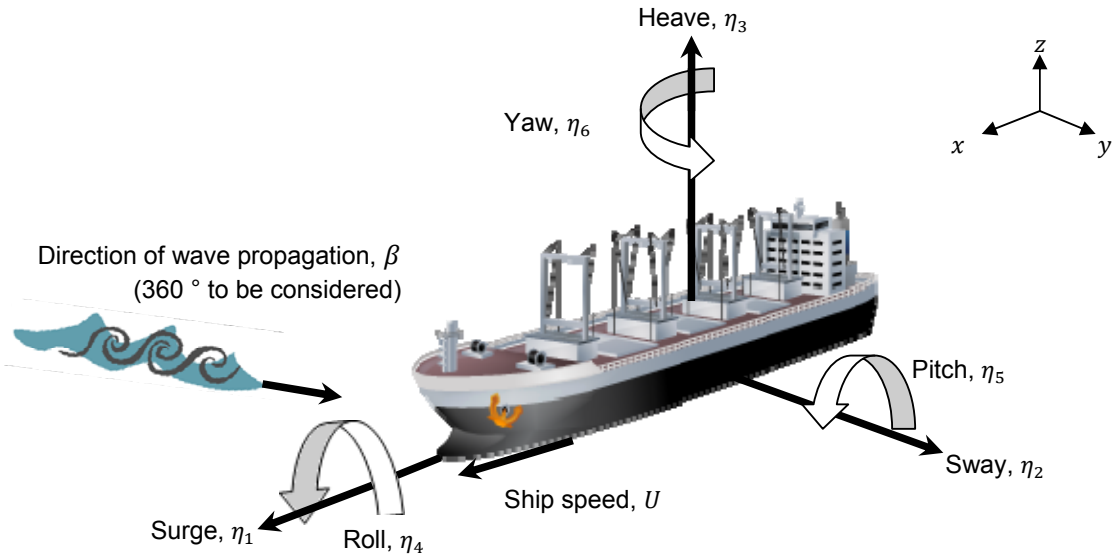


Figure 1: Definitions of coordinate system and rigid-body motion modes. *U* is the forward speed for the ship. Wave propagation direction may be varied.

To compute the response of a vessel to waves, the equations of motion are solved. In the simple case of a monochromatic incident wave, the equations might appear as

$$\sum_{j=1}^{6}[(M_{ij} + A_{ij})\ddot{\eta}_j + B_{ij}\dot{\eta}_j + C_{ij}\eta_j] = \sum F_i \quad \textit{for i=1,…,6} \qquad \ldots (1)$$

where, η is the position of the body, M_{ij} are the elements of the generalized mass matrix, A_{ij} are the elements of the hydrodynamic mass or added mass matrix, B_{ij} are the hydrodynamic damping or radiation damping coefficients, C_{ij} are the hydrostatic stiffness or restoring coefficients, and $\sum F_i$ represents the resultant of all other forces in the *i* direction (i.e. those in addition to the wave exciting force and hydrodynamic reaction forces) that may be present. A dot indicates a derivative with respect to time. Once the above quantities have been found, the motion of the ship in waves can be predicted by solving the equations of motion.

When solving the equations of motion, one may use either the frequency domain method or the time domain method. In the frequency domain, the ship's motions are treated as low amplitude sinusoidal motions. The calculation is fast since the computation is done in one step for a particular frequency response. It can produce usable results. However, the method should not be used if the Froude number is too high and the vessel of interest is not sufficiently slender (i.e. the ratio between the breadth and draft is larger than 6). [2] On the other hand, the time domain method is popular when very large ships are considered. [7] The downside is that the method itself is complex and more computationally intensive. It requires many thousands of small incremental time steps in computations. Figure 2 illustrates the relationship between a time domain solution of waves and a frequency domain representation of the waves by a wave spectrum.

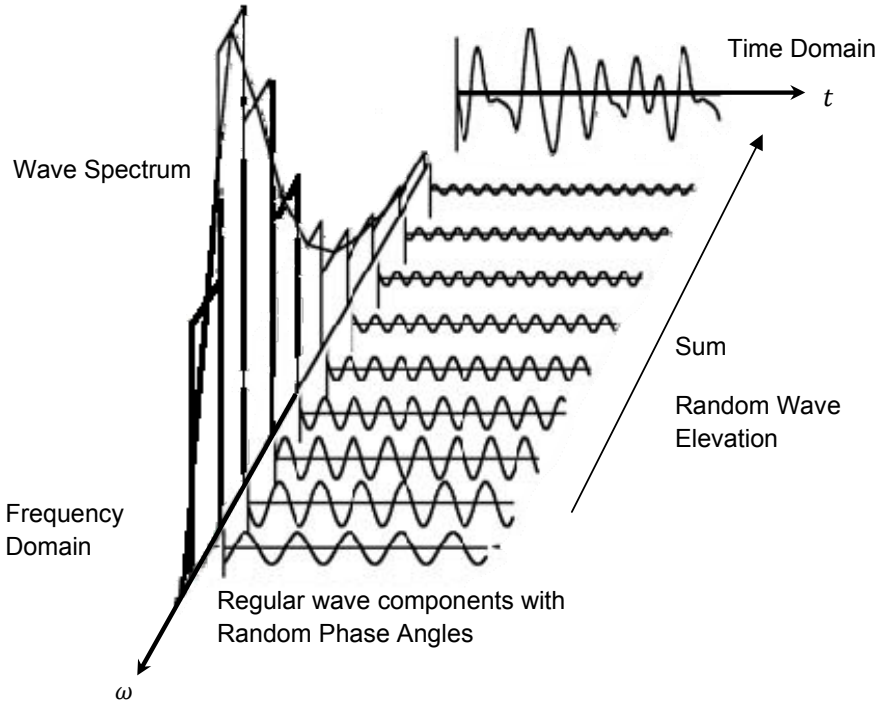


Figure 2: The relation between a frequency domain and time domain solutions.

1.4. Summary

The purpose of this work, then, is to develop open source MATLAB code for the time domain numerical method for ship motion analysis. First, ways to compute ship hydrostatic properties, hydrodynamic coefficients, and required forces in the equations of motion will be sought. Then, the code to solve the equations will be developed. The equations are solved and verified with other results predicated by other institutions. Finally, the mobile device application will be developed based on MATLAB codes, and the application will be tested to determine whether it is feasible for seakeeping study.

2. Background Theory

2.1. Overview

This section establishes the theoretical background of the source code developed in this project and highlights the solution methods used to achieve reliable results. The section has three parts. The first part is concerned with setting up the equations of motion for the time domain method. The second part outlines each element of the equations. The last part provides ways to solve the equations numerically.

2.2. Equations of Motion in Time Domain

When establishing the ship motion equations, Newton's Second Law of Motion may be used as a basic. In general, a ship's equations of motion might appear as

$$(M)\ddot{\eta} = \sum F \quad \ldots (2)$$

where, *M* is the mass of the ship, $\ddot{\eta}$ is the acceleration of the ship, and $\sum F$ on the right hand side of the equation is the sum of all applied forces from waves which causes the response of the untethered ship as described on the left hand side. The task now is to identify the forces on the ship.

The wave forces on the ship may be categorized into two parts: viscous forces from drag and inertia forces, as indicated by potential flow wave theory. In this seakeeping project, inertial forces are more important than drag forces and will only be considered in the force equilibrium described in equation 2. The inertia forces can be identified by considering a spectrum of waves interacting with the ship. One of the components of the spectrum consists of waves that will diffract upon the ship hull and scatter in all directions. The other is the waves that set the ship into motion. In the latter case, the motion of the ship will generate additional waves moving out in all directions. [17] Thus, forces arising from the potential flow wave theory may be written as

$$F = \iint P\,\hat{n}\,ds \quad \ldots (3)$$

$$P = -\rho\left(\frac{\partial \phi}{\partial t} + gy + \frac{1}{2}|\nabla\phi|^2\right) \quad \ldots (4)$$

where *P* is the pressure, ϕ is the velocity potential. To address linear theory, the velocity potential can be decomposed into a sum of three velocity potentials,

$$\phi = \phi^I + \phi^R + \phi^D \quad \ldots (5)$$

where ϕ^I is incident wave potential, ϕ^R is radiated wave potential, and the last component is diffracted wave potential. The Froude-Krylov force, introduced by the unsteady pressure field generated by undisturbed waves, and the restoring force can be categorized under incident wave potential. The other forces will then be radiation forces and diffraction forces.

When the ship moves, it creates waves because of forces related to added mass (inertia added to the motion due to an accelerating or decelerating ship hull must move some volume of surrounding water) and the damping coefficient and hence they are radiation forces. The

damping can be subdivided into two parts: inviscid damping or memory forces and viscous roll damping. [12] Figure 3 shows the composition of all considered forces from waves.

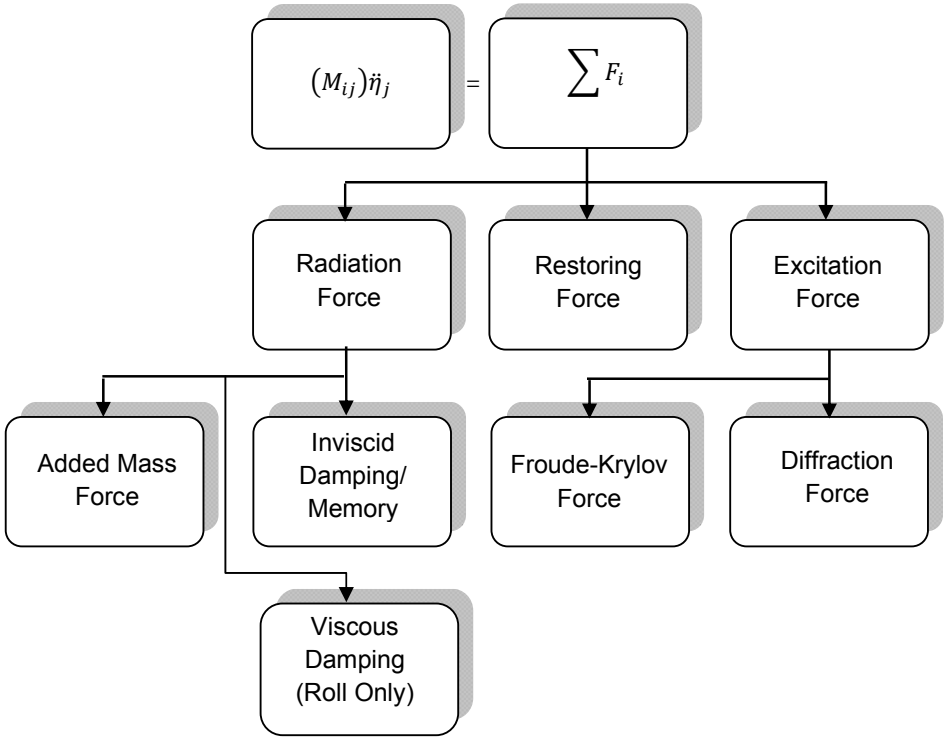


Figure 3: Block diagram of forces included in the equations of motion. Froude-Krylov and diffraction forces are excitation forces and are listed under that type.

Now, in the frequency domain, the different wave heights and wave lengths of the irregular seas, which contribute the above forces, may be superimposed, and the equations can be solved based on linear signal analysis theory. The downside of this solution is that it may not capture non-linear effects. [12] Therefore, the motion equations must be solved in the time domain. The equations of motion remain the same. The derived equations of motion are shown as a block diagram in figure 4. The subscript *j* refers to six degrees of freedom, and thus there are six motion equations to be solved.

Figure 4: The simplified block diagram of equations of motion

The time domain approaches for ship motion can be categorized into four representative methods: the impulse-response-function method, the sectional nonlinear or strip method, the unsteady green function approach, and the Rankine panel method.[7] The impulse-response function method is based on the method developed by Cummins [3] and is essentially a conversion of the frequency domain solution to the time domain solution. In other words, it applies the linear frequency domain solution to compute the retardation function for the radiation forces except for viscous damping. Then, nonlinear Froude-Krylov forces, nonlinear

viscous damping, and nonlinear restoring forces can be combined with the linear diffraction force, and the equations can be solved numerically. Since this method is basically a conversion of the frequency-domain solution to the time domain, there is a strong advantage of reduced computational time, and thus it is chosen for this project. Linearity of the forces is summarized in table 2. [7]

Table 2: Summary of Linearity in Impulse-Response Function Method

Incident Wave	Froude-Krylov and Restoring Forces	Diffraction Forces	Viscous Damping	Over all Method
Linear	Nonlinear	Linear	Nonlinear	Weakly Nonlinear

2.3. Components of Time Domain Motion Equations

This section discusses ways of computing the forces introduced in the preceding subsection.

2.3.1. Radiation Forces

To convert radiation forces, added mass, and damping from a linear frequency domain to a time domain solution, a retardation function, *K*, can be used. [3] The function is derived from the damping and is given as

$$K_{i,j}(\tau) = \frac{2}{\pi}\int_0^{\alpha} B_{i,j}(\omega).\cos(\omega\tau)\,.d\omega \qquad \dots (6)$$

where B_{ij} is the hydrodynamic damping coefficient of the ship in the frequency domain and τ is time. For numerical computation, α and $d\omega$ is replaced by maximum frequency Ω and $d\tau$. Thus, the frequency range is $0 \leq \omega \leq \Omega$. Maximum Ω value for normal merchant ships is 5 radians per second. [8] Then the retarding function can be approximated by the numerical solution of the integral as

$$K_{i,j}(\tau) = \frac{2}{\pi}\int_0^{\Omega} B_{i,j}(\omega).\cos(\omega\tau)\,.d\tau \qquad \dots (7)$$

which gives the numerical integration, at constant intervals $\Delta\omega$, as

$$K_{i,j}(\tau) = \frac{2}{\pi\tau^2}\sum_{n=1}^{N_\omega}\left\{\frac{\Delta B_n}{\Delta\omega}.[\cos(\omega_n\tau) - \cos(\omega_{n-1}\tau)]\right\} + \frac{2}{\pi.\tau}.B_{N_\omega}\sin(\omega_{N_\omega}\tau) \qquad \dots (8)$$

where $N_\omega.\Delta\omega = \Omega$ and $\Delta B_n = B_{i,j}(n) - B_{i,j}(n-1)$. [8]. Now, the frequency dependent added mass can be converted to the time domain using equation 8 as follows:

$$A_{i,j}(\tau) = A_{i,j}(\omega) + \frac{1}{\omega}\int_0^{\Gamma_{ij}} K_{i,j}(\tau).\sin(\omega\tau)\,.d\tau \qquad \dots (9)$$

where maximum time, replacing infinity term in the integral, $\Gamma_{ij} = N_\tau . \Delta\tau$, and $N_\tau$and $\Delta\tau$ are number of time step and constant time step size respectively. The integral in equation 9 can be expressed numerically as [8]

$$\int_0^{\Gamma_{ij}} K_{i,j}(\tau).\sin(\omega\tau)\,.d\tau = \frac{1}{\omega}\,.$$

$$\sum_{n=1}^{N_\omega}\left\{\frac{\Delta B_n}{\Delta\tau}.[\sin(\omega.n.\tau)-\sin(\omega.(n-1)\Delta\tau)]\right\}+\frac{1}{\omega}.$$

$$\left\{B_{ij}(\tau=0)-B_{ij}\left(\tau=\tau_{N_\tau}\right).\cos(\omega.N_\tau.\Delta\tau)\right\} \quad \ldots (10)$$

In general, the radiation force will have two terms; the first term contains the force due to added mass and the second one is related to the memory or inviscid damping force. Therefore this force can be given as in equation 11 [5].

$$F_i^{rad}(t) = \sum_{j=2}^{6} -A_{i,j}(\tau)\ddot{\eta}_j(t) - \int_0^t K_{i,j}(t-\tau).\dot{\eta}_j(\tau)\,.d\tau \quad \ldots (11)$$

2.3.2. Restoring and Froude-Krylov Forces

The restoring and Froude-Krylov forces, $F^{rest}$and F^{F-K}, may be evaluated by integrating the pressure caused by hydrostatic pressure and by the incident wave over the wetted hull surface using the incident wave potential. For numerical computation, these forces along a cross section may be written as

$$F_j = \sum_{yz} \frac{P_{yz}+P_{yz+1}}{2} l_{yz}\hat{n}_{yz} \qquad for\ 1 \le j \le 3 \quad \ldots (12)$$

$$F_j = \sum_{yz} \frac{P_{yz}+P_{yz+1}}{2} l_{yz}(\hat{r}_{yz} \times \hat{n}_{yz}) \qquad for\ 4 \le j \le 6 \quad \ldots (13)$$

where F_j is the restoring or Froude-Krylov force, l_{yz} is Euclidian length in y-z plane, $\hat{n}_{yz}$ is a unit vector, $\hat{r}_{yz}$ is a position vector, and P_{yz} is the pressure at a point on the cross section of the hull. [5] The subscript *y-z+1* refers to the next consecutive *y-z* point. Detailed computation of these forces is shown in Appendix A. The pressure for the Froude-Krylov force is then simply

$$P_{yz}^{F-K} = -\rho\frac{\partial\phi(x,z,t)}{\partial t} - \rho g z \quad \ldots (14)$$

The velocity potential of the incident wave is

$$\phi = \frac{a_w g}{\omega}\frac{\cosh(k(z+d))}{\cosh(kd)}\sin(kx-\omega t) \quad \ldots (15)$$

where a_ω is wave amplitude, *k* is wave number, ω is wave angular frequency, and *d* is water depth. [1]

Furthermore, the Froude-Krylov force may be divided into two parts: sectional force, F_T^{F-K}, and waterline force, F_W^{F-K}, depending on the direction of integration. As shown in figure 5, sectional Froude-Krylov forces may be computed by integrating the forces defined in equations 12 and 13 over the ship length, while the waterline forces are found by integrating

the forces over the ship draft. [5] Therefore, the Froude-Krylov force in six degrees of freedom is

$$F_j^{F-K} = F_T^{F-K} + F_W^{F-K} \quad \dots (16)$$

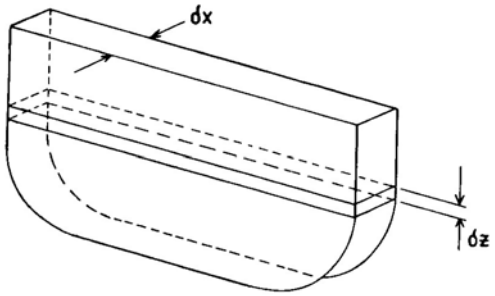


Figure 5: Finding sectional Froude-Krylov forces in dx direction and waterline forces in dz direction. [13]

2.3.3. Diffraction Forces

The diffraction forces may be computed by assuming the ship to be held fixed as waves of some given frequency impinge upon it. By doing so, the force required to hold the ship still and the effect the ship has on the waves due to its presence can be computed. The diffraction force in surge may be neglected. Forces in other directions are shown below as described by Hua. [5].

$$F_2^{Diff}(t) = \int_0^L \left[a_{22}(\tau).a_y(t) + \int_0^t k_{22}(t-\tau)v_y(\tau)\,d\tau\right]dx$$

$$F_3^{Diff}(t) = \int_0^L \left[a_{33}(\tau).a_z(t) + \int_0^t k_{33}(t-\tau)v_z(\tau)\,d\tau\right]dx$$

$$F_4^{Diff}(t) = \int_0^L \left[a_{24}(\tau).a_y(t) + \int_0^t k_{24}(t-\tau)v_y(\tau)\,d\tau\right]dx$$

$$F_5^{Diff}(t) = \int_0^L -x\left[a_{33}(\tau).a_z(t) + \int_0^t k_{33}(t-\tau)v_z(\tau)\,d\tau\right]dx$$

$$F_6^{Diff}(t) = \int_0^L x\left[a_{22}(\tau).a_y(t) + \int_0^t k_{22}(t-\tau)v_y(\tau)\,d\tau\right]dx$$

… (17)

k_{ij} are again the sectional retardation functions and $a_{ij}$are the sectional added masses in the time domain. Computations for the particle velocities and accelerations in the *y* and *z* directions, v_y, v_z, a_y, and a_z are shown in Appendix A. The forces may be computed section by section and integrated over the ship length.

2.3.4. Viscous Damping Force in Roll Motion

In calculating roll motion, viscous damping must be included. The viscous roll damping coefficient may be estimated from a roll decay test as determined experimentally. In this

project, for programming purposes, empirical roll damping prediction methods are preferred. However, because the roll damping is significantly influenced by the viscosity of the fluid, it is not possible to compute damping in a purely theoretical way. Therefore, some prediction methods are used to get the roll damping. A widely used method is Ikeda's prediction, which divides the roll damping into the frictional, B^F, the wave, B^W, the eddy, B^E, the bilge keel, B^{BK}, and the lift, B^L, components.[6] Expressed mathematically,

$$B_{44} = B_{44}^F + B_{44}^W \dot{\eta} + B_{44}^E + B_{44}^{BK} + B_{44}^L \quad \ldots (18)$$

At zero speed, the lift component is neglected. Detailed calculation of each component is shown in Appendix B. It is recommended, however, that this formula not be used for ships that have a buttock flow stern such as large passenger ships and dedicated car carriers. [6] Using Ikeda's prediction method, the viscous damping force can be expressed as

$$F^{damp} = B_{44}^F \dot{\eta} + B_{44}^W \dot{\eta} |\dot{\eta}| + B_{44}^E \dot{\eta} + B_{44}^{BK} \dot{\eta} + B_{44}^L \dot{\eta} \quad \ldots (19)$$

2.4. Computing Hydrodynamic Coefficients

Other elements in the equations of motion remaining to be determined are the hydrodynamic potential coefficients in the frequency domain. They can be obtained by any kind of method such as strip or panel methods. In recent years, strip theory methods have been extended for use in the nonlinear time domain. Considering the future development of Chalmers' time-domain source code, the linear strip method is used in the present project.

