Direct Inclusion of Seakeeping Ergonomic Criteria in the Computerized Design of Small Crafts

by

Mizanur Rahman Majumder

Naval Architect & Marine Engineer (B.Sc.)
Bangladesh University of Engineering & Technology, Dhaka, Bangladesh, 1980.

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
(Department of Mechanical Engineering)

We accept this thesis as conforming to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA
January 12, 2000
© Mizanur Rahman Majumder, 2000
In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of **MECHANICAL ENGINEERING**

The University of British Columbia
Vancouver, Canada

Date **11 JAN 2000**
Abstract

Dr. Akinturk showed that the design process of a small craft can be computerized, and new design nodes, such as crew safety or acceleration levels can be included in the preliminary design. This study goes one step further by directly including ergonomic criteria in the preliminary ship design process. In addition, the design platform selected, is an easily available one. It is shown that an integrated technical computing environment such as Matlab, can do the preliminary small craft design with the inclusion of ergonomic criteria. This demonstrates also that the methodology developed in this study can be implemented and extended easily by others. The design process used in this study will be referred as “Matship”.

The ergonomic criteria, implemented in Matship, include constraints on RMS motion amplitudes and accelerations, peak motion responses and ergonomic criteria that relate maintaining one’s balance onboard of a ship to ship motions. In the last one, an empirical relation, developed by Kimura, is used between the first moments of the spectra of vertical and longitudinal accelerations on the deck and human balance. The study shows the effect of human factors on the principle dimensions of the boats and their operational speed.

The preliminary results reported in this thesis show that seakeeping ergonomic criteria can be implemented explicitly into the preliminary design of a small craft, for crew safety and passenger comfort. This thesis presents a methodology to include seakeeping ergonomic criteria in the preliminary design process and discusses its effects on the design parameters such as principal dimensions, power requirement, cost etc. The effects of these criteria on the final design can be quantified in terms of principal dimensions or monetary terms. Matship presents a methodology to include ergonomic criteria in the preliminary design and discusses their effects on the ship parameters. This study suggests that length restriction imposed on fishing vessels in Canada and elsewhere may cause unnecessary risk to crew of smaller vessels and should be further studied. The Matlab files used in this study are available at University of British Columbia’s (UBC) Mechanical Engineering web site, http://www.mech.ubc.ca/.
# Table of Contents

Abstract ii

Table of Contents iii

List of Tables v

List of Figures vi

List of Files viii

List of Equations x

Acknowledgements xi

Nomenclature xii

1 Background 1

1.1 Introduction: 1

1.2 History: 1

2 An overview of Ship Design Processes 5

3 Optimization 13

3.1 Introduction: 13

3.2 Constraints: 15

3.3 Sensitivity Analysis: 17

3.4 Optimization in Ship Design: 17

3.4.1 Introduction: 17

3.4.2 Optimization Criterion: 18

3.4.3 Optimization Software: 19

3.5 Optimization procedures: 20

3.5.1 Optimization of Linear Function: 20

3.5.2 Optimization of Non-linear Function: 20

3.5.3 Searching Techniques: 21

3.5.4 Mathematical Programming Methods: 21

3.6 Features of Matlab's Optimization Toolbox 21
List of Tables

Table 1: Cost function 26
Table 2: Comparison of fishing vessel parameters of different series 29
Table 3: Geometric Properties of the UBC Series Models 29
Table 4: Accidents onboard Ships by Vessel type 30
Table 5: 10 year averages of percentage values of primary contributing factors in casualties aboard all vessel types. 31
Table 6: Comparison of the results from Echidna and Matlab to the real Kynoc. 48
**List of Figures**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Design Spiral</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>Andrew's Overall Model of the Ship Design Process</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Design Process by Mistree et al</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>Reduction of freedom of making changes to the design and increase in design knowledge with the progress</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>Shift of peak frequencies of sea waves and ship's response</td>
<td>35</td>
</tr>
<tr>
<td>6</td>
<td>Flow diagram of Input and Output variables in the Matship Process</td>
<td>43</td>
</tr>
<tr>
<td>7</td>
<td>Influence of including Equation 11 as a constraint on the lengths of the designs</td>
<td>47</td>
</tr>
<tr>
<td>8</td>
<td>Influence of including Equation 11 as a constraint on the costs of the designs</td>
<td>47</td>
</tr>
<tr>
<td>9</td>
<td>Comparison of length values obtained by Matlab and Echidna for UBC Series.</td>
<td>50</td>
</tr>
<tr>
<td>10</td>
<td>Comparison of beam values obtained by Matlab and Echidna for UBC Series.</td>
<td>50</td>
</tr>
<tr>
<td>11</td>
<td>Comparison of draft values obtained by Matlab and Echidna for UBC Series.</td>
<td>51</td>
</tr>
<tr>
<td>12</td>
<td>Results for vessel's displacement from Matship and Echidna.</td>
<td>52</td>
</tr>
<tr>
<td>13</td>