The strip method requires computation of the two-dimensional potential coefficients for each cross section. Then, the total hydrodynamic coefficients can be found easily by integrating the cross section values over the ship length. The theory on the calculation of the two-dimensional potential coefficients is given by Ursell [16] and Tasai [15]. First, using Ursell's method, a ship cross-section in a complex plane is mapped to the more convenient circular cross section in another complex plane so that the velocity potential of the fluid around the ship cross-section can be computed. To achieve a more accurate transformation of a cross-sectional hull, close-fit conformal mapping is used. Then, the y and z coordinates of a ship's cross section in a complex plane may be described as

$$y = M_S \sum (-1)^n a_{2n-1} \sin((2n-1)\theta) \quad \ldots (20)$$

$$z = M_S \sum (-1)^n a_{2n-1} \cos((2n-1)\theta) \quad \ldots (21)$$

where the constants a_{2n-1} are the conformal mapping coefficients, θ is the mapped angle, and M_s is the scale factor. The parameters are 0 to N number of parameters, where N equals to 10 is used in close-fit conformal mapping. These values are used in Tasai's theory for deep water to solve hydrodynamic coefficients. Therefore, two-dimensional added mass and damping coefficients for sway, heave, roll, and the coupling coefficient of sway into roll are described as in equation 22 and 23 respectively. [9] Coefficients: $A_0, B_0, M_0, N_0, P_0, Q_0, X_R$, and Y_R can be computed using Ursell and Tasai methods. The detailed computation of

conformal mapping, Ursell and Tasai methods of finding 2-D potential coefficients and determining mass and damping matrices are presented in Appendix C.

$$
\begin{aligned}
a_{22} &= \frac{\rho b_0^2}{2}\,\frac{M_0 Q_0 + N_0 P_0}{P_0^2 + Q_0^2} \\
a_{33} &= \frac{\rho b_0^2}{2}\,\frac{M_0 B_0 + N_0 A_0}{A_0^2 + B_0^2} \\
a_{44} &= \frac{\rho b_0^4}{8}\,\frac{Y_R Q_0 + X_R P_0}{P_0^2 + Q_0^2} \\
a_{42} &= \frac{-\rho b_0^3}{2}\,\frac{Y_R Q_0 + X_R P_0}{P_0^2 + Q_0^2}
\end{aligned}
\qquad \ldots (22)
$$

$$
\begin{aligned}
b_{22} &= \frac{\rho b_0^2}{2}\,\frac{M_0 P_0 - N_0 \theta_0}{P_0^2 + Q_0^2}\omega \\
b_{33} &= \frac{\rho b_0^2}{2}\,\frac{M_0 A_0 - N_0 B_0}{A_0^2 + B_0^2}\omega \\
b_{44} &= \frac{\rho b_0^4}{8}\,\frac{Y_R P_0 + X_R Q_0}{P_0^2 + Q_0^2}\,\omega \\
b_{42} &= \frac{-\rho b_0^3}{2}\,\frac{Y_R P_0 - X_R Q_0}{P_0^2 + Q_0^2}\,\omega
\end{aligned}
\qquad \ldots (23)
$$

where b_0 is the breadth of the cross section.

2.5. Numerical Method

The equations of motion in six degrees of freedom can be simplified as the following:

$$(M + A_{11}).\ddot{\eta}_1 + A_{15}\ddot{\eta}_5 = F_1^{ext} - B_{11}\dot{\eta}_1 - B_{15}\dot{\eta}_5 + M(\dot{\eta}_2\dot{\eta}_6 - \dot{\eta}_3.\dot{\eta}_5) \quad \ldots (24)$$

$$(M + A_{22}).\ddot{\eta}_2 + A_{24}\ddot{\eta}_4 + A_{26}\ddot{\eta}_6 = F_2^{ext} - B_{22}\dot{\eta}_2 - B_{24}\dot{\eta}_4 - B_{26}\dot{\eta}_6 + M(-\dot{\eta}_1\dot{\eta}_6 + \dot{\eta}_3.\dot{\eta}_4) \quad \ldots (25)$$

$$(M + A_{33}).\ddot{\eta}_3 + A_{35}\ddot{\eta}_5 = F_3^{ext} - B_{33}\dot{\eta}_3 - C_{33}\eta_3 - B_{35}\dot{\eta}_5 - C_{35}\eta_5 + M(\dot{\eta}_1 5 - \dot{\eta}_2.\dot{\eta}_4) \quad \ldots (26)$$

$$(I_{xx} + A_{44}).\ddot{\eta}_4 + A_{42}\ddot{\eta}_2 + A_{46}\ddot{\eta}_6 = F_4^{ext} - B_{42}\dot{\eta}_2 - B_{44}\dot{\eta}_4 - B_{46}\dot{\eta}_6 - C_{44}\eta_4 + (I_{yy} - I_{zz})\dot{\eta}_5\dot{\eta}_6 \quad \ldots (27)$$

$$\left(I_{yy} + A_{55}\right).\ddot{\eta}_5 + A_{51}\ddot{\eta}_1 + A_{53}\ddot{\eta}_3 = F_5^{ext} - B_{51}\dot{\eta}_1 - B_{53}\dot{\eta}_3 - C_{53}\eta_3 - B_{55}\dot{\eta}_5 - C_{55}\eta_5 + (I_{zz} - I_{xx})\dot{\eta}_4\dot{\eta}_6 \quad \ldots (28)$$

$$(I_{zz} + A_{66}).\ddot{\eta}_6 + A_{62}\ddot{\eta}_2 + A_{64}\ddot{\eta}_4 = F_6^{ext} - B_{62}\dot{\eta}_2 - B_{64}\dot{\eta}_4 - B_{66}\dot{\eta}_6 + (I_{xx} - I_{yy})\dot{\eta}_4\dot{\eta}_5 \quad \ldots (29)$$

where F_{1-6}^{ext} are diffraction forces and Froude-Krylov forces. To solve the above second-order differential equations, they are reduced to a set of first-order differential equations and solved with MATLAB's built-in functions, such as ODE45, ODE113, ODE15s, and DDE23.

MATLAB documentation recommends first trying one of the ordinary differential equation solvers: ODE45, which is implemented as a fourth- or fifth-order Runge-Kutta numerical scheme. If the ODE45 is slow because the equations are stiff, ODE15s is to be used. This ODE15s is a quasi-constant step-size implementation of the numerical differentiation formulas in terms of backward differences. The MATLAB ode113 solver uses a multistep Adams-Bashforth-Moulton (ABM) method. ABM is multistep method, consisting of the explicit, linear multistep Adams-Bashforth formula, which is used as the predictor formula that determines an approximation of *y* at time *t+h* using values at times *t*, *t-h*, *t-2h*, and *t-3h* and the implicit, linear multistep Adams-Moulton formula, which is used as the corrector formula. The advantage of using this two-step method is that it reduces the amount of computation.

On the other hand, DDE23 is a delay differential equation solver that computes derivatives based on the solution at the previous time steps. Similar to ODE solvers, DDE23 is based on explicit Runge-Kutta method. [14] The advantage is that this solver handles the memory force computation that involves past time steps better than ordinary differential equation solvers. The efficiency of these differential equation solvers is to be compared.

2.6. Error Assessments and Numerical Stability

To obtain confidence in the numerical solution, its stability must be studied. The solutions is said to be in stable when the difference between the continuous and discrete solutions does not grow over time. In other words, the solution must converge to achieve an accurate representation of the equations of motion.

In solving the equations of motion numerically, six second degree differential equations are written as twelve first degree equations. For instance, the dynamical system can be represented as

$$\frac{d\vec{y}}{dt} = \vec{f}(t) \quad \dots (30)$$

where

$$\vec{y}(t) = \begin{Bmatrix} \vec{y}_1(t) \\ \vec{y}_2(t) \end{Bmatrix} = \begin{Bmatrix} \dot{\vec{\xi}}(t) \\ \vec{\xi}(t) \end{Bmatrix} \quad \dots (31)$$

and

$$\vec{f}(t) = \begin{Bmatrix} \vec{f}_1(t) \\ \vec{f}_2(t) \end{Bmatrix} = \begin{Bmatrix} -[M + a_0]^{-1}(b_0\vec{y}_1(t) + (C + c_0)\vec{y}_2(t) + \int_0^t K(t-\tau)\vec{y}_1(\tau)d\tau \\ \vec{y}_1(t) \end{Bmatrix} \quad \dots (32)$$

Then, a global error equation can be expressed as

$$\epsilon(\vec{t}_n) = \vec{y}(t_n) - \vec{y}_n \quad \dots (33)$$

where the vector $\vec{y}_n$ represents the numerical solution for each direction of motion. The discrete time step is t_n. From the error equation, the discrepancy between continuous and discrete solutions can be measured. [10] To examine the error equation in Laplace space, the equation is linearized with respect to time. By doing so the impulse response function and the error are separated, and error propagation can be studied independently.

A stability polynomial can be derived from the von-Neumann assumption. The stability condition which is the combination of the stability polynomial and the absolute stability criteria defines the range of the time-step size in this time domain study.

2.7. Summary

Radiation, viscous damping, diffraction, Froude-Krylov, and restoring forces are composed in second-order differential motion equations and will be solved for the six degrees of freedom time domain solution. The radiation force in time domain will be derived from the frequency domain using the retardation function. Thus, the solution technique will follow the Impulse-Response Function method. Viscous damping will be considered only in roll motion and the Ikeda's prediction will be used to compute the damping. Diffraction and Froude-Krylov forces will be included in the motion equations as external forces. Finally, second-order differential equations will be reduced to a set of first-order differential equations and solved with MATLAB's built-in differential motion solvers.

3. Source Code Development and Software Implementation

3.1. Overview

The purpose of this section is to highlight the computer program development steps. Program development may be divided into two parts. The first part includes the MATLAB source code development, and the second part describes the development of a mobile device application. The development follows the ITTC-recommended procedures and guidelines for seakeeping computer codes. [7] The MATLAB source code can be compiled and used as an executable program. This program is named Chalmers Big Time (CBT). CBT has a graphical user interface that may be used to enter the information for ship-motion simulation and to view the results. The mobile device application is named Chalmers Big Time Mobile (CBT-M). The two programs, CBT and CBT-M, are designed so that an in-progress computation on the CBT can be transferred to the CBT-M and vice-versa. In this project, only the linear frequency domain solution can be transferred, and only from the CBT to the CBT-M.

3.2. Development of MATLAB Source Code

MATLAB source code may be subdivided into two parts: the code that supports the time domain simulation and the code that communicates with the user and controls the computation. The co-operation between the two parts is described as a flow chart in figure 5.

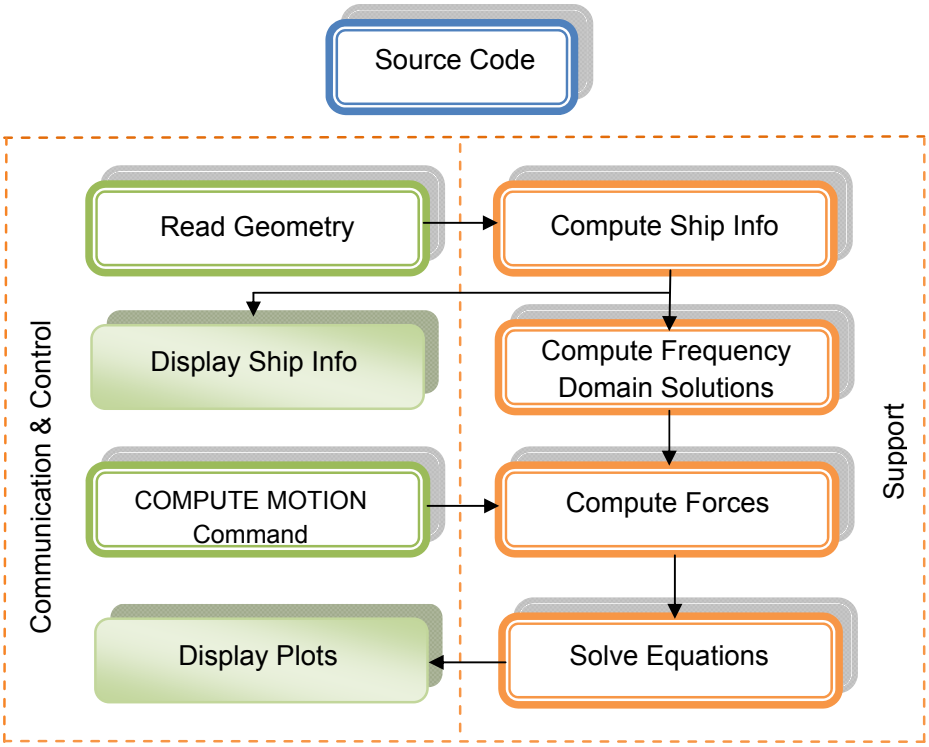


Figure 5: Relationship between communication and control code and the support code. The Graphical User Interface contains the communication and control code.

3.2.1. Development of MATLAB User Interface

The main functions of the CBT user interface are to allow the user to enter data, to start the computation, to view the progress of the computation, and to retrieve the results. The CBT user interface implemented with the above functions is shown in figure 6. Selecting the correct menu option causes CBT to read the Non-Uniform Rational Basis Spline (NURBS) geometry file and display the correct representation of the model. Once the model is read, the CBT evaluates and computes the stability information and displays the results.

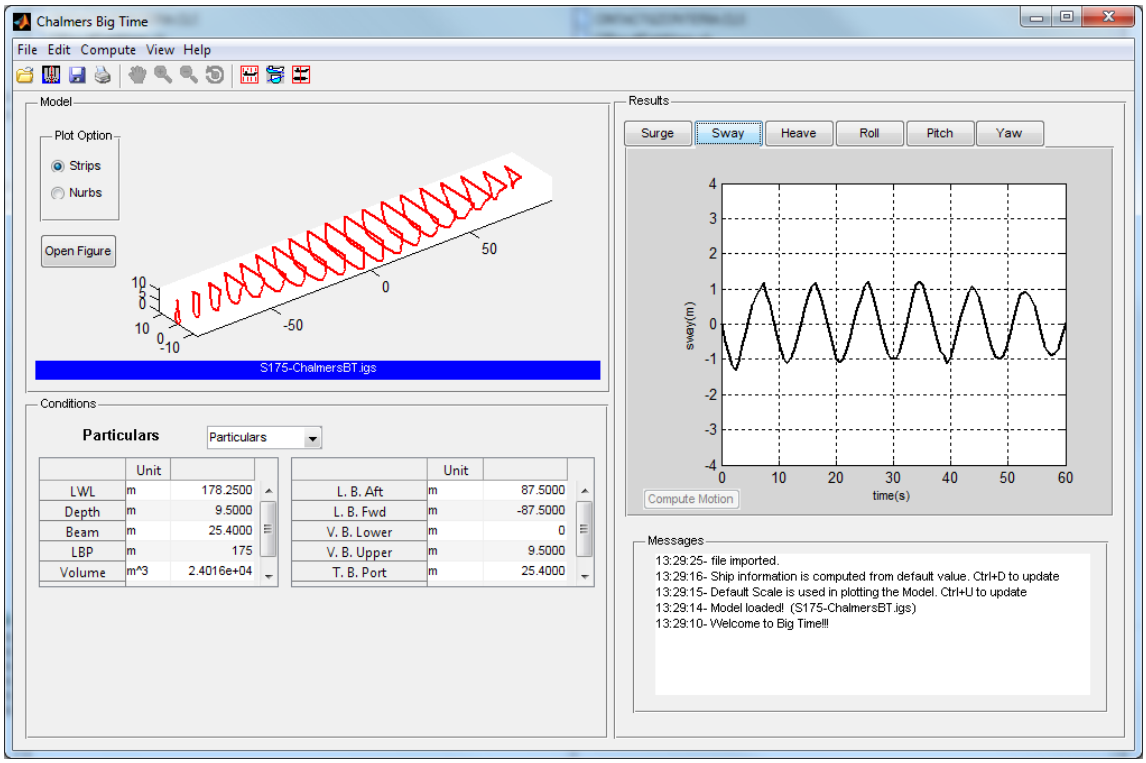


Figure 6: Screen shot of Chalmers Big Time Program Graphical User Interface. GUI has four displays. Upper left quadrant displays the model profile and its file name. Lower left quadrant shows the computed data. Result Plots are shown in the Upper right quadrant. The last quadrant display messages. COMPUTE MOTION command is placed in the result quadrant.

The CBT includes four user data entry forms, which can be called from dropdown menus. These forms collect the data required for computation. Important user inputs include time step size, angular frequency, and draft. These forms are shown in figure 7. The CBT is implemented with several command buttons. The COMPUTE MOTION command starts the full computation process if the computation for the loaded model has not already been run. During the first computation, the routine calculates both the frequency and the time domain solutions. The frequency solution is stored in the environment for further time domain computation until a new ship model is loaded or new strip calculation data is received.

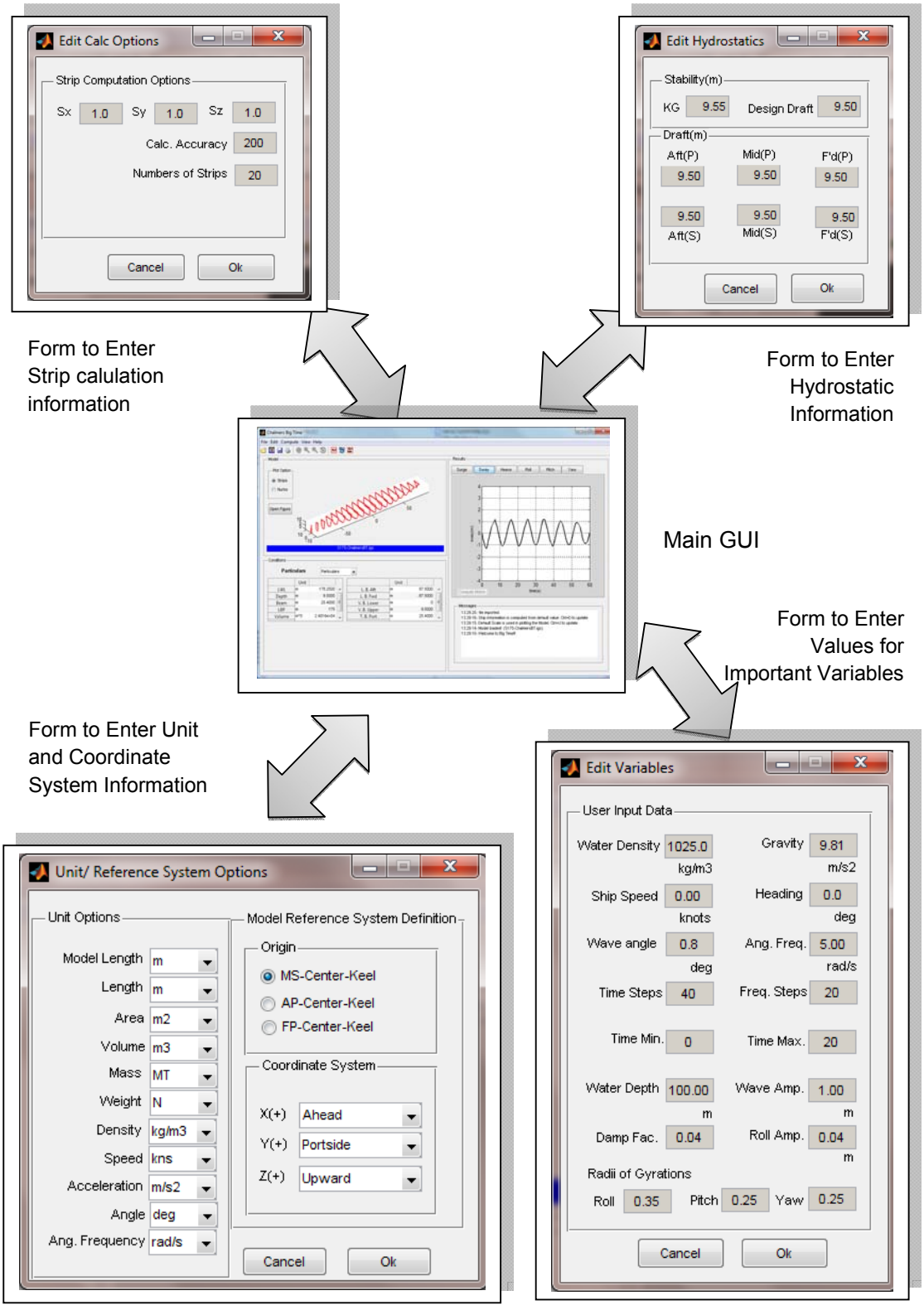


Figure 7: Screen shots of the main GUI and user data entry forms. The entry forms can be called from the GUI. It holds all defaults and user entry values for forms.

During the computation process, the stage of computation, error, and warning are shown in the text box of the user interface under the plot tab. The same information is shown in the MATLAB command window. Once the computation is completed, the ship responses in six degree of freedoms are presented in respective plots. In the future, the customizable printing of results will be possible.

The support code can be subdivided into two parts: the code to support the system and the code that performs actual computation. An example of system support code includes unit and reference frame conversion. The background theory described in the previous section must be developed in the latter part. The information on computational source code development is described in the following subsections.

3.2.2. Establishing Reference Systems

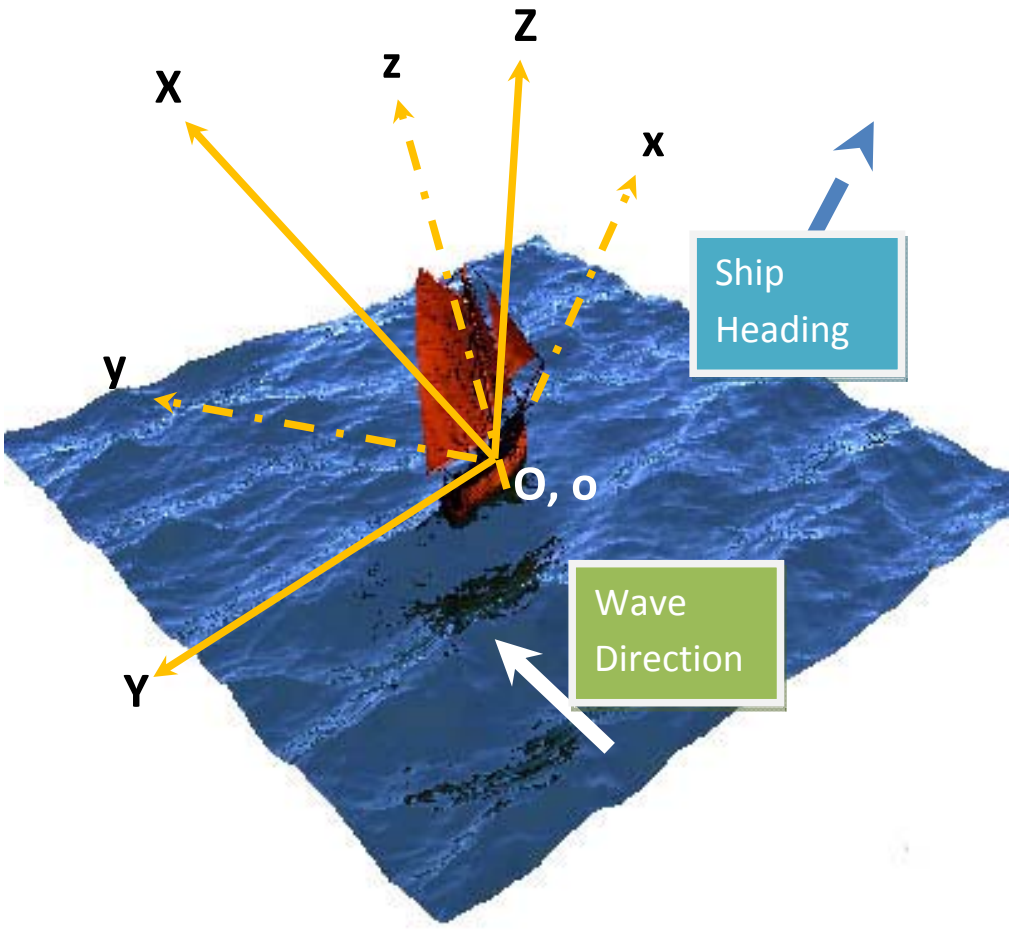


Figure 8: Relationship between the global reference system OXYZ and the ship reference system oxyz. The global system will be based on wave direction and still water plane, while the ship reference system is related to the ship heading and motions. The angle between X and x direction is β

The definition of a ship reference system oxyz was introduced in section 1.3. To develop a relationship between waves and ship motions, a global reference system OXYZ should be established. In the global reference system, the x direction is defined by the general direction of wave propagation and is denoted by the capital letter X as shown in figure 8. Thus, the X-Y plane represents the still water plane while the x-y plane in a ship reference system is parallel to the horizontal plane of the ship in motion. The directions x and X differ by an angle between the ship heading and the wave direction. Furthermore, the Z direction is shown vertical and perpendicular to the water plane while the z direction is perpendicular to the ship horizontal plane with respect to roll and pitch motions at a given time. Origin of these reference systems are placed at the ship's center of gravity. Thus, for the ship in upright position following the waves, the systems will be the same.

Coordinates may be transformed from one system to another by multiplying the coordinate matrix and the transformation matrix. [5] The transformation matrix T is defined in equation 34.

$$[T] = \begin{bmatrix} \cos\Psi\cos\theta & \cos\Psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\Psi\cos\theta & \sin\Psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad \dots (34)$$

where ψ, θ, and ϕ are the Euler angles for yaw, pitch, and roll respectively. Yaw angle is measured from the ship heading. The wave angle, β, is measured between ship heading and wave direction; β must be applied to ψ before using it in the equation. Transformation from XYZ coordinates to xyz, $T^{OXYZ \to oxyz}$, or vice versa can be carried out as in equation 35.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = [T].\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad \dots (35)$$

3.2.3. Computing Froude-Krylov Forces

An important task in computing Froude-Krylov forces is to find the underwater pressure points along the ship hull. The wave model lies on the OXYZ reference system, while the ship sectional points along the hull are constructed based on the oxyz system. Therefore, the sectional points may be transformed from xyz coordinates to XYZ coordinates at each time step to determine whether a strip point is under the water surface as described in figure 9. For each transformed ship section, the points immediately under the wave model will be collected and used to compute underwater pressure in the OXYZ system as described in equation 14. Waterline forces can be computed in a similar way. In addition to the integration direction changes from the ship length to the ship draft, the matrix transformation in the $\hat{n}_{yz}{}^{oxyz}$ unit vector is required for computing waterline Froude-Krylov forces [5], as shown in equation 36.

$$\hat{n}^{oxyz} = T^{oxyz \to OXYZ}(T^{OXYZ \to oxyz}.\hat{n}^{OXYZ}.\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}) \quad \dots (36)$$

Finally, the Froude-Krylov forces computed in equation 12 and 13, which are still in the OXYZ reference system, must be transformed back to the oxyz reference system for ship motions computation. The transformed x, y, and z components of the Froude-Krylov force

computed in equation 12 will be used for surge, sway, and heave computations and other transformed x, y, and z components related to equation 13 will be used for roll, pitch, and yaw computations respectively.

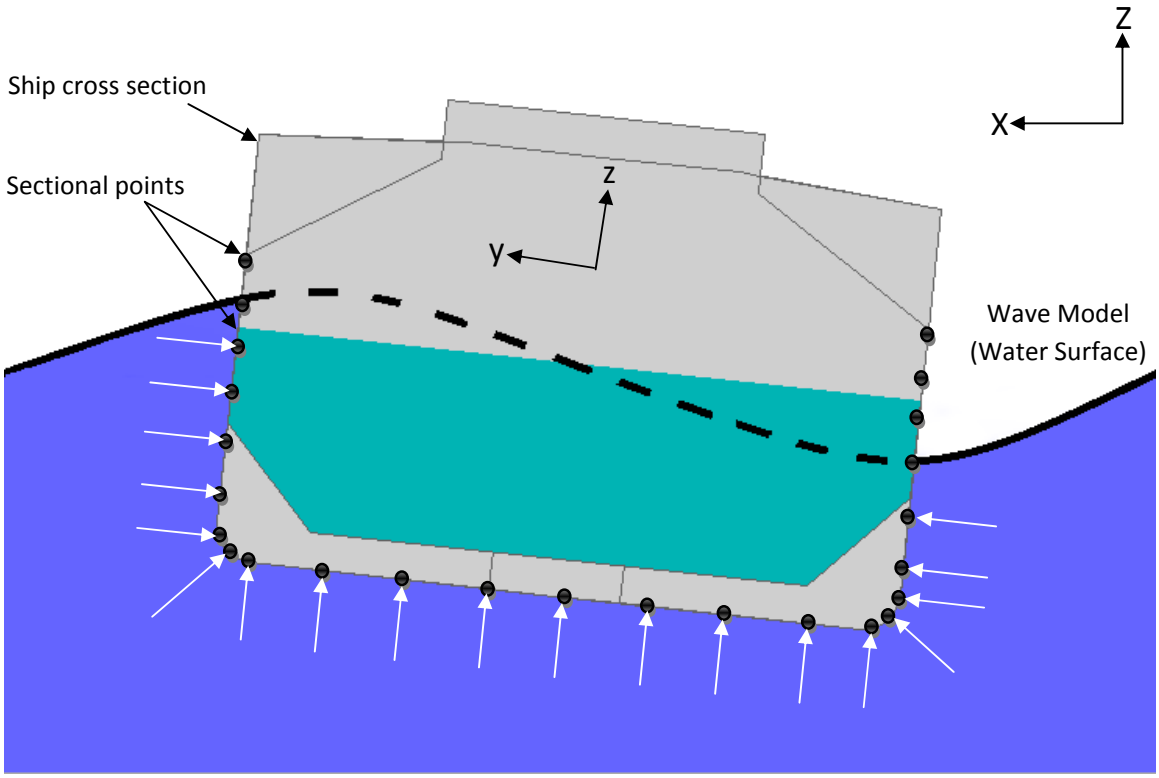


Figure 9: Intersection between wave model and ship cross section. Underwater points are determined at each time step and are used to compute non-linear Froude-Krylov force.

3.2.4. Mathematical Model

Other forces such as memory forces, viscid damping, and diffraction are straightforward and can be computed as described in the theory section. The equations of motion in six degrees of freedom provided in equation 24 through 29 can now be solved numerically. To solve the second order differential motion equations, the equations can be reduced to the first order ones as shown in equation 37 and 38. They may be included in a function and called from an ODE solver.

$$dy(1) = y(7)$$

$$dy(2) = y(8)$$

$$dy(3) = y(9)$$

$$dy(4) = y(10)$$

$$dy(5) = y(11)$$

$$dy(6) = y(12) \quad \ldots (37)$$

$$dy(7) = -\sum_{j=1}^{6} \frac{C(1,j)}{A+M} y(j) - \sum_{j=1}^{6} \int_0^t \frac{K_{1,j}(t-\tau)}{A+M} . y(j) . d\tau \; + \frac{F_1^{F-K}}{A+M}$$

$$dy(8) = -\sum_{j=1}^{6} \frac{C(2,j)}{A+M} y(j) - \sum_{j=1}^{6} \int_0^t \frac{K_{2,j}(t-\tau)}{A+M} . y(j) . d\tau \; + \frac{F_2^{F-K}}{A+M} + \frac{F_2^{Diff}}{A+M}$$

$$dy(9) = -\sum_{j=1}^{6} \frac{C(3,j)}{A+M} y(j) - \sum_{j=1}^{6} \int_0^t \frac{K_{3,j}(t-\tau)}{A+M} . y(j) . d\tau \; + \frac{F_3^{F-K}}{A+M} + \frac{F_3^{Diff}}{A+M}$$

$$dy(10) = -\sum_{j=1}^{6} \frac{C(4,j)}{A+M} y(j) - \sum_{j=1}^{6} \int_0^t \frac{K_{4,j}(t-\tau)}{A+M} . y(j) . d\tau \; + \frac{F_4^{F-K}}{A+M} + \frac{F_4^{Diff}}{A+M} - \frac{B_{44} y(4)}{A+M}$$

$$dy(11) = -\sum_{j=1}^{6} \frac{C(5,j)}{A+M} y(j) - \sum_{j=1}^{6} \int_0^t \frac{K_{5,j}(t-\tau)}{A+M} . y(j) . d\tau \; + \frac{F_5^{F-K}}{A+M} + \frac{F_5^{Diff}}{A+M}$$

$$dy(12) = -\sum_{j=1}^{6} \frac{C(6,j)}{A+M} y(j) - \sum_{j=1}^{6} \int_0^t \frac{K_{6,j}(t-\tau)}{A+M} . y(j) . d\tau \; + \frac{F_6^{F-K}}{A+M} + \frac{F_6^{Diff}}{A+M}$$

… (38)

3.3. Mobile Device Software Implementation

Due to the intense computation demands of the time domain method, the full implementation of MATLAB source codes in a mobile device is currently not possible. After several attempts, the computation transfer process has been introduced.

3.3.1. Architecture

The common approach to develop mobile device simulation should be the web-based simulation approach. In this approach, simulation is chosen to take place on the server instead of on the client mobile device. In server-side simulation, only the codes for the interactive graphical user interface will be developed. The main idea behind this approach is to compute a large portion of the numerical calculations in the MATLAB program and retrieve the solution from a mobile device. In this way, mobile device users with Internet access can review, process, and analyze solutions. The numerical calculations and visualizations, such as generation of plots, are carried out on the web server using MATLAB. The latest “cloud computing” technologies may be implied in this approach. Applications already present are GNU Octave web interfaces, Google Chart API, and SAGE open source numerical analysis software.

Although the approach should focus on dealing with technology advances, implementing the server-side web-based simulation requires physical resources and online services that may not be feasible in this thesis project. An alternative approach would be a stand-alone simulation with an outsourcing preprocessor. In this approach, the ship model would be read, and the frequency domain part would be computed in MATLAB or CBT in a preprocessing

step. Then, the frequency domain solution could be imported to a mobile device to continue computing the time domain part. An Internet connection would not be required for time domain calculations. The simulation and postprocessing would be carried out in a mobile device. The flow chart of the CBT-M system is shown in figure 10.

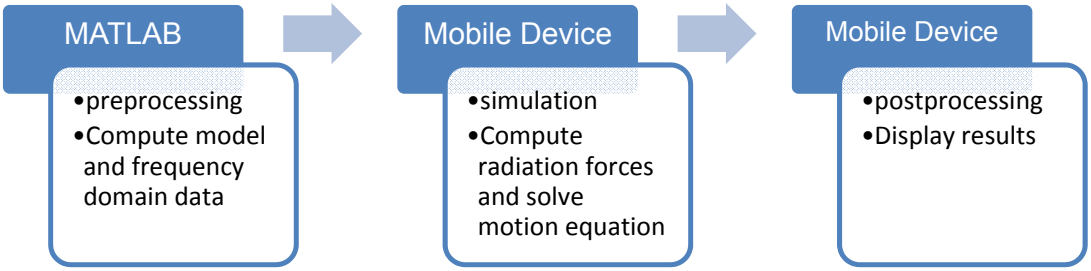


Figure 10: Flow chart of Chalmers Big Time Mobile system

3.3.2. Development Environment

Mobile devices such as smartphones, handheld PDAs, and Tablet PCs operate on mobile operating systems including iOS from Apple, Android Os from Google, and Windows Mobile from Microsoft. Based on research conducted for the July 2011 by Net Market Share [18], as in figure 11, iPhones running iOS dominated the Smartphone market, with approximately 55% of all mobile web browsing performed on the device. Therefore, the iOS platform is chosen for CBT-M to reach a wider audience.

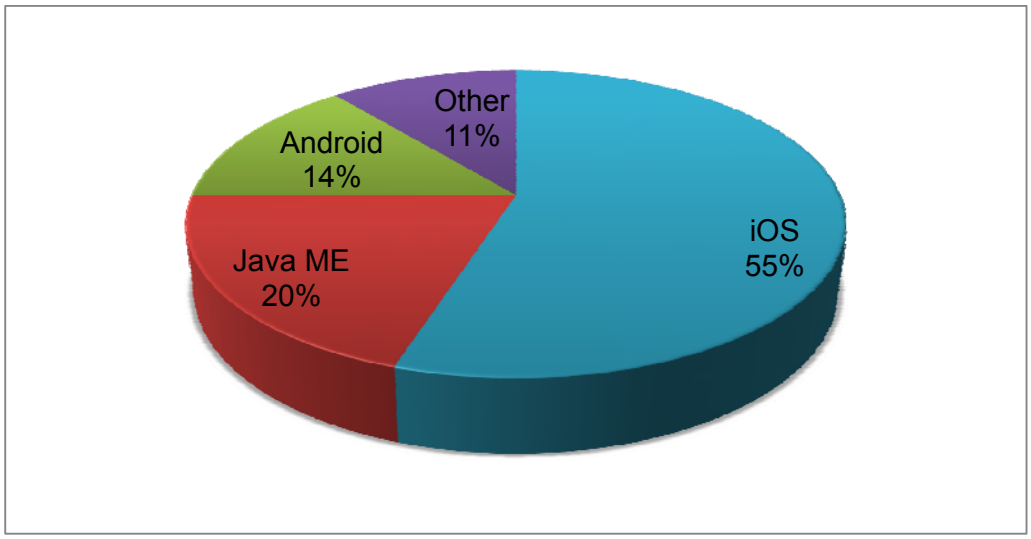


Figure 11: Pie chart of all mobile web browsing performed on the mobile device in July 2011. About 55 percent of all mobile users relied on iOS platform according to Net Market Share. [18]

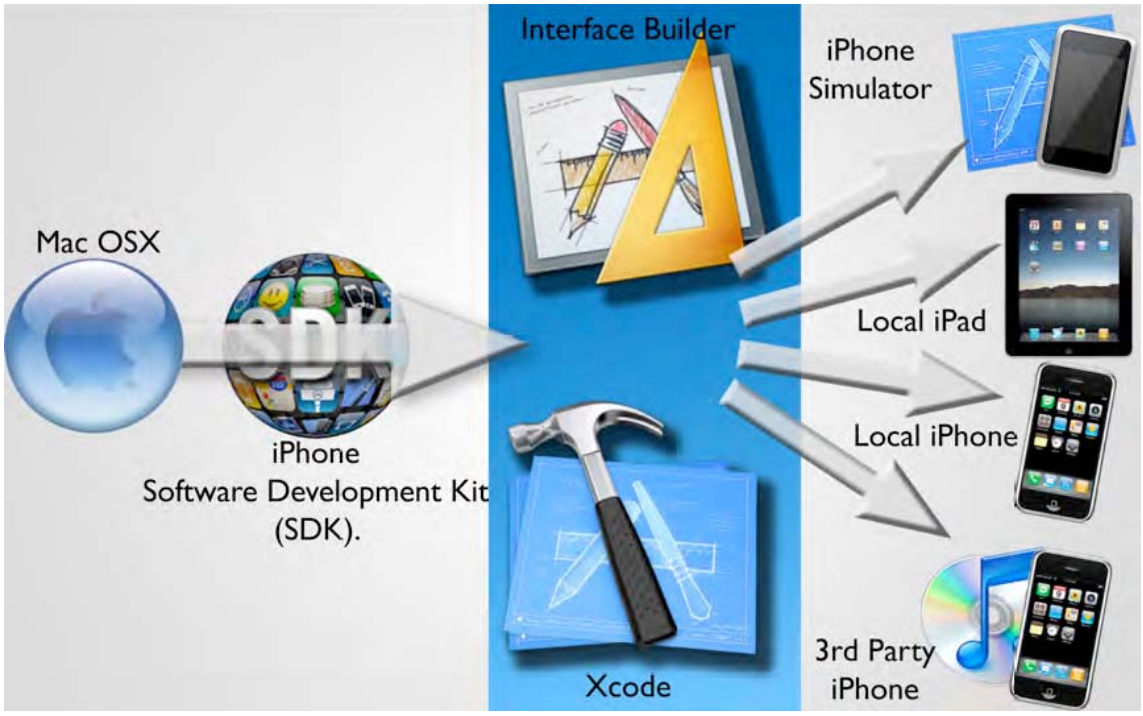


Figure 12: Mac OS X is housing the iPhone and iPad SDK. The SDK's Xcode and the Interface Builder are used to develop CBT-M. (Photo Courtesy of Lewis [11])

As show in the figure 12, CBT-M is developed using Mac OSX and the SDK, which includes Interface Builder and Xcode. The Xcode is used to manage, write, run, and debug the CBT-M—to create the computation code and functionality. Then, Interface Builder is used to drag and drop items onto the CBT-M interface. Finally, the program is compiled to the iPhone/iPad Simulator, as shown in figure 13.The CBT-M is developed to read frequency domain output from CBT and compute the time domain for motion due to radiation forces. Because of a lack of available researcher time, CBT-M has not been implemented for plotting. The heave, roll, and pitch response values in meters and degrees are shown, along with the specified time value. The graphical user interface for CBT-M is shown in figure 13.

Finally, discipline and rules for the source code development must be explained. This development work is an ongoing project. The codes must be written in a standardized way so that other researchers can easily modify, edit, or use them for future development. Thus, the name for user interface or communication and control code files will include the letters (frm) along with a given name. A support code file used for specific purposes will be named starting with the letter (c). Similarly, structure array variables will be named with the letters (struct) at the beginning. String variables will be named starting with the letter (s). The detailed discipline and rules developed in this project is listed in Appendix E. The MATLAB code can be downloaded from the thesis website as provided below:

http://web.student.chalmers.se/groups/mpnav/section/thesis/timedomain/

Figure 13: Screen shot of Chalmers Big Time Mobile program user interface in the iOS simulator mode. Users can update calculation options by pressing the green command button on the right. CBT-M reads the frequency domain and required external force solutions, prepared by the source code, and computes the ship motions.

3.4. Summary

The source code, developed in MATLAB, includes two major parts: the user interface and the calculation engine. The user interface can read NURBS ship models. The input data required for calculation can be entered through the user interface. Before time domain computation, the NURBS ship form is analyzed for stability information as well as hydrodynamic coefficients: added mass, damping, and stiffness coefficients. The coefficients are saved in a temporary file for later used without re-computing the NURBS form. The full strip computation is carried out only if a new NURBS form is loaded. The file is updated as soon as the first part, strip computation, is completed. At that time, strip information may be written to a file to be used in iOS. The strip information can be read both from the MATLAB-based CBT program and from the iOS-based CBT-M. The source code CBT can compute full time-domain analysis while CBT-M only computes basic motions equations in the time domain.

4. Results, Validation, and Discussion

4.1. Overview

The purpose of this section is to validate the results of the Chalmers Big Time source code. The code is tested against wave loads on a well-known ITTC container model, S175. The comparison is also made with the simple box rectangular shape.

4.2. Box Ship Validation

To analyze simple results such as hydrodynamic coefficients in the frequency domain, a rectangular box-shaped ship is tested first. When the model is loaded from Chalmers Big Time, CBT computes the main particulars, and hence the particulars are validated and listed in table 3. User entry data for computation are listed in table 4. Dimensionless box-ship response results for heave, roll, and pitch are compared with those of Ship Motion Control–Lewis and shown in figure 9. Most of the data used in this validation process have been borrowed from Erik Ovegård, who has previously worked on parametric rolling.

Table 3: The particulars of the rectangular box shaped vessel computed by CBT

Particulars	Unit	Model Value	CBT Result	Percent Error (%)
LBP	m	40	40	0
Beam	m	8	8	0
Depth	m	2	2	0
Water Plane Area	m^2	320	320	0
Volume	m^3	640	640	0

Table 4: User entry data for box ship computation

Particulars	Unit	Model Value
LCG	m	20
VCG	m	2
Radius of Inertia, roll	m	3.2
Radius of Inertia, pitch	m	10
Radius of Inertia, yaw	m	10

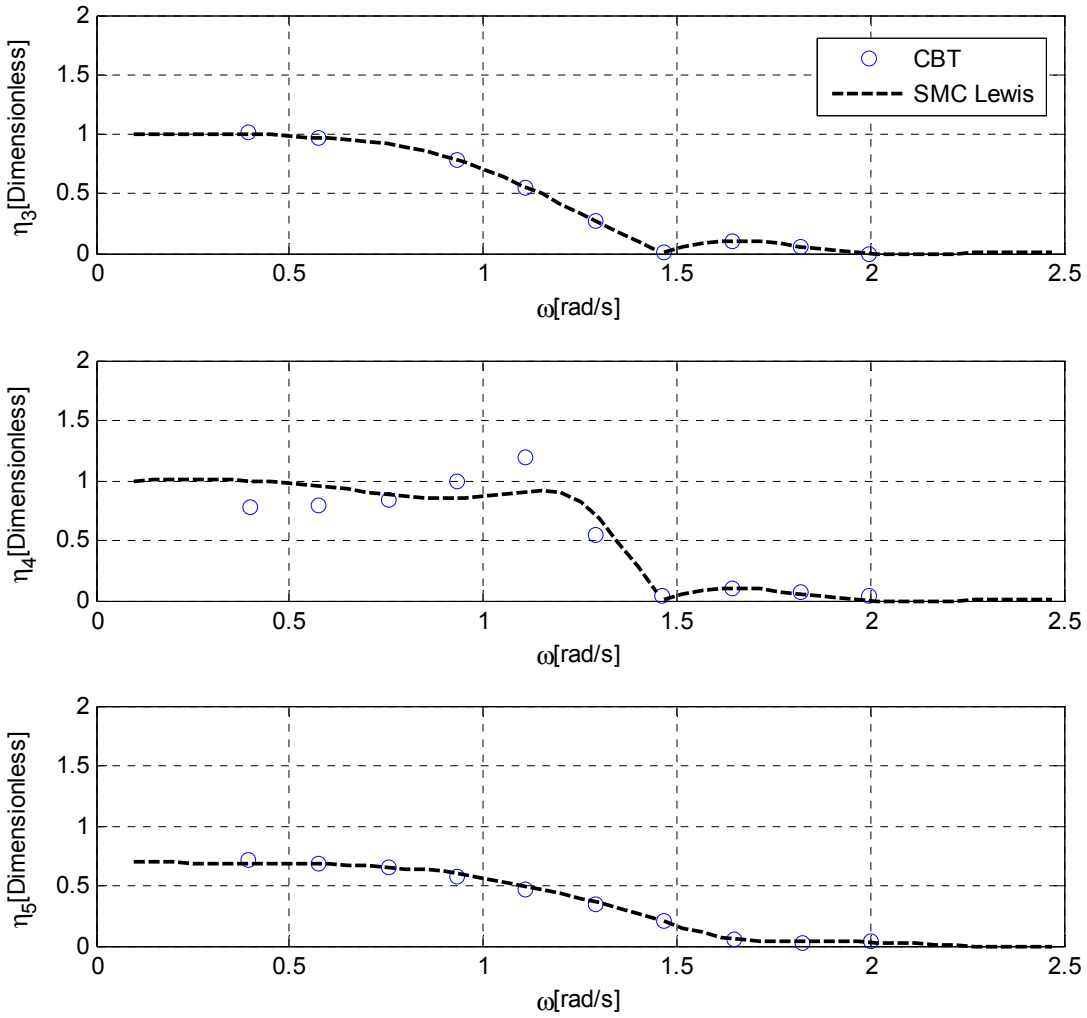


Figure 14: Frequency response functions for the box shaped vessel at zero speed. The amplitude of the wave is 0.1 m and the angle of attack is 45 degree.

Heave and pitch responses closely follow the SMC-Lewis results. However, in roll motion, the simulation predicts a higher response than SMC's values for cases in which the wave frequency is closer to the neutral frequency. Beam sea responses are further studied; the differences between the CBT-simulated results and the SMC's results become smaller in beam seas. The discrepancy may arise from using different roll prediction methods. The improvement in roll motion prediction will be required for future development and modification of source code. Validating the results with more reliable model test data may be necessary.

4.3. Validation of Adopted Methods

The portion of the source code that supports the conformal mapping is tested using the data provided by Journée. [9] The plot is shown in figure 15 and demonstrates that the conformal mapping code is valid. Furthermore, Ikeda's prediction method implemented for viscous damping is verified by comparing the results with Ikeda's published results [6]. The plot shown in figure 16 indicates that the implemented prediction code is a correct representation of the Ikeda's method.

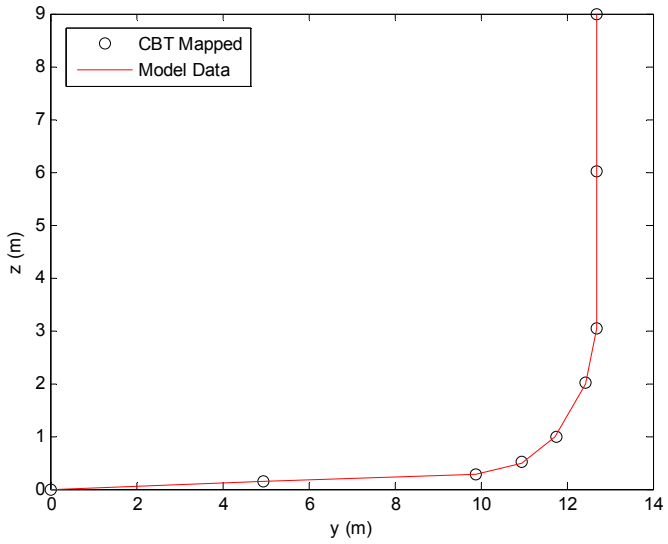


Figure 15: CBT Close-Fit Conformal Mapping Validation of a half cross section model. The computation achieves good mapping result for the time domain simulation.

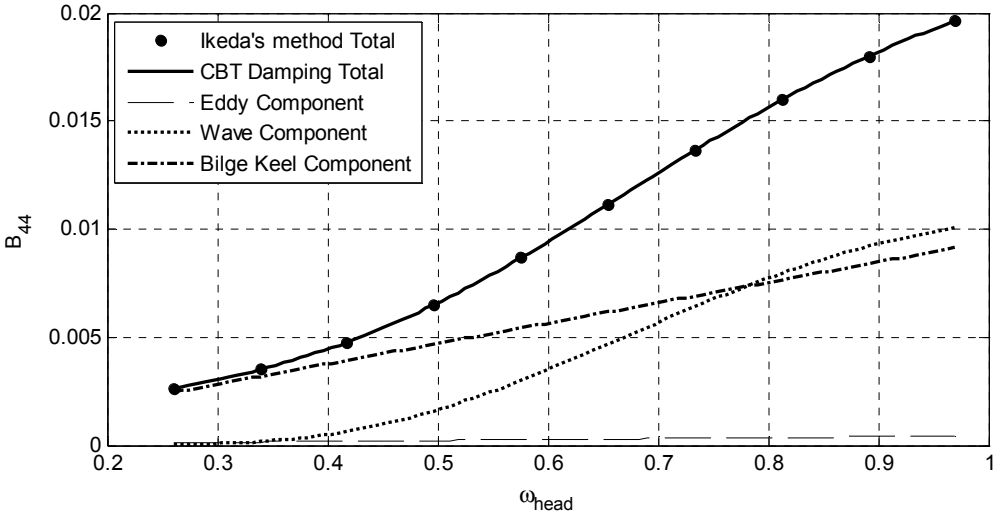


Figure 16: Results of Ikeda's prediction method at length/breadth=6.0, breadth/depth=4.0, block coefficient =0.65, mid-ship section coefficient =0.98, roll angle=10°, bilge keel breadth/breadth=0.025 and bilge keel length/Lpp=0.2.

4.4. Validation and Numerical Results for the S175

In this subsection, the motions of ITTC container S175 will be simulated using CBT. As per ITTC guidelines and recommendations for the numerical study, the body plan of S175 is first described in figure 16. The main particulars of the S-175 container ship are given in table 5 and 6.

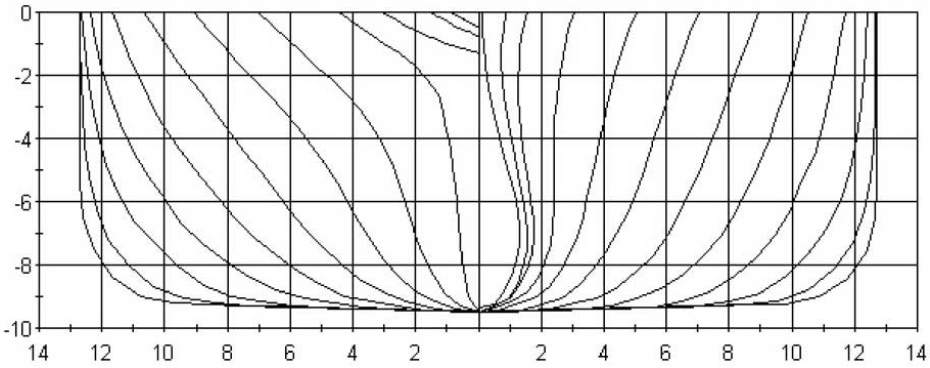


Figure 16: Body plan of ITTC container ship S175. IGS file of S175 was created during the project.

Table 5: The particulars of container vessel S175 computed by CBT

Particulars	Unit	Model Value	CBT Result	Percent Error (%)
LBP	m	175	175	0
Beam (amidships)	m	25.4	25.4	0
Depth (amidships)	m	9.5	9.5	0
Water Plane Area	m^2	3147	3152	0.2
Volume	m^3	24140	24016	0.5

Table 6: User entry data for S175 simulation

Particulars	Unit	Model Value
LCG (aft midship)	m	2.34
VCG (above base line)	m	9.52
Radius of Inertia, roll	m	8.3
Radius of Inertia, pitch	m	42.8
Block Coefficient		0.57

For S-175 model, bilge keel length and breadth are 43.75 m and 0.45 m respectively. Valid application of the Ikeda roll damping prediction method is limited to certain ships. Table 7 verifies that S175 is a valid model for using Ikeda's prediction method.

Table 7: S175 Validity test to use the Ikeda's roll damping prediction

Parameter	S175 Value	Requirement	Pass/Fail
Block Coefficient, C_b	0.57	$0.5 \leq C_b \leq 0.85$	Pass
Breadth to Draft Ratio, *B/T*	2.7	$2.5 \leq \mathrm{B/T} \leq 4.5$	Pass
OG to draft Ratio, *OG/T*	-2.0 e-3	$-1.5 \leq \mathrm{OG/T} \leq 0.2$	Pass
Mid-ship section Coefficient, C_m		$0.9 \leq C_m \leq 0.99$	N/A

A regular wave model is used to validate S175 motion. Constant amplitude 1 m, angular frequency 0.598 rad/s (wave period 10.5s), and water depth 100 m are used to generate regular waves as shown in figure 17. The simulation of the S175 model includes three wave directions: head waves, four-point bow waves (45 degrees from ahead), and beam waves.

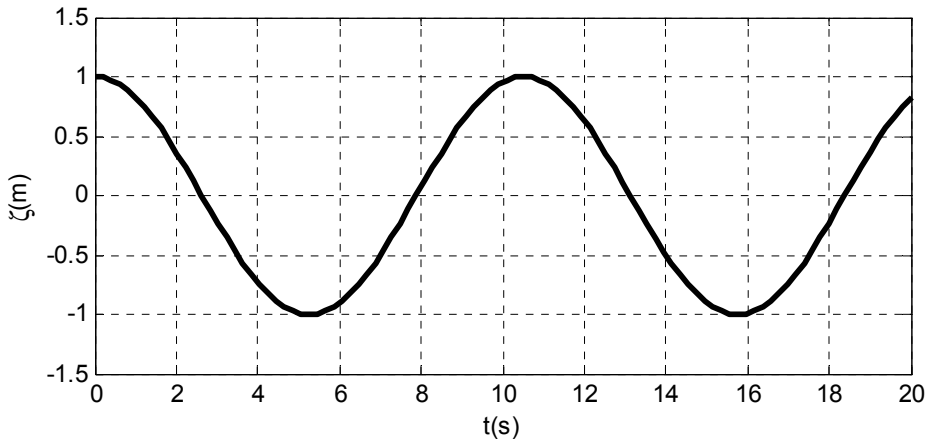


Figure 17: Time histories of wave elevation at mid-ship. Amplitude 1 m and wave period 10.5 s are used in S175 validations. Wave number is 0.03650 m^{-1}. Wave length to ship length ratio is 0.98.

Frequency dependent added mass and potential damping solutions at zero speed are shown in figures 18 and 19, respectively. They are validated against ShipX Vessel Responses (VERES) data. Coefficients not shown here are zero.

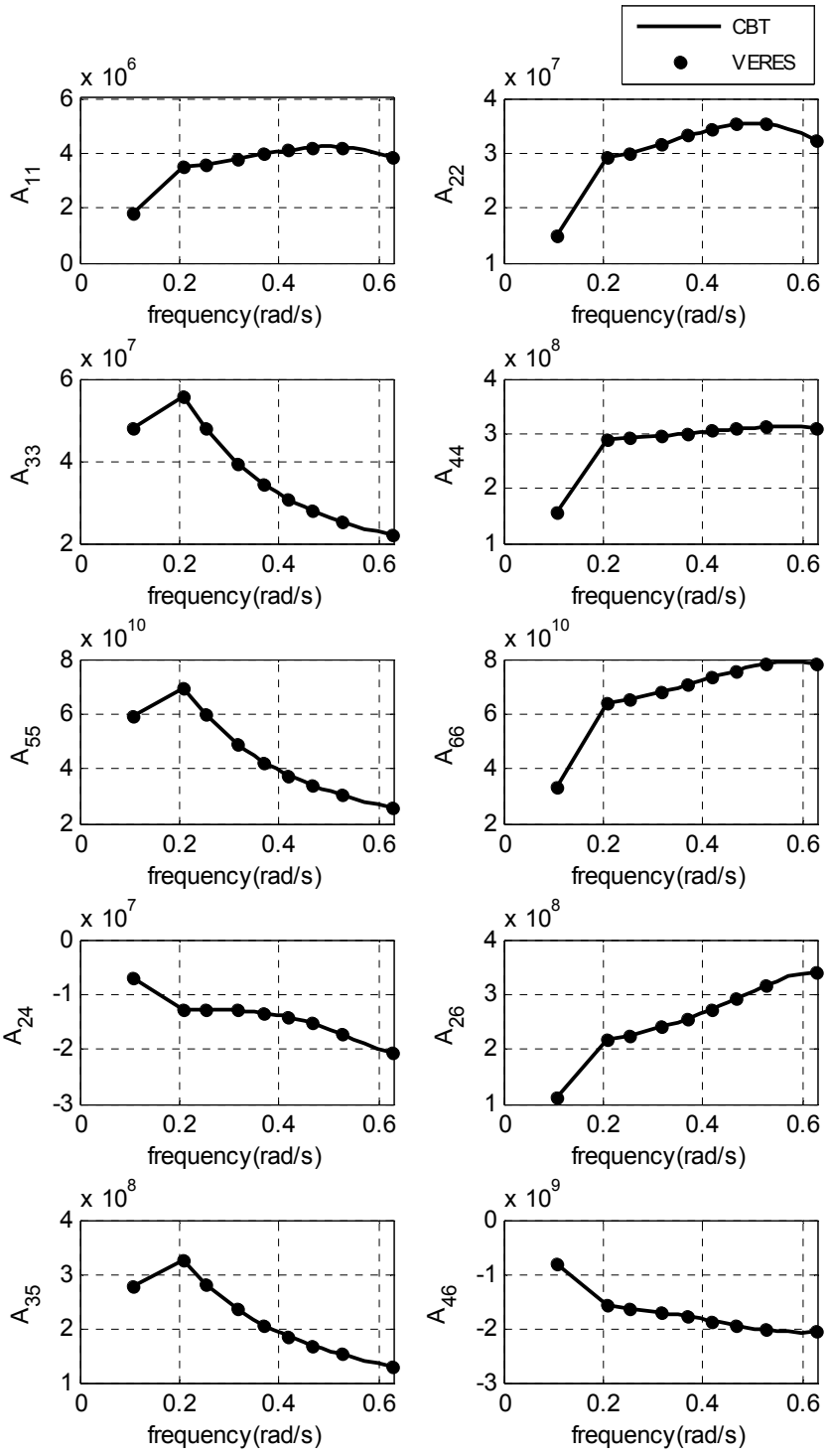


Figure 18: Frequency dependent added mass A11, A22, A33, A44, A55, A66, A24, A26, A35, and A46 for the S-175 container ship at zero speed.

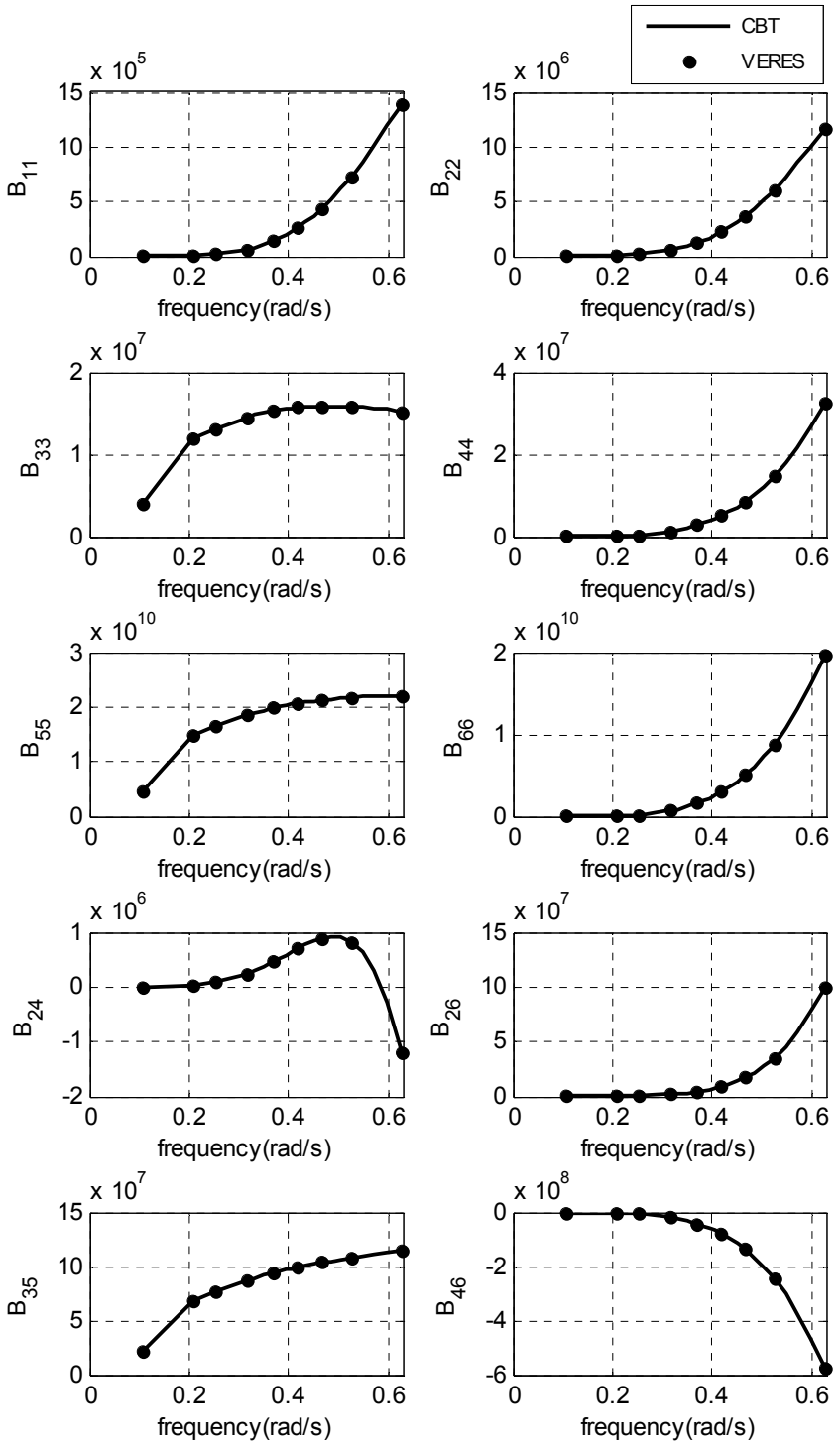


Figure 19: Potential damping B11, B22, B33, B44, B55, B66, B24, B26, B35, and B46 for the S-175 container ship at zero speed.

In the numerical calculation of current Chalmers Big Time Mobile, CBT-M, the linear Froude-Krylov force is used for a performance reason. In contrast, Chalmers Big Time source code fully implies the non-linear Froude-Krylov method. Response of S175 at zero speed with head on waves is shown in figure 20. The initial condition for the simulation is taken from a previous trail; heave motion 0.56 m and pitch motion 0.003 rad are used.

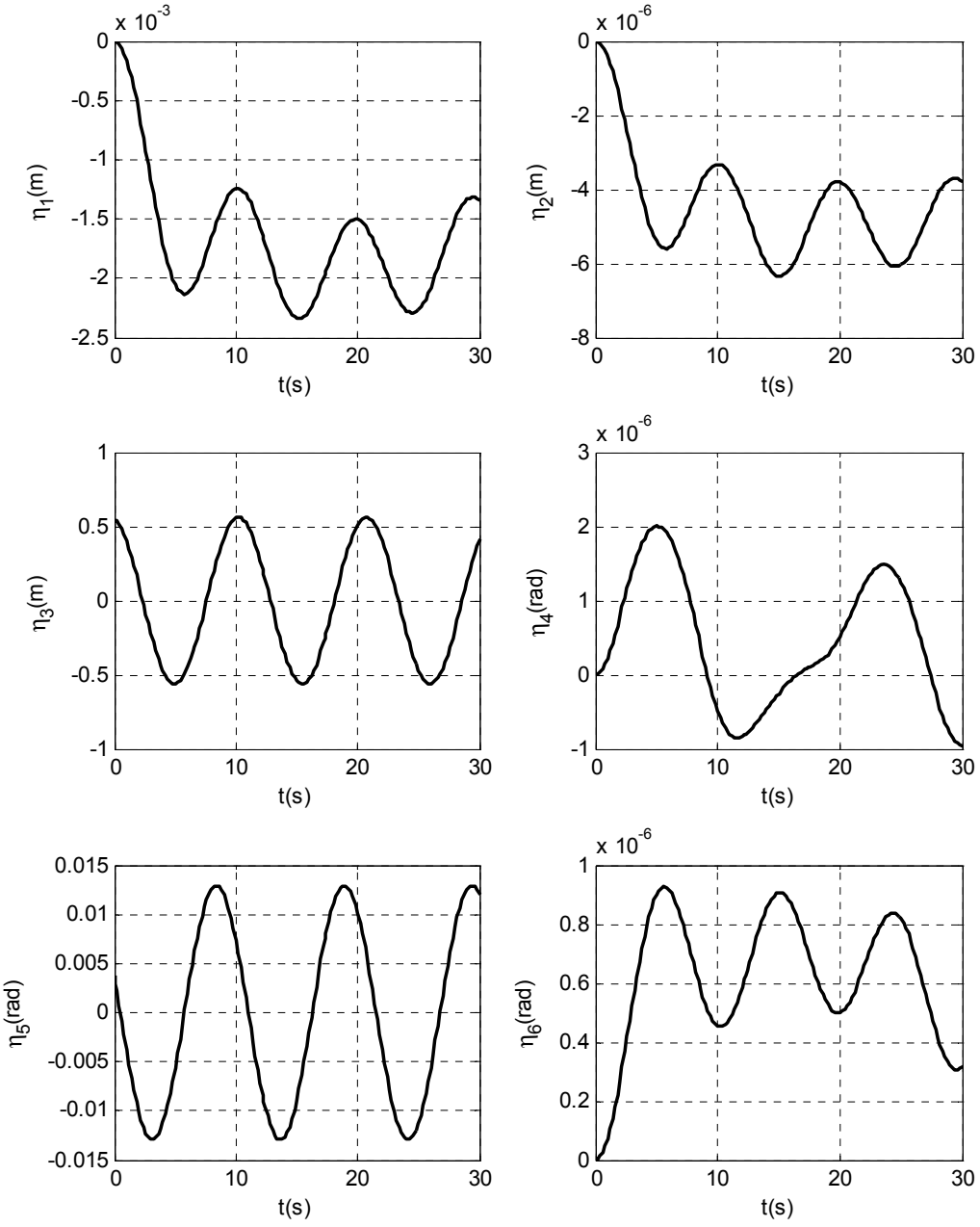


Figure 20: Chalmers Big Time's time domain numerical simulation results for S-175 at zero speed with zero degree head-on waves. Wave amplitude is 1m and peak frequency is 0.598 rad/s.

The results indicate that the ship has about 0.56m oscillating heave motion and 0.7 degree oscillating pitch motion. Very small surge, sway, roll, and yaw are negligible. They show that the heave and pitch motions follow the wave pattern as shown in figure 17. When the wave elevation is at the maximum, the heave motion reaches its peak that agrees with the theory. Since the wave length is slightly less than the ship length (98% of the ship length), 0 pitch angle occurs slightly after the wave elevation is at the maximum or minimum. Simulation agrees that there is no sway motion or rolling when a ship encounters with head on waves. However, there should be surge motion in the simulation. Further investigation is needed to validate the surge motion. The heave and pitch motion results are then validated against ShipX Vessel Responses (VERES) linear frequency domain strip solutions and shown in figure 21 and 22.

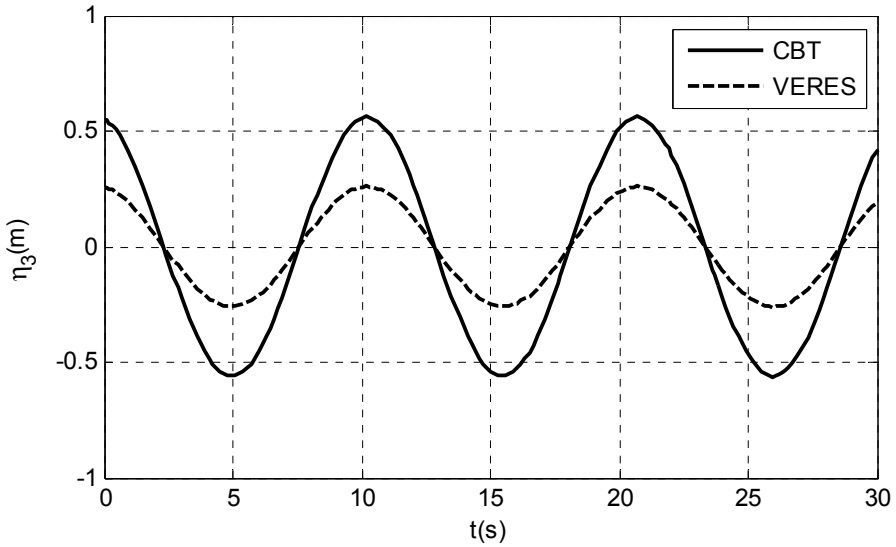


Figure 21: Comparison of S-175 heave motion time histories between Chalmers Big Time and VERES programs. S-175 is at zero speed with zero degree head-on waves.

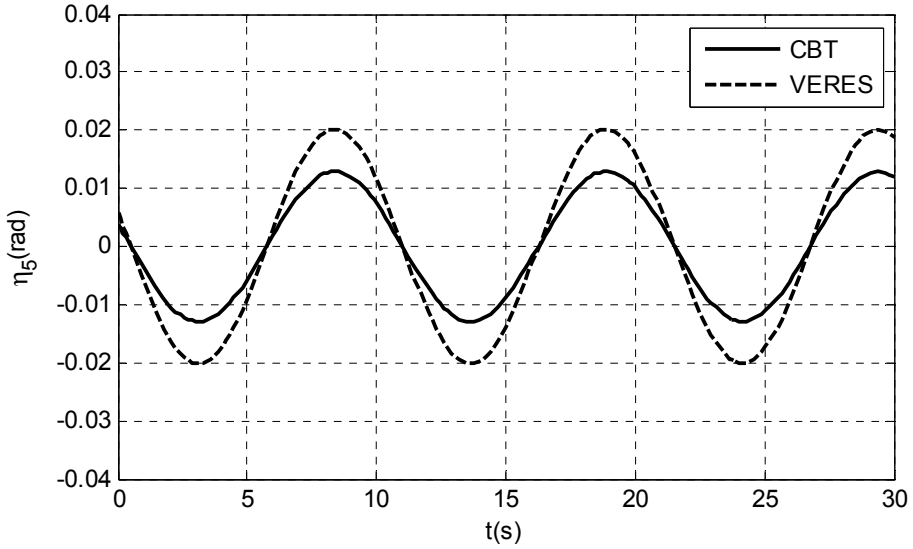


Figure 22: Comparison of S-175 pitch motion time histories between Chalmers Big Time and VERES programs. S-175 is at zero speed with zero degree head-on waves

The comparisons suggest that CBT over predicts the heave motion and captures less pitch response than the linear solutions of VERSE program. one of the possible reasons for discrepancies is that CBT cannot capture full Froude-Krylov effect from the current S-175 model used in the simulation. The model considers only the pressure points below the 9.5 m design draft, the draft used by VERSE. Thus, for 1 m amplitude wave model, the more accurate solution than the solutions in figure 21 and 22 may be achieved if 8.5 m draft is used instead.

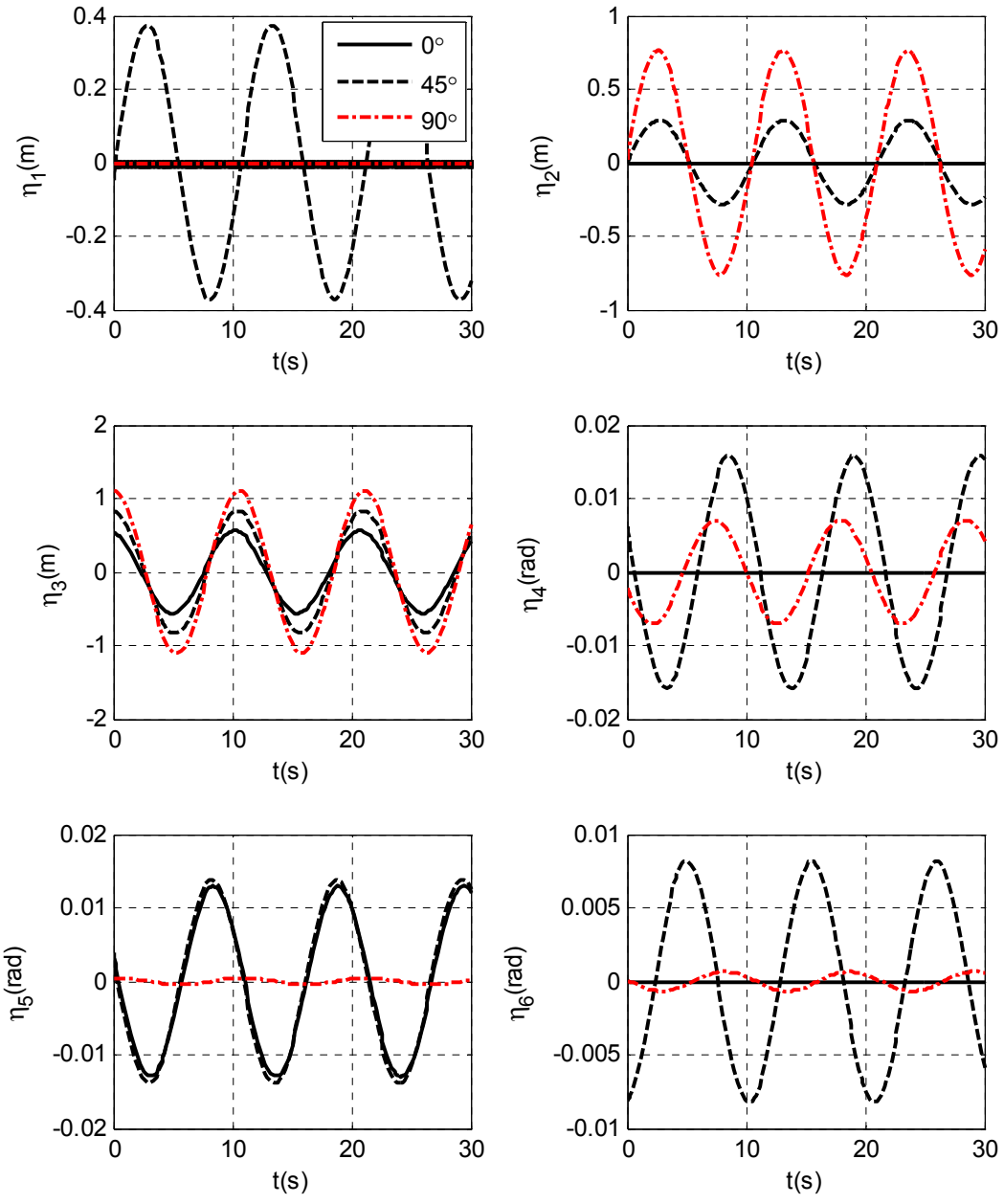


Figure 23: Simulation results for S-175 at zero forward speed with head sea 0 degree, bow sea 45 degree, and beam sea 90 degree.

The simulation is repeated with the starboard bow 45 degree and starboard abeam 90 degree wave directions. The comparison between the three sea states is shown in figure 23. Surge motion coming from water line Froude-Krylov forces is large enough to be seen in the 45 degree bow sea. Sway motions are found both in the bow and beam sea. The beam sea produces the largest 0.76m sway motion. The heave motion becomes stronger as the direction of wave increases. Peak heave motion of 1.1 m is computed in the beam sea. Roll motion in the bow sea is larger than in the beam Sea. The peak roll motion of 1 degree is computed in the bow sea while only 0.4 degree roll is found in the beam sea. Similar pitch motion is computed in the bow sea. The pitch motion in the beam sea is negligible. Similarly, the largest yaw motion 0.5 degree is computed in the bow sea. The comparison shows that the ship motion in six rigid body degrees of freedom are computed as expected.

4.5. Discussion

For the radiation problem, the accuracy of the numerical solution is estimated by observing the added masses and damping coefficients in the frequency domain. They are also checked for irregular behavior at the high (5 radian) and low (0.001 radian) frequencies. The computed coefficients are also compared with analytical results. The coefficient results from CBT indicate that the radiation forces computed and used in the equations are accurate. The viscous forces are treated using Ikeda's semi-empirical method. Some discrepancies in roll motion may have come from neglecting some of the components of viscous force in the approximation. Some of the data used in Ikeda's method are not taken from S-175. They are arbitrary and may affect the accuracy of the S-175 simulation.

A range of time steps is tested to achieve the computational criteria: the best solution and the shortest computation time. MATLAB has seven ODE solvers available for different numerical schemes. The Runge–Kutta fifth order one-step solver (MATLAB ODE45), the Adams-Bashforth-Moulton multistep solver (MATLAB ODE113), and the ODE solver based on the numerical, backward differentiation formulas (MATLAB ODE15s), and the delay differential equation solver (dde23) were used to compute the best time domain solution and are tested for the above criteria. This process showed that the first solver did not converge to a solution. The Adams-Bashforth-Moulton solver provided a good solution but took more computation time than the Runge–Kutta scheme. The stiff solver saved time but did not produce a solution as good as that produced by the ABM solver. Thus, for the more accurate solution, the Adams-Bashforth-Moulton solver is recommended; for faster analysis, the stiff ODE solver is recommended. However, all of the above solvers require significant additional computing time when computing the memory force, which requires inclusion of solutions from previous time steps. The DDE23 solver is thus more efficient for this project.

The results may be improved by increasing the number of strip used in computations and reducing the time and frequency step sizes. At this reporting time, reliable benchmark data and experimental results are not available for the researcher and much of the validation work is not completed. It is recommended to perform additional tests before using the current version of Chalmers Big Time program.

5. Conclusions

Chalmers Big Time MATLAB source codes are developed to simulate ship motions in six degrees of rigid body freedom. The simulation uses a time domain method applied to a ship travelling with a specified mean velocity in waves. It uses the impulse-response-function time domain method, in which the hydrodynamic coefficients are computed with a 2-D linear strip method in the frequency domain and converted into the time domain. MATLAB ODE solvers are compared, and the Adams-Bashforth-Moulton multistep solver is selected in the present project. Canonical forced and free motions are considered, and the equations of motion are solved simultaneously. A range of ship profiles, wave periods, and speeds are considered. The S175 container model and a box-shape hull are used to validate the source codes. Steady-state results are compared to other published results. The simulations agree with classical ship seakeeping response.

The current Chalmers Big Time MATLAB time domain simulation has several limitations. First, the simulation is intended for studying the motion of ships only. Other floating structures that are not shaped like a ship hull may not be used. The ship must be free floating; forces from mooring lines and risers are not included in the computations. Since hydrodynamic coefficients are computed in the frequency domain using strip theory, the ratio between the breadth and draft of the vessel should not be larger than 6. Therefore, the simulation does not favor very wide tanker hulls. To compute viscous damping in roll motion, Ikeda's prediction is used. The prediction limits the type of ship to be used in the simulation. Thus, motions of vessels with a buttock flow stern such as car-carriers and large passenger ships cannot be simulated in roll motion with high accuracy. Anti-roll tanks and other stabilizers are also not considered. No maneuvering simulation is available.

The present study contributes to time domain practical usage by introducing a touch-screen-based mobile device application. The intensive computation requirements of the time domain simulation and the lower capacities of current mobile devices prohibit implementation of the full six degrees of freedom time domain simulation. However, an application that reads a frequency domain pre-processed file generated by MATLAB and simulates heave, roll, and pitch motions has been developed as an outcome of this project. The study also provides an open source MATLAB code for the impulse-response function time domain analysis.

In the future, the linear strip method should be replaced with a nonlinear one, which may be similar to the sectional nonlinear method. The nonlinear strip method needs to be further developed to overcome its low frequency limitation as well as its requirement for slenderness. The DDE solver should be rewritten to customize time domain computation. In time, this weak nonlinear method should be completely replaced by a fully nonlinear CFD method. The computation routines in source code should be revised so that computation time may be reduced significantly. For faster analysis, the stiff ODE solver should also be considered.

Development of a touch-screen-based mobile device application should take advantage of technological advances. Its user interface must be developed to the degree that ship models and plots can be displayed. Currently, running the CBT on an iOS system slows down its multitasking performance. Perhaps the standalone file exchange design should be replaced with a cloud computing solution to achieve better computing performance. The limitations

described previously should also be addressed. Simulated results from institutions should be explored, and the program should be validated intensively.

The findings show that it is feasible to develop mobile device applications for a seakeeping analysis. In general, the current Chalmers Big Time MATLAB simulation and mobile application, with their underlying numerical analysis, provide a flexible tool that can be practically applied to current problems in seakeeping. They can also be extended to increasingly complex problems in naval architecture.

6. References

[1] Bergdahl, Lars. 2010 *Wave-induced loads and ships motions*. Göteborg, Sweden: Chalmers University of Technology

[2] Couser, P. 2000. Seakeeping Analysis for Preliminary Design, Formation Design Systems. UK

[3] Cummins, W.E. 1962. The Impulse Response Function and Ship Motions, Symposium on Ship Theory, Institüt für Schiffbau der Universität Hamburg, Hamburg, Germany, January 1962, Schiffstechnik, 9, 101-109

[4] Faltinsen, O.M. 1990*. Sea Loads on Ships and Offshore Structures.* Cambridge U.P.

[5] Hua, Palmquist.1995. *A Description of SMS- A Computer Code for Ship Motion Calculation*. Stockholm, Sweden: KTH

[6] Ikeda, Y. and etc. 2009. A Simple Prediction Formula of Roll Damping of Conventional Cargo Ships on the Basis of Ikeda's Method and Its Limitation, Proceedings of the 10th International Conference on Stability of Ships and Ocean Vehicles.

[7] ITTC. 2011. Recommended Procedures and Guidelines: Verification and Validation of Linear and Weakly Nonlinear Seakeeping Computer Codes , ITTC

[8] Journee J.M.J. 1993. Hydromechanic Coefficients for Calculating Time Domain Motions of Cutter Suction Dredges by Cummins Equations. Report No. 0968. Delft, Netherlands: Delft University of Technology

[9] Journee J.M.J.and and Adegeest L.J.M. 2003. Theoretical Manual of "SEAWAY for windows. Report No 1370. Delft, Netherlands: Delft University of Technology.

[10] Kring, David C.1994. Time Domain Ship Motions by a Three-Dimensional Rankine Panel Method. Ph.D. Thesis. Massachusetts Institute of Technology

[11] Lewis, Rory. 2011. *iPhone and iPad Apps for Absolute Beginners, iOS 5 Edition* 2nd Edition, New York :Apress

[12] Ovegård, Erik.2009. Numerical Simulation of Parametric Rolling in Waves. Master's Thesis. Royal Institute of Technology Centre for Naval Architecture.

[13] Renilson, M.R. 1982, An Investigation into the Factors Affecting the Likelihood of Broaching-to in Following Seas. *Proceedings of the 2nd International Conference on Stability and Ocean Vehicles*, 551-564. Tokyo, Japan: The Society of Naval Architects of Japan.

[14] Shampine L.F. and S. Thompson, Solving DDEs in Matlab, manuscript.

[15] Tasai, F. 1959. On the Damping Force and Added Mass of Ships Heaving and Pitching. Report No 26, Vol. VII. Fukuoka, Japan: Research Institute for Applied Mechanics, Kyushu University.

[16] Ursell, F. 1949. On the Heaving Motion of a Circular Cylinder on the Surface of a Fluid. *Quarterly Journal of Mechanics and Applied Mathematics*. Vol. II.

[17] Wehausen, J. V. 1971. The motion of floating bodies. *Ann. Rev. Fluid Mech*., 3:237_268

[18] "Mobile/Tablet Top Operating System Share Trend." netmarketshare.com. Net Applications, n.d. Web. 24 Aug. 2011.

Appendix A: Supplementary Notes in Computing Restoring and Excitation Forces

The present part is largely borrowed from Hua's Description of SMS. [5]

Computing Restoring and Froude-Krylov Forces

The vectors and Euclidian length in restoring and Froude-Krylov forces equation can be described in figure A1.

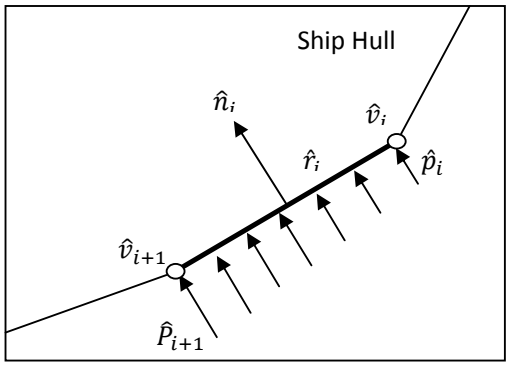


Figure A1: Illustration of Pressure and associated vectors in a segment between two consecutive points along the cross section hull

The Euclidian length of the segment i is defined as

$$l_i = \|\hat{v}_{i+1} + \hat{v}_i\|_2 \quad \ldots (A1)$$

where $\hat{v}_i$is the vector to the offset point i. The position vector is expressed in two ways based on Pressure value. When the addition of two consecutive pressures at each end of the segment i is zero, the position vector is

$$\hat{r}_i = \hat{v}_i + \frac{\frac{2}{3}P_{i+1}+\frac{1}{3}P_i}{P_{i+1}+P_i}.(\hat{v}_{i+1} - v_i) \quad \ldots (A2)$$

If the addition is zero, the position vector then is half of the difference between the vectors at each end.

$$\hat{r}_i = \frac{.(\hat{v}_{i+1}-v_i)}{2} \quad \ldots (A3)$$

Computing Diffraction Forces

To compute diffraction forces, particle velocities and accelerations in the y and z directions are required. These velocities and accelerations can be derived from wave potential ϕ and shown in equation A4 and A5 respectively.

$$v_y = \left[\frac{gak}{\omega}.\frac{\cosh(k.(Z+d)}{\cosh(k.d)}.\cos(k.X + \omega.t + \varepsilon) + U\right]\sin(\mu)$$

$$v_z = -\frac{gak}{\omega}.\frac{\sinh(k.(Z+d))}{\cosh(k.d)}.\sin(k.X + \omega.t + \varepsilon) \quad \ldots (A4)$$

$$a_y = \left[-gak.\frac{\cosh\left(k.(Z+d)\right)}{\cosh(k.d)}.\sin(k.X+\omega.t+\varepsilon)\right]\sin(\mu)$$

$$a_z = -gak.\frac{\sinh\left(k.(Z+d)\right)}{\cosh(k.d)}.\cos(k.X+\omega.t+\varepsilon) \quad \ldots \text{(A5)}$$

where U is the forward ship's speed, ε is phase angle, and μ is wave angle.

Appendix B: Basic Formulae for Viscous Damping Computation

The present part is largely borrowed from Ikeda's roll prediction method. [6]

Computing Frictional Component, B^F

The Friction damping at F_n=0 can be expressed as

$$B^F = \frac{4}{3\pi}\rho . s_f . r_f^3 \varphi_a \omega . c_f \quad \ldots (B1)$$

where c_f is frictional coefficient, $r_f$is average radius from the axis of rolling, s_f is wetted surface area., φ_a is roll amplitude. The frictional coefficient is

$$c_f = 1.328\sqrt{\frac{3.22\, r_f^2 . \varphi_a^2}{T_R . \upsilon}} \quad \ldots (B2)$$

The average radius can be computed from

$$r_f = \frac{(0.887+0.145\, C_b)(1.7\, T + C_b . B) - 2OG}{\pi} \quad \ldots (B3)$$

and wetted surface is simply

$$s_f = L_{pp}(1.75\, T + C_b B) \quad \ldots (B4)$$

where $T_R$is roll period, T is draft, υ is dynamic coefficient of viscosity, and OG is the distance from calm water surface to the axis of rolling (downward direction is positive in this equation.)

Computing Wave Component, B^W

The computed is computed as

$$B^W = \frac{A_1}{\hat{\omega}} \exp\left(\frac{-A_2(\log(\hat{\omega}) - A_3)^2}{1.44}\right) \quad \ldots (B5)$$

where coefficients A_1, A_2, A_3, and circular frequency $\hat{\omega}$ are defined below.

$$\hat{\omega} = \omega\sqrt{\frac{B}{2g}} \quad \ldots (B6)$$

$$A_1 = (A_{11}x_2^2 + A_{12}x_2 + A_{13})AA_1 \quad \ldots (B7)$$

$$A_2 = -1.402\, x_2^3 + 7.189\, x_2^4 - 10.993\, x_2 + 9.45 \quad \ldots (B8)$$

$$A_3 = A_{31}x_2^6 + A_{32}x_2^5 + A_{33}x_2^4 + A_{34}x_2^3 + A_{35}x_2^2 + A_{36}x_2 + A_{37} + AA_3 \quad \ldots (B9)$$

$x_1 = B/T$, $x_2 = 1 - OG/T$, and other coefficients are given below:

$$AA_1 = (AA_{11}x_3 + AA_{12})(1 - x_2) + 1$$

$$AA_3 = AA_{31}\begin{pmatrix} -1.05584\,x_3^9 + 12.688\,x_3^8 - 63.70534\,x_3^7 + 172.84571\,x_3^6 - 274.05701\,x_3^5 \\ +257.68705x_3^4 - 141.40915x_3^3 + 44.13177x_3^2 - 7.1654x_3 - 0.0495\,x_1^2 \\ +0.4518\,x_1 - 0.61655 \end{pmatrix}$$

$$AA_{31} = (-0.3767x_1^3 + 3.39\,x_1^2 - 10.356x_1 + 11.588)AA_{311}$$

$$AA_{32} = -0.0727\,x_1^2 + 0.7x_1 - 1.2818$$

$$AA_{311} = \left(-17.102\,C_b^3 + 41.495\,C_b^2 - 33.234C_b + 8.8007\right)x_{2+}$$

$$36.566\,C_b^3 - 89.203\,C_b^2 + 71.8C_b - 18.108$$

$$AA_{31} = -7686.0287\,C_b^6 + 30131.5678\,C_b^5 — 49048.9664C_b^4 + 42480.7709\,C_b^3$$

$$-20665.147\,C_b^2 + 5355.2035C_b - 577.8827$$

$$AA_{32} = 61639.9103C_b^6 - 241201.0598\,C_b^5 + 392579.5937C_b^4 - 340629.4699\,C_b^3$$

$$+166348.6917\,C_b^2 - 43358.7938C_b + 4714.7918$$

$$AA_{33} = -130677.4903C_b^6 + 507996.2604\,C_b^5 — 826728.7127C_b^4 + 722677.104\,C_b^3$$

$$-358360.7392\,C_b^2 + 95501.4948C_b - 10682.8619$$

$$AA_{34} = -110034.6584C_b^6 + 446051.22\,C_b^5 — 724186.4643C_b^4 + 599411.9264\,C_b^3$$

$$-264294.7189C_b^2 + 58039.7328C_b - 4774.6414$$

$$AA_{35} = 709672.0656C_b^6 - 2803850.2395C_b^5 + 4553780.5017C_b^4 - 3888378.9905\,C_b^3$$

$$+1839829.259\,C_b^2 - 457313.6939C_b + 46600.823$$

$$AA_{36} = -822735.9289C_b^6 + 3238899.7308\,C_b^5 — 5256636.5472C_b^4 + 4500543.147\,C_b^3$$

$$-2143487.3508C_b^2 + 538548.1194C_b - 55751.1528$$

$$AA_{37} = 299122.8727C_b^6 - 1175773.1606C_b^5 + 1907356.1357C_b^4 - 1634256.8172\,C_B^3$$

$$+780020.9393\,C_b^2 - 196679.7143C_b + 20467.0904$$

$$AA_{11} = AA_{111}C_b^2 + AA_{112}C_b + AA_{113}$$

$$AA_{12} = AA_{121}C_b^2 + AA_{122}C_b + AA_{123}$$

$$AA_{111} = 17.945x_1^3 - 166.294\,x_1^2 + 489.799x_1 - 493.142$$

$$AA_{112} = -25.507x_1^3 + 236.275\,x_1^2 - 698.683x_1 + 701.494$$

$$AA_{113} = 9.077x_1^3 - 84.332\,x_1^2 + 249.983x_1 - 250.787$$

$$AA_{121} = -16.872x_1^3 + 156.399\,x_1^2 - 460.689x_1 + 463.848$$

$$AA_{122} = 24.015x_1^3 - 222.507\,x_1^2 + 658.027x_1 - \ 660.665$$

$$AA_{123} = -8.56x_1^3 + 78.549\,x_1^2 - 235.827x_1 + 236.579$$

$$A_{11} = A_{111}C_b^2 + A_{112}C_b + \ A_{113}$$

$$A_{12} = A_{121}C_b^3 + A_{122}C_b^2 + A_{123}C_b + \ A_{124}$$

$$A_{13} = A_{131}C_b^3 + A_{132}C_b^2 + A_{133}C_b + \ A_{134}$$

$$A_{111} = -0.002222x_1^3 + 0.040871\,x_1^2 - 0.286866x_1 + 0.599424$$

$$A_{112} = 0.010185x_1^3 - 0.161176\,x_1^2 + 0.904989x_1 - 1.641389$$

$$A_{113} = -0.015422x_1^3 + 0.220371\,x_1^2 - 1.084987x_1 + 1.834167$$

$$A_{121} = -0.0628667x_1^4 + 0.4989259x_1^3 + 0.52735\,x_1^2 - 10.7918672x_1 + 16.616327$$

$$A_{122} = 0.1140667x_1^4 - 0.8108963x_1^3 - 2.2186833\,x_1^2 + 25.1269741x_1 - 37.7729778$$

$$A_{123} = -0.0589333x_1^4 + 0.2639704x_1^3 + 3.1949667\,x_1^2 - 21.8126569x_1 + 31.4113508$$

$$A_{124} = 0.0107667x_1^4 + 0.0018704x_1^3 - 1.2494083\,x_1^2 + 6.9427931x_1 - 10.2018992$$

$$A_{131} = 0.192207x_1^3 - 2.787462\,x_1^2 + 12.507855x_1 - 14.764856$$

$$A_{132} = -0.350563x_1^3 + 5.222348x_1^2 - 23.974852x_1 + 29.007851$$

$$A_{133} = 0.237096x_1^3 - 3.535062\,x_1^2 + 16.368376x_1 - 20.539908$$

$$A_{134} = -0.067119x_1^3 + 0.966362x_1^2 - 4.407535x_1 + 5.894703$$

$$x_6 = x_4 - AA_{32}$$

… (B10)

This prediction and empirical formulae may be used if the requirements listed in table B-1 are met.

Table B-1: Requirements for Using Wave and Eddy Components Predictions

Parameter	Requirement
Block Coefficient, C_b	$0.5 \leq C_b \leq 0.85$
Breadth to Draft Ratio, B/T	$2.5 \leq \mathrm{B/T} \leq 4.5$
Circular Frequency, $\hat{\omega}$	$\hat{\omega} \leq 1.0$
VCG to draft Ratio, OG/T	$-1.5 \leq \mathrm{OG/T} \leq 0.2$
Mid-ship section Coefficient, C_m	$0.9 \leq C_m \leq 0.99$

Computing Eddy Component, B^E

The eddy component can be predicted as follow:

$$B^E = \frac{4\,.\hat{\omega}\varphi_a}{3\,\pi\,x_2.x_1^3}\,C_R \qquad \ldots \text{(B11)}$$

where

$$C_R = A_E \exp\,(B_{E1} + B_{E2}.\,x_3^{B_{E3}}) \qquad \ldots \text{(B12)}$$

and the coefficients are

$$A_E = (-0.0182\,C_b + 0.0155).\,(x_1 - 1.8)^3 - 79.414C_b^4 + 215.695C_b^3$$
$$-215.883C_b^2 + 93.894C_b - 14.848$$
$$B_{E1} = (-0.2\,x_1 + 1.6).\,(3.98C_b - 5.1525).\,x_2 - \{0.9717C_b^2 - 1.55C_b$$
$$+0.723).\,x_4 + (0.04567C_b + 0.9408)\}$$
$$B_{E2} = (0.25\,x_2 + 0.95).\,x_2 - 219.2C_b^3 + 443.7C_b^2 - 283.3C_b + 59.6$$
$$B_{E3} = (46.5\ + 15.\,x_1).\,C_b + 11.2\,x_1 - 28.6$$

… (B13)

while $x_1$and x_2 are given in above wave component part. The requirements for using this prediction method are listed in table B-1.

Computing Bilge Keel Roll Damping Component, B^{BK}

Similarly, the eddy component can be found as follow:

$$B^{BK} = A_{BK}.\exp(B_{BK1} + B_{BK2}\,.\,x_3^{BK3})\,.\,x_5 \qquad \ldots \text{(B14)}$$

where

$$A_{BK} = f_1(x_1, x_2).\,f_2(x_6).\,f_3(x_7 + x_8) \qquad \ldots \text{(B15)}$$

and the coefficients are

$$f_1 = (-0.3651\,C_b + 0.3907).\,(x_1 - 2.83)^2 - 2.21C_b + 2.632$$
$$f_2 = 0.00255\varphi_a^2 + 0.122x_6 - 0.4794$$
$$f_3 = (-0.8913\,x_4^2 - 0.0733x_4).\,x_5^2 + (5.2857\,x_4^2 - 0.01185x_4 + 0.00189).\,x_5$$
$$B_{BK1} = (5x_4 + 0.3\,x_1 - 0.2\,x_5 + 0.00125\ \varphi_a^2 - 0.0425\varphi_a - 1.86).\,x_2$$
$$B_{BK2} = -15x_4 + 1.2\,C_b - 0.1\,x_1 - 0.0657\ x_2^2 - 0.0586x_2 - 1.6164$$
$$B_{BK3} = (2.5\,x_2 + 15.75)$$

… (B16)

x_4 is ratio between width of the bilge keel and ship breadth while x_5 is ratio between length of the bilge keel and ship L_{pp}. This prediction and empirical formulae may be used if the requirements listed in table B-2 in addition to table B-1 requirements are met.

Table B-1: Additional Requirements for Using Bilge Keel components Prediction

Parameter	Requirement
width to Beam ratio, $^{b_{BK}}/_{B}$	$0.01 \leq {}^{b_{BK}}/_{B} \leq 0.06$
length to L_{pp} ratio, $^{l_{BK}}/_{\mathrm{Lpp}}$	$0.05 \leq {}^{l_{BK}}/_{\mathrm{Lpp}} \leq 0.4$

Appendix C: Determining Added Mass and Damping Coefficients of a Ship

Ursell[14] and Tasai[13] Methods are used in CBT to compute hydrodynamic coefficients. They are detailed in this section.

Conformal Mapping

To use Ursell's method for determining potential coefficients, the ship cross section have to be mapped conformally to a unit circle which the method is based upon. The general transformation formula between the ship cross sectional plane and plane of the unit circle is given by:

$$y + iz = M_S \sum a_{2n-1} \zeta^{2n-1} \quad \ldots \text{(C1)}$$

where the left hand side of the equation represents the plane of the ship cross section, ζ^{2n-1} on the left hand side is the plane of the unit circle, $M_S$is the scale factor, and a_{2n-1} are conformal mapping coefficients. Therefore, the relationships between the coordinates in the two planes can be defined as

$$y = -M_S \sum (-1)^n a_{2n-1} \sin((2n-1)\theta) \quad \ldots \text{(C2)}$$

$$z = M_S \sum (-1)^n a_{2n-1} \cos((2n-1)\theta) \quad \ldots \text{(C3)}$$

where the scale factor is.

$$M_S = \frac{\frac{B}{2}}{\sum_{n=0}^{N} a_{2n-1}} \quad \ldots \text{(C4)}$$

The value of M_s together with a_{2n-1} values represent any complex hull geometry form and thus are useful in Tasai's theory which will be discussed later in this section. The first guess for these coefficients is done with the Lewis conformal mapping and then an iteration loop is used to achieve a correct representation of the ship cross section. N value for Lewis conformal mapping is 3 while 10 is recommended for Close-Fit Conformal mapping. First, the half breadth to draught ratio H_0 and the sectional area coefficient σ_S are defined:

$$H_0 = \frac{\frac{B}{2}}{T} = \frac{1 + a_1 + a_3}{1 - a_1 + a_3} \quad \ldots \text{(C5)}$$

$$\sigma_S = \frac{A_S}{B\,T} = \frac{\pi}{4} \frac{1 - {a_1}^2 - 3{a_3}^2}{(1 + a_3)^2 - {a_1}^2} \quad \ldots \text{(C6)}$$

where A_S is the area of the cross section. However, there are some limitations for using this method. The boundaries for the acceptable Lewis conventional, combined bulbous and tunneled forms are given below:

For $H_0 \leq 1$

$$\frac{3\pi}{32}(2 - H_0) < \sigma_S < \frac{\pi}{32}(10 + H_0 + \frac{1}{H_0}) \quad \text{... (C7)}$$

For $H_0 \leq 1$

$$\frac{3\pi}{32}(2 - \frac{1}{H_0}) < \sigma_S < \frac{\pi}{32}(10 + H_0 + \frac{1}{H_0}) \quad \text{... (C8)}$$

These boundaries are described in figure C-1.

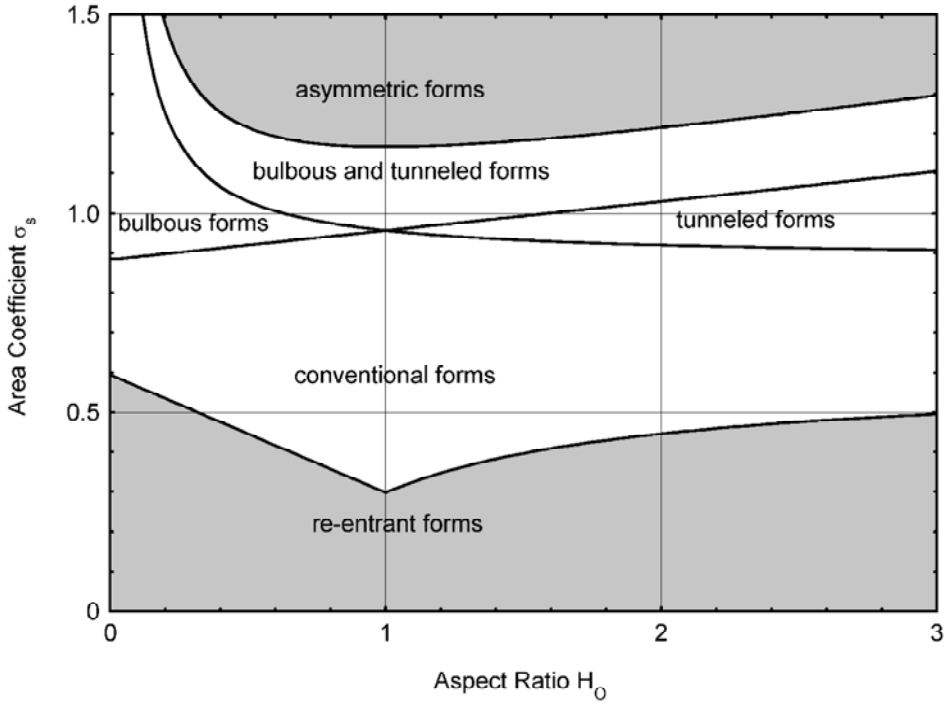


Figure C1: Range of H_0 and $\sigma_S$of Lewis Forms [9]. If the sectional area coefficient is outside of the range, the closest value inside the range is applied.

. By using the equations C5 and C6, the following quadratic equation is derived

$$c_1 {a_3}^2 + c_2 a_3 + c_3 = 0 \quad \text{... (C9)}$$

where:

$$c_1 = 3 + \frac{4\sigma_S}{\pi} + \left(1 - \frac{4\sigma_S}{\pi}\right)\left(\frac{H_0 - 1}{H_0 + 1}\right)^2$$

$$c_2 = 2c_1 - 6$$

$$c_3 = c_1 - 4$$

Then the solutions for a_1 and a_3 are:

$$a_3 = \frac{-c_1+3+\sqrt{9-2c_1}}{c_1} \quad \text{... (C10)}$$

$$a_1 = \left(\frac{H_0-1}{H_0+1}\right)(a_3+1) \quad \text{... (C11)}$$

For the first guess, a_2 and all the other coefficients a_{2n-1} are equal to zero. Then, the angles θ_i is estimated using the guessed conformal mapping coefficients. With the new θ_i, the first approximation of the contour of the hull is achieved as follow:

$$y_i = M_S \sum(-1)^n a_{2n-1} \sin((2n-1)\theta_i) \quad \text{... (C12)}$$

$$z_i = M_S \sum(-1)^n a_{2n-1} \cos((2n-1)\theta_i) \quad \text{... (C13)}$$

Then, the deviation between the approximated contour and the real contour of the section (y_{0i},z_{0i}) is given as

$$e_i = (y_{0i} - y_i)^2 + (z_{0i} - z_i)^2 \quad \text{... (C14)}$$

Thus, the sum of the squares of deviations is:

$$E_{tot} = \sum_{i=0}^{I} e_i \quad \text{... (C15)}$$

Using the new $\theta_{i,}$, M_s and a_{2n-1} are recomputed to minimized the sum of the squares of deviations.

$$\frac{\partial E}{\partial(M_S a_{2j-1})} = 0 \;\; for\; j = 0 \ldots N \quad \text{... (C16)}$$

Then, N+ 1 equation is:

$$\sum_{n=0}^{N}(-1)^n[M_S a_{2n-1}] \sum_{i=0}^{I} \cos((2j-2n)\theta_i = \sum_{i=0}^{I} -x_{0i} \sin((2j-1)\,\theta_i) + y_{0i}\cos((2j-1)\theta_i) \quad \text{... (C17)}$$

To achieve the exact breadth and draught coordinates, the last two line of the N+1 equation C17 is replaced by C18 and C19 the equations for the breadth at the water line and draught.

$$\sum_{n=0}^{N}(-1)^n[M_S a_{2n-1}] = T_S \quad \text{... (C18)}$$

$$\sum_{n=0}^{N}[M_S a_{2n-1}] = \frac{B_S}{2} \quad \text{... (C19)}$$

The new M_s and a_{2n-1} are solved using equations C17, C18, and C19. Please note that M_S has to be found by dividing the final solutions [M_S a_{2n-1}] by the old M_S value. Once again the new angles θ_i are estimated. The iteration of finding θ_i, M_s and a_{2n-1} will continue until the difference between two values of E is small as given in equation C20 Journée [9]

$$\Delta E = (I+1)(0.00005\sqrt{{b_{max}}^2 + {d_{max}}^2} \quad \ldots(C20)$$

Computing 2-D Potential Coefficients

The 2-D potential coefficients can be computed empirically using Tasai's theory for deep water, Keil's theory for shallow and deep water, or Frank's theory for deep water. CBT uses the Tasai's theory which is based on Ursell's potential theory for circular cylinders and N-parameter conformal mapping. This portion is largely borrowed from Journée's SEAWAY theory manual. [9]

Heaving Motion and its Coefficients

In the heaving motion, as described in equation 22 and 23, the coefficients A_0, B_0, M_0, and N_0 are required to compute sectional potential coefficients a_{33} and b_{33}: A brief procedure of finding A_0, B_0, M_0, and N_0 is described below.

$$\phi_m = \cos(2m\theta) + \frac{\xi_0}{1+\sum_{n=1}^{N} a_{2n-1}}\left(\frac{\cos(2m-1)\,\theta}{2\mathrm{m}-1} + \sum_{n=1}^{N}(-1)^n \frac{(2n-1)}{(2m+2n-1)} a_{2n-1}\cos(2m+2n-1)\,\theta\right)$$

$$\Psi_m = \sin(2m\theta) + \frac{\xi_0}{1+\sum_{n=1}^{N} a_{2n-1}}\left(\frac{\sin(2m-1)\,\theta}{2\mathrm{m}-1} + \sum_{n=1}^{N}(-1)^n \frac{(2n-1)}{(2m+2n-1)} a_{2n-1}\sin(2m+2n-1)\,\theta\right)$$

$$\varphi_C = \pi e^{-kz}\cos\frac{\omega^2}{g}y$$

$$\psi_C = \pi e^{-kz}\sin\frac{\omega^2}{g}|y|$$

$$\varphi_S = \pi e^{-kz}\sin\frac{\omega^2}{g}|y| + (Q cos\frac{\omega^2}{g}y + (S-\pi)\sin\frac{\omega^2}{g}y)e^{-kz}$$

$$\psi_S = -\pi e^{-kz}\cos\frac{\omega^2}{g}|y| + (Q sin\frac{\omega^2}{g}y - (S-\pi)\cos\frac{\omega^2}{g}y)e^{-kz}$$

Then, the Porter series is:

$$Q = \mathrm{Y} + \ln\left(\frac{\omega^2}{g}\sqrt{y^2+z^2}\right) + \sum_{n=1}^{\infty} p_n \cos(n\beta)$$

$$S = \beta + \sum_{n=1}^{\infty} p_n \sin(n\beta)$$

$$\beta = \arctan\left(\frac{y}{z}\right)$$

$$p_n = \frac{\left(\frac{\omega^2}{g}\sqrt{y^2+z^2}\right)^n}{n\,n!}$$

$$\mathrm{Y} = 0.57722$$

p_m amd q_m are calculated as follows:

$$\sum_{m=1}^{M} p_m \int_0^{\frac{\pi}{2}} f_{2m}(\theta)\, f_{2n}(\theta) d\theta = \int_0^{\frac{\pi}{2}} \left(\psi_C(\theta) - h(\theta)\psi_C\left(\frac{\pi}{2}\right)\right) f_{2n}(\theta)\, d\theta$$

$$\sum_{m=1}^{M} q_m \int_0^{\frac{\pi}{2}} f_{2m}(\theta)\, f_{2n}(\theta) d\theta = \int_0^{\frac{\pi}{2}} \left(\psi_S(\theta) - h(\theta)\psi_S\left(\frac{\pi}{2}\right)\right) f_{2n}(\theta)\, d\theta$$

where

$$f_{2m}(\theta) = -\psi_m(\theta) + h(\theta)\psi_m\left(\frac{\pi}{2}\right)$$

$$h(\theta) = \frac{1}{1+\sum_{n=1}^{N} a_{2n-1}} \sum_{n=0}^{N} (-1)^n\, a_{2n-1} \sin(2n-1)\,\theta$$

The coefficient A_0 and B_0 can be described as:

$$A_0 = \psi_C\left(\frac{\pi}{2}\right) + \sum_{m=1}^{M} p_m \psi_m\left(\frac{\pi}{2}\right)$$

$$B_0 = \psi_S\left(\frac{\pi}{2}\right) + \sum_{m=1}^{M} q_m \psi_m\left(\frac{\pi}{2}\right)$$

Finally, M_0 and N_0 are computed as:

$$M_0 = -\int_0^{\frac{\pi}{2}} \frac{\varphi_S}{1+\sum_{n=1}^{N} a_{2n-1}} \sum_{n=0}^{N} (-1)^n (2n-1)\, a_{2n-1} \cos(2n-1)\,\theta$$
$$- \sum_{m=1}^{M} \frac{(-1)^m q_m}{1+\sum_{n=1}^{N} a_{2n-1}} \sum_{n=0}^{N} \frac{(2n-1)^2}{4m^2-(2n-1)^2} a_{2n-1} + \frac{4\pi}{\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^2} (q_1$$
$$+ \sum_{m=1}^{M} (-1)^m q_m \sum_{n=0}^{N-m} (2n-1) a_{2n-1} a_{2m+2n-1}$$

$$N_0 = -\int_0^{\frac{\pi}{2}} \frac{\varphi_C}{1+\sum_{n=1}^{N} a_{2n-1}} \sum_{n=0}^{N} (-1)^n (2n-1)\, a_{2n-1} \cos(2n-1)\,\theta$$
$$- \sum_{m=1}^{M} \frac{(-1)^m p_m}{1+\sum_{n=1}^{N} a_{2n-1}} \sum_{n=0}^{N} \frac{(2n-1)^2}{4m^2-(2n-1)^2} a_{2n-1} + \frac{\xi_b \pi}{4\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^2} (p_1$$
$$+ \sum_{m=1}^{M} (-1)^m p_m \sum_{n=0}^{N-m} (2n-1) a_{2n-1} a_{2m+2n-1}$$

Roll Motion and its Coefficients

In the roll motion, as described in equation 22 and 23, the coefficients P_0, Q_0, X_0, and Y_0 are required to compute the sectional potential coefficients a_{44} and b_{44}: A brief procedure of finding P_0, Q_0, X_0, and Y_0 is described below.

$$\phi_m = \sin((2m+1)\theta) - \frac{\xi_0}{1+\sum_{n=1}^{N} a_{2n-1}}\left(\frac{\sin(2m)\,\theta}{2\mathrm{m}} + \sum_{n=1}^{N}(-1)^n \frac{(2n-1)}{(2m-2n)} a_{2n-1} \sin(2m+2n)\,\theta\right)$$

$$\Psi_m = -\cos((2m+1)\theta) + \frac{\xi_0}{1+\sum_{n=1}^{N} a_{2n-1}}\left(\frac{\cos(2m)\,\theta}{2\mathrm{m}} + \sum_{n=1}^{N}(-1)^n \frac{(2n-1)}{(2m-2n)} a_{2n-1} \cos(2m+2n)\,\theta\right)$$

$$\varphi_C = \mp\pi e^{-kz} \sin\frac{\omega^2}{g}|y|$$

$$\psi_C = \pi e^{-kz} \cos\frac{\omega^2}{g} y$$

$$\varphi_S = \pm\pi e^{-kz} \cos\frac{\omega^2}{g} y \mp \left(Q sin\frac{\omega^2}{g} y - (S-\pi)\cos\frac{\omega^2}{g} y\right) e^{-kz} \pm \frac{|y|}{\frac{\omega^2}{g}(z^2+y^2)}$$

$$\psi_S = \pi e^{-kz} \cos\frac{\omega^2}{g}|y| + \left(Q cos\frac{\omega^2}{g} y + (S-\pi)\sin\frac{\omega^2}{g} y\right) e^{-kz} - \frac{z}{\frac{\omega^2}{g}(z^2+y^2)}$$

p_m amd q_m are calculated as follows:

$$\sum_{m=1}^{M} p_m \int_0^{\frac{\pi}{2}} f_{2m}(\theta)\, f_{2n}(\theta) d\theta = \int_0^{\frac{\pi}{2}} (\psi_C(\theta) - \psi_C\left(\frac{\pi}{2}\right)) f_{2n}(\theta)\, d\theta$$

$$\sum_{m=1}^{M} q_m \int_0^{\frac{\pi}{2}} f_{2m}(\theta)\, f_{2n}(\theta) d\theta = \int_0^{\frac{\pi}{2}} (\psi_S(\theta) - \psi_S\left(\frac{\pi}{2}\right)) f_{2n}(\theta)\, d\theta$$

where

$$f_{2m}(\theta) = -\psi_m(\theta) + \psi_m\left(\frac{\pi}{2}\right)$$

and

$$-\chi \frac{\pi \dot{\omega} b_0}{4g\eta_\alpha}(\mu(\theta)-1)(P_0 \cos(\omega t) + Q_0 \sin(\omega t))$$

Thus, P_0 and Q_0 are

$$P_0 = \frac{\pi \chi_a}{2\eta_a} \xi_b \sin(\gamma)$$

$$Q_0 = \frac{\pi \chi_a}{2\eta_a} \xi_b \cos(\gamma)$$

Then, Y_R and X_R are calculated:

$$
\begin{aligned}
&Y_R \\
&= \frac{1}{2\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^2} \int_0^{\frac{\pi}{2}} \varphi_S \sum_{n=0}^{N}\sum_{i=0}^{N} (-1)^{n+i}(2i-1)a_{2n-1}a_{2i-1}\sin\big((2n-2i)\theta\big)\,d\theta \\
&+ \frac{1}{2\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^2} \sum_{m=1}^{M} (-1)^m q_m \sum_{n=0}^{N}\sum_{i=0}^{N} \frac{(2i-1)(2n-2i)}{(2m+1)^2-(2n-2i)^2} a_{2n-1}a_{2i-1} \\
&- \frac{\pi\xi_0}{8\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^3} \sum_{m=1}^{N} (-1)^m q_m \left(\sum_{n=m}^{N}\sum_{i=0}^{n-m} \frac{(-2m+2n-2i-1)(2i-1)}{(2n-2i)} a_{2n-1}a_{2i-1}a_{-2m+2n-2i-1} \right. \\
&\left. + \sum_{n=0}^{N-m}\sum_{i=m+n}^{N} \frac{(-2m-2n+2i-1)(2i-1)}{(2n-2i)} a_{2n-1}a_{2i-1}a_{-2m-2n+2i-1} \right)
\end{aligned}
$$

$$
\begin{aligned}
&X_R \\
&= \frac{1}{2\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^2} \int_0^{\frac{\pi}{2}} \varphi_C \sum_{n=0}^{N}\sum_{i=0}^{N} (-1)^{n+i}(2i-1)a_{2n-1}a_{2i-1}\sin\big((2n-2i)\theta\big)\,d\theta \\
&+ \frac{1}{2\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^2} \sum_{m=1}^{M} (-1)^m p_m \sum_{n=0}^{N}\sum_{i=0}^{N} \frac{(2i-1)(2n-2i)}{(2m+1)^2-(2n-2i)^2} a_{2n-1}a_{2i-1} \\
&- \frac{\pi\xi_0}{8\left(1+\sum_{n=1}^{N} a_{2n-1}\right)^3} \sum_{m=1}^{N} (-1)^m p_m \left(\sum_{n=m}^{N}\sum_{i=0}^{n-m} \frac{(-2m+2n-2i-1)(2i-1)}{(2n-2i)} a_{2n-1}a_{2i-1}a_{-2m+2n-2i-1} \right. \\
&\left. + \sum_{n=0}^{N-m}\sum_{i=m+n}^{N} \frac{(-2m-2n+2i-1)(2i-1)}{(2n-2i)} a_{2n-1}a_{2i-1}a_{-2m-2n+2i-1} \right)
\end{aligned}
$$

Sway Motion and its Coefficients

In the sway motion, as described in equation 22 and 23, the coefficients M_0, N_0, P_0, and Q_0 are required to compute the sectional potential coefficients a_{22} and b_{22}: A brief procedure of finding M_0, N_0, P_0, and Q_0 is described below.

$$\phi_m = \mathrm{e}^{-(2\mathrm{m}+1)\alpha}\sin((2m+1)\theta) - \frac{\xi_b}{\sigma_a}\sum_{n=0}^{N}(-1)^n \frac{(2n-1)}{(2m+2n)} a_{2n-1}.\, e^{-(2m+2n)\alpha}\sin\left((2\mathrm{m}+2\mathrm{n})\theta\right)$$

$$\Psi_m = -\mathrm{e}^{-(2\mathrm{m}+1)\alpha}\cos((2m+1)\theta) + \frac{\xi_b}{\sigma_a}\sum_{n=0}^{N}(-1)^n \frac{(2n-1)}{(2m+2n)} a_{2n-1}.\, e^{-(2m+2n)\alpha}\cos\left((2\mathrm{m}+2\mathrm{n})\theta\right)$$

$$\varphi_C.j = -\pi e^{-\upsilon z}\sin(\upsilon x)$$

$$\psi_C = \pi e^{-\upsilon z}\cos(\upsilon x)$$

$$\varphi_S.j = \pi e^{-\upsilon z}\cos(\upsilon|y|)| - \int_0^{\alpha} \frac{\upsilon.cos(kz)+k.sin(kz)}{k^2+\upsilon^2} e^{-ky}dk - \frac{y}{\upsilon(y^2+z^2)}$$

$$\psi_S = -\pi e^{-\upsilon z}\sin(\upsilon|y|)| + \int_0^{\alpha} \frac{\upsilon sin(kz)-kcos(kz)}{k^2+\upsilon^2} e^{-kz}dk - \frac{y}{\upsilon(y^2+z^2)}$$

Then, the Porter series is:

$$Q = \mathrm{Y} + \ln\left(\upsilon\sqrt{y^2 + z^2}\right) + \sum_{n=1}^{\infty} p_n \cos(n\beta)$$

$$S = \beta + \sum_{n=1}^{\infty} p_n \sin(n\beta)$$

$$\beta = \arctan\left(\frac{y}{z}\right)$$

$$p_n = \frac{\left(\upsilon\sqrt{y^2 + z^2}\right)^n}{n\, n!}$$

$$\mathrm{Y} = 0.57722$$

p_m amd q_m are calculated as follows:

$$\sum_{m=0}^{M} p_m \int_0^{\frac{\pi}{2}} f_{2m}(\theta)\, f_{2n}(\theta) d\theta = \int_0^{\frac{\pi}{2}} \left(\psi_C(\theta) - \psi_C\left(\frac{\pi}{2}\right)\right) f_{2n}(\theta)\, d\theta$$

$$\sum_{m=0}^{M} q_m \int_0^{\frac{\pi}{2}} f_{2m}(\theta)\, f_{2n}(\theta) d\theta = \int_0^{\frac{\pi}{2}} \left(\psi_S(\theta) - \psi_S\left(\frac{\pi}{2}\right)\right) f_{2n}(\theta)\, d\theta$$

where

$$f_{2m}(\theta) = -\psi_m(\theta) + \psi_m\left(\frac{\pi}{2}\right)$$

The coefficient A_0 and B_0 can be described as:

$$-\chi \frac{\pi \dot{\omega} b_0}{4 g \eta_\alpha} (\mu(\theta) - 1)(P_0 \cos(\omega t) + Q_0 \sin(\omega t))$$

Thus, P_0 and Q_0 are

$$P_0 = \frac{-x_a \pi}{\eta_a} \xi_b \sin(\varepsilon)$$

$$Q_0 = \frac{-x_a \pi}{\eta_a} \xi_b \cos(\varepsilon)$$

Finally, M_0 and N_0 are computed as:

$$M_0 = -\frac{1}{\sigma_a} \int_0^{\frac{\pi}{2}} \phi_{B0_s(\theta)} \sum_{n=0}^{N} \{(-1)^n (2n-1)\, a_{2n-1} \sin(2n-1)\, \theta\} d\theta + \frac{\pi}{4\sigma_a} \sum_{m=1}^{N-1} \{(-1)^m Q_{2m} (2m+1) a_{2m+1}\}$$
$$+ \frac{\xi_b}{\sigma_a^2} \sum_{m=1}^{M} \{(-1)^m Q_{2m} \sum_{n=0}^{N} \sum_{i=0}^{N} \{\frac{(2n-1)(2i-1)}{(2m+2i)^2 - (2n-1)^2} a_{2n-1} a_{2i-1}\}$$

$N_0$is obtained by replacing ϕ_{BO_s} by ϕ_{BO_c} and Q_{2m} by P_{2m} in the above M_0 expression.

Coupling of Roll into Sway Motion and its Coefficients

In the coupling of roll into sway motion, as described in equation 22 and 23, the coefficients M_0, N_0, P_0, and Q_0 are required to compute the sectional potential coefficients a_{24} and b_{24}: These M_0, N_0, P_0, and Q_0 are already computed above.

Determining Added Mass and Damping Coefficients of a Ship with Forward Speed in Waves

As described by Hua, [5] Ship added masses and damping coefficients are calculated as:

$$A_{22} = \int_0^L a_{22}.dx - \frac{U}{\omega_e^2}\, b_{22}^A$$

$$A_{24} = \int_0^L a_{24}.dx - \frac{U}{\omega_e^2}\, b_{24}^A = A_{42}$$

$$A_{26} = \int_0^L x.a_{22}.dx + \frac{U}{\omega_e^2}\int_0^L b_{22}.dx - \frac{U}{\omega_e^2}\, x_a b_{22}^A + \frac{U^2}{\omega_e^2} a_{22}^A = A_{62}$$

$$A_{33} = \int_0^L a_{22}.dx - \frac{U}{\omega_e^2}\, b_{33}^A$$

$$A_{35} = -\int_0^L x.a_{33}.dx - \frac{U}{\omega_e^2}\int_0^L b_{33}.dx + \frac{U}{\omega_e^2}\, x_a b_{33}^A - \frac{U^2}{\omega_e^2} a_{33}^A$$

$$A_{44} = \int_0^L a_{44}.dx - \frac{U}{\omega_e^2}\, b_{44}^A$$

$$A_{46} = \int_0^L x.a_{24}.dx + \frac{U}{\omega_e^2}\int_0^L b_{24}.dx - \frac{U}{\omega_e^2}\, x_a b_{24}^A + \frac{U^2}{\omega_e^2} a_{24}^A$$

$$A_{53} = -\int_0^L x.a_{33}.dx + \frac{U}{\omega_e^2}\int_0^L b_{33}.dx + \frac{U}{\omega_e^2}\, x_a b_{33}^A$$

$$A_{55} = \int_0^L x^2.a_{33}.dx - \frac{U^2}{\omega_e^2}\int_0^L b_{33}.dx - \frac{U}{\omega_e^2}\, x_a b_{33}^A + \frac{U^2}{\omega_e^2} a_{33}^A$$

$$A_{64} = \int_0^L x.a_{24}.dx - \frac{U}{\omega_e^2}\int_0^L b_{24}.dx - \frac{U}{\omega_e^2}\, x_a b_{24}^A$$

$$A_{66} = \int_0^L x^2.a_{22}.dx + \frac{U}{\omega_e^2}\int_0^L a_{22}.dx - \frac{U}{\omega_e^2}\, x_a^2 b_{22}^A + \frac{U^2}{\omega_e^2} x_a a_{22}^A$$

…(C21)

Super subscript A and subscript a refer to mid-ship cross-section.

$$B_{22} = \int_0^L b_{22}.dx + U\ a_{22}^A$$

$$B_{24} = \int_0^L b_{24}.dx + U\ a_{24}^A = B_{42}$$

$$B_{26} = \int_0^L x.b_{22}.dx - U\int_0^L a_{22}.dx + U\ x_a a_{22}^A + \frac{U^2}{\omega_e^2} b_{22}^A$$

$$B_{33} = \int_0^L b_{22}.dx + U\ a_{33}^A$$

$$B_{35} = -\int_0^L x.b_{33}.dx + U\int_0^L a_{33}.dx - U\ x_a a_{33}^A - \frac{U^2}{\omega_e^2} b_{33}^A$$

$$B_{44} = \int_0^L b_{44}.dx + U\ a_{44}^A$$

$$B_{46} = \int_0^L x.b_{24}.dx - U\int_0^L a_{24}.dx + U\ x_a a_{24}^A + \frac{U^2}{\omega_e^2} b_{24}^A$$

$$B_{53} = -\int_0^L x.b_{33}.dx - U\int_0^L a_{33}.dx - U\ x_a a_{33}^A$$

$$B_{55} = \int_0^L x^2.b_{33}.dx + \frac{U^2}{\omega_e^2}\int_0^L b_{33}.dx + U\ x_a^2 a_{33}^A + \frac{U^2}{\omega_e^2} x_a b_{33}^A$$

$$B_{62} = \int_0^L x.b_{22}.dx + U\int_0^L a_{22}.dx + U\ x_a a_{22}^A$$

$$B_{64} = \int_0^L x.b_{24}.dx + U\int_0^L a_{24}.dx + U\ x_a a_{24}^A$$

$$B_{66} = \int_0^L x^2.b_{22}.dx + \frac{U}{\omega_e^2}^2 \int_0^L b_{22}.dx + U\ x_a^2 a_{22}^A + \frac{U^2}{\omega_e^2} x_a b_{22}^A$$

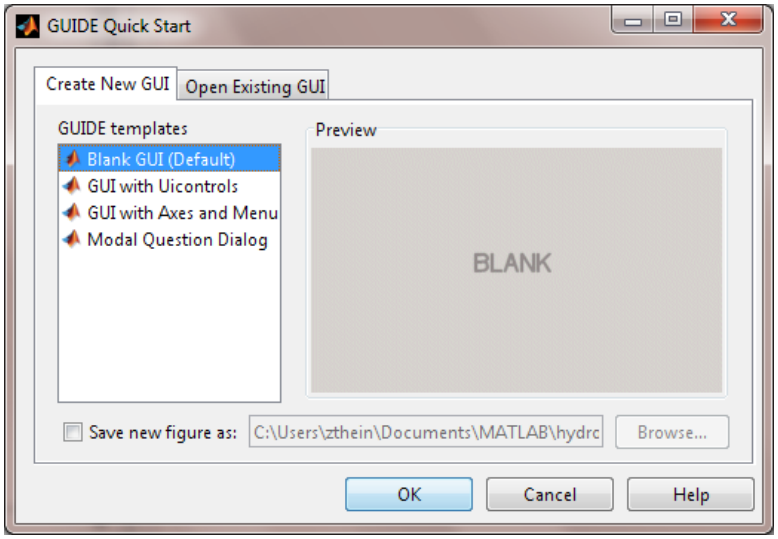
…(C22)

Appendix D: Big Time Tutorials

Starting Chalmers Big Time

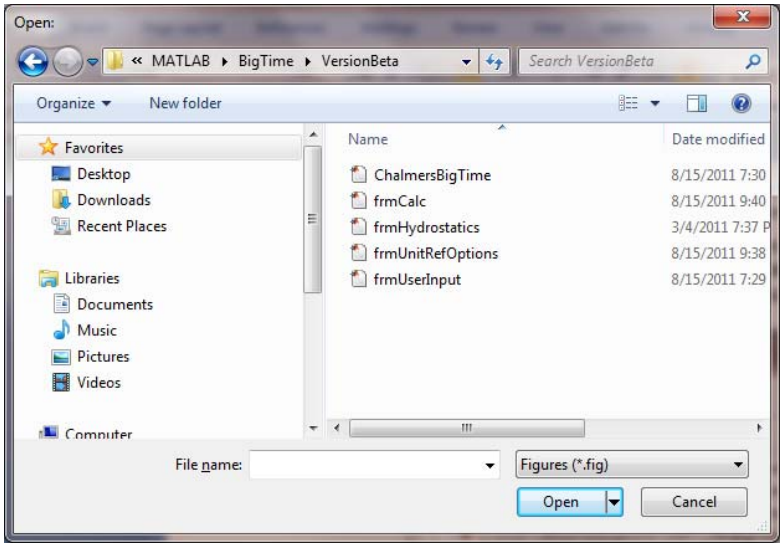
Chalmers Big Time CBT can be launched either in Standalone mode or directly from MATLAB. One of the easiest ways to run CBT in MATLAB environment is starting it from MATLAB's Guide interface.

- Enter the word "***guide***" in MATLAB Command Window.



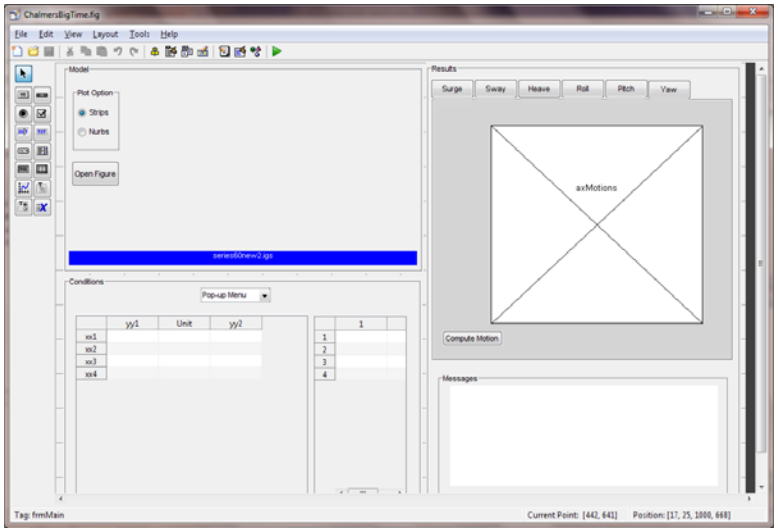
MATLAB GUIDE Quick Start Interface

- MATLAB will take you to **GUIDE Quick Start** Interface.
- Click on **Open Existing GUI** tab
- Browse to the folder where you kept CBT source code.



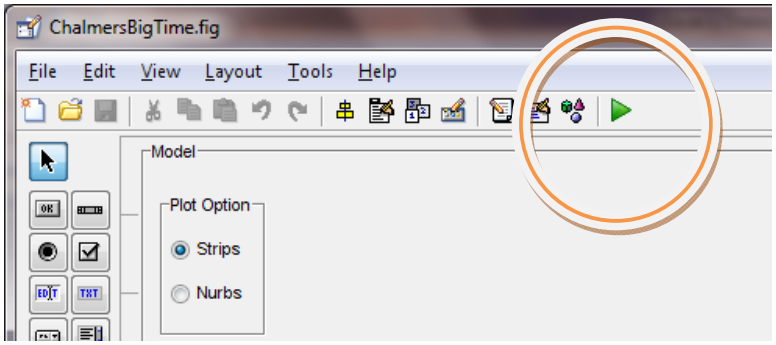
Browsing forChalmersBigTime.fig from MATLAB GUIDE Quick Start Interface

- Look for **ChalmersBigTime.fig** file. This file should be listed at the top.
- Choose the file and Click on **Open**
- MATLAB will take you to ChalmersBigTime Interface.



ChalmersBigTime Interface

- Click on **RUN** menu icon an arrow with green color



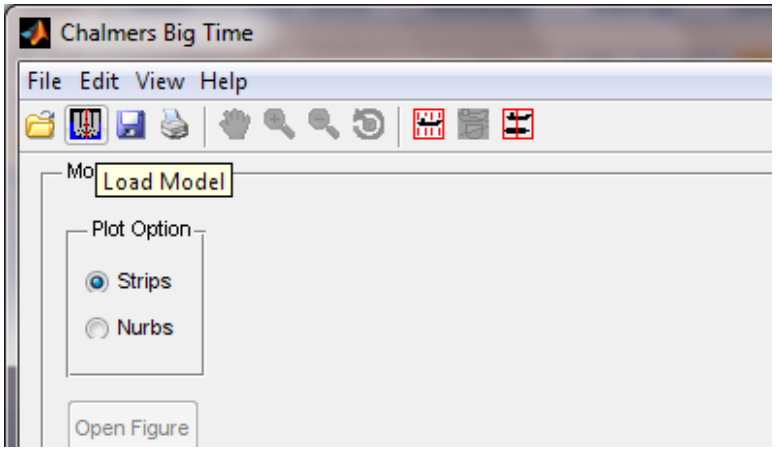
ChalmersBigTime Interface run menu icon

You are now in MATLAB Big Time program environment

Loading a Ship Model

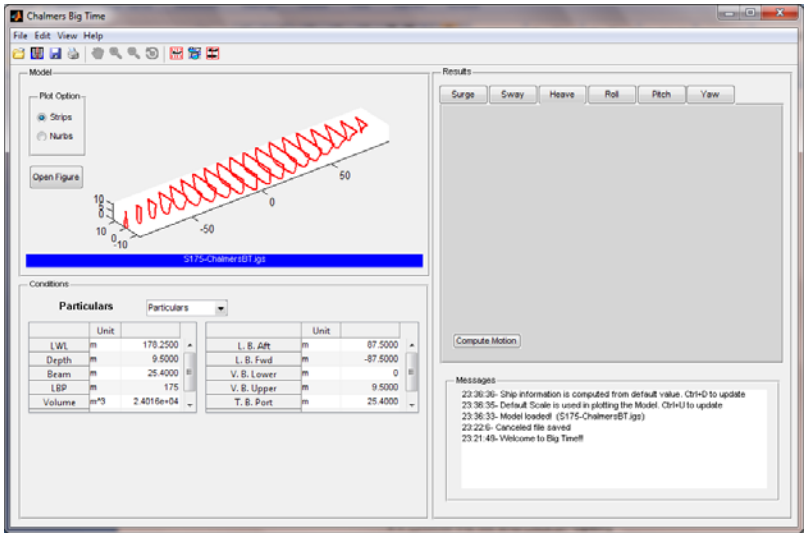
Currently, CBT can only read (.igs) geometry files and (.cbt) Chalmers Big Time files. (.igs) can be loaded into CBT in two ways either from drop down file menu or from menu icon.

- Click on the **Load Model** menu icon between the file open icon and file save icon. Tool Tip will show you the words “Load Model” when you place your mouse pointer above the icon.



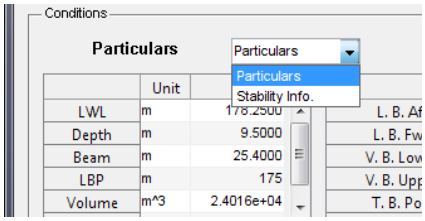
Loading a model from an ico.

- To load model using file menu, click on **File**, and then **Load Model**.
- Browse to your (.igs) file and then click open. Within a couple of seconds, you will receive model information with strip profile.

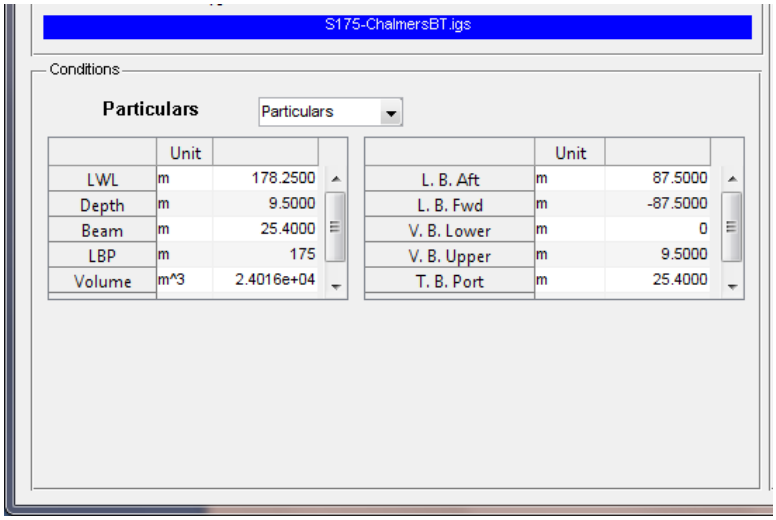


Receiving the model info.

- You can review your model information such as LBP at the lower left condition result area.
- Stability information such as GM can be found in the same area. To review the stability information, choose from the **combo box** above the result field.

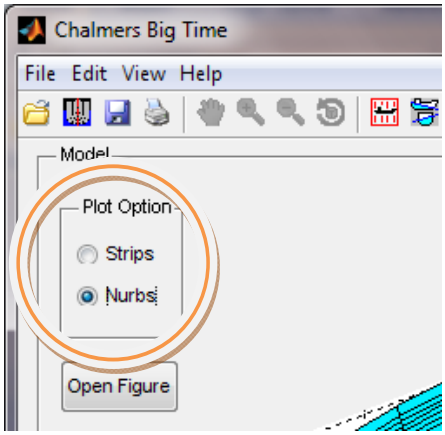


Drop down combo box to upload available results

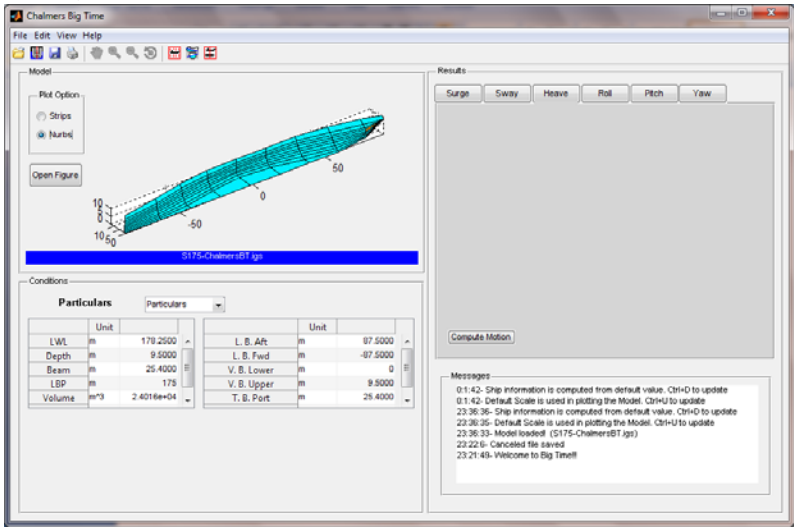


Reviewing Model Information

- Click on **Nurbs** in Plot Option to view Rendered model

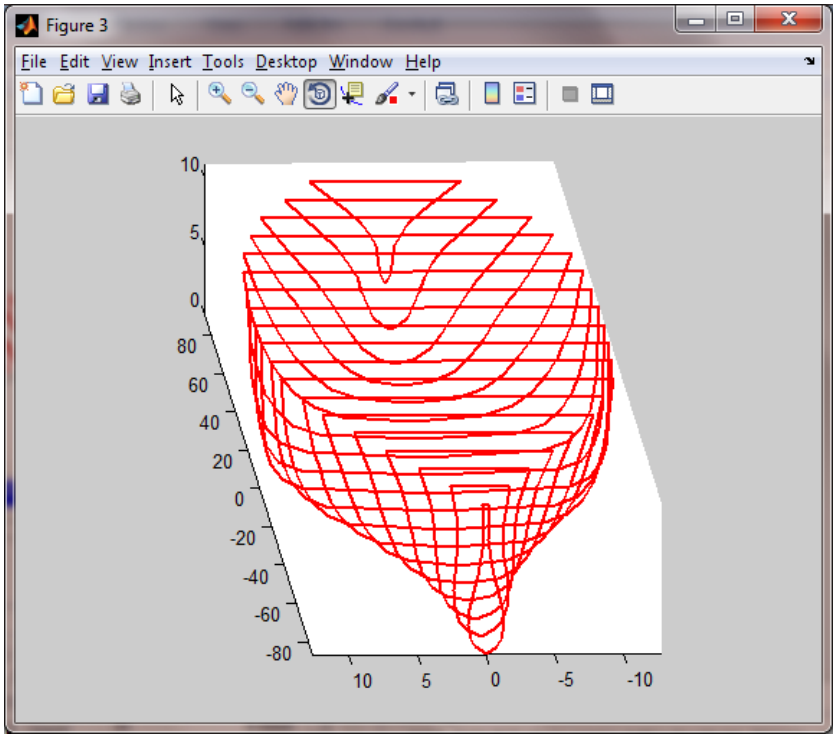


Choosing Nurbs



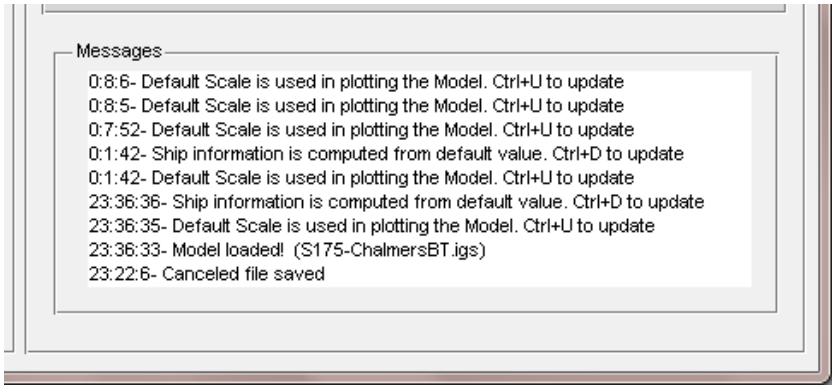
Reviewing Nurbs plot

- Click on **Open Figure** command button under plot option to open the model in a new window.



Reviewing a plot in new window

You should also check if there is any error or message for you in the message window on the lower right hand side area.



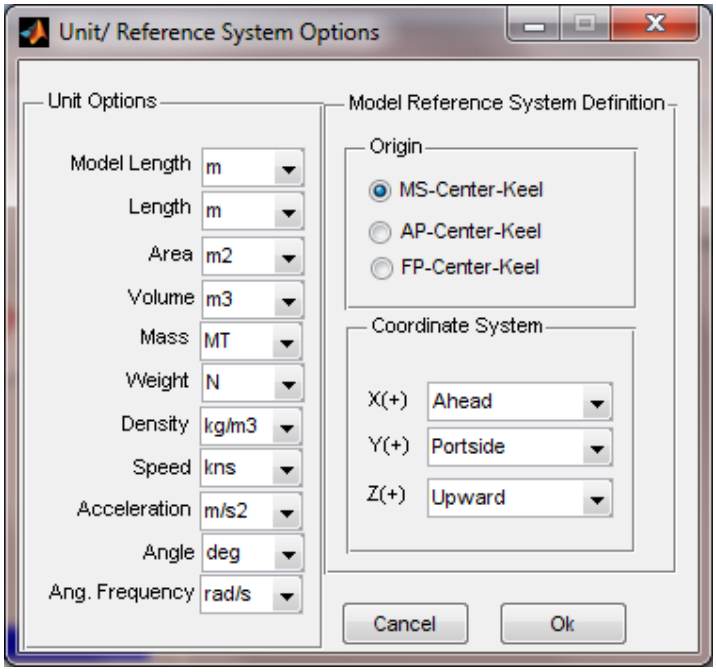
Checkingmessages

Entering User Information

Currently, there are four entry forms which you may want to update the default values with your data before running the simulation. First, you should update units and coordinate system.

- Click on Menu **Edit**, and then **Edit References and Units** menu.

You could also clik on an appropiate icon or use short cut key **Ctrl+R** to update the reference and unit.



Unit/Reference system entry form.

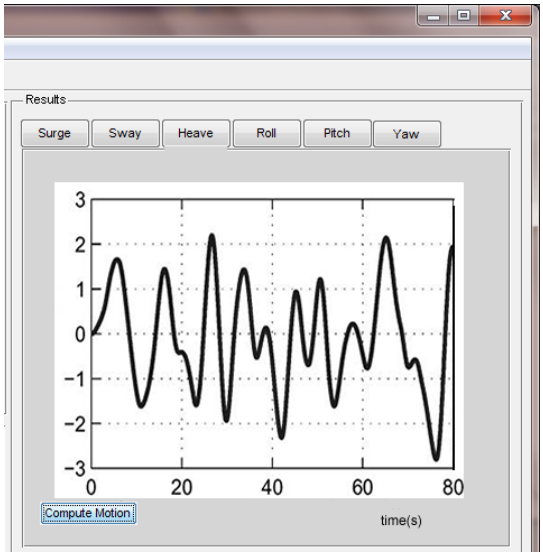
Some of the models you would analyze are in different units and reference systems. Please make sure that your settings are correct before you run the time domain simulation.

- Click on **Edit Draft & Hydrostatic** menu to update model computation. Click on **Edit Computation Options** to update frequency domain computation.
- Click on **Edit Variables** menu to update all required variables in the simulation.

Running Time Domain Simulation

A single click on **Compute Motion** command button will compute complete time domain simulation. Clicking on **Cancel** button will force the computation to stop. When the computation is completed, the "computation successful" message will appear in the message list box. The plots are automatically plotted in the plot area on the top right hand side.

- Click on an appropriate tab to view the result plot in that direction of motion



Result plots.

Saving the Results and User Entry

If you wish to save your work, you could do so by clicking **Save** or **Save As** from the file menu. The file will save as (.cbt) file.

Use File **Open** menu to load the file you have saved. You could also load this file from your mobile device if your device is installed with CBT-M.

Appendix E: Programming Guide

The purpose of this section is to provide architectural background of the Chalmers Big Time Source Code.

Naming System

Naming system is developed for the source code so that other developers could easily verify and distinguish the variables, objects, and files. The system is listed in table E-1.

Table E-1: Naming System

Type		Leading Letter(s) in a Given Name
Variable	String	s
	Double	d
	Boolean	b
	Array	arr
	Structure Array	struct
File	figure	frm
	m-file	c (for specific purpose) mod (for general purpose)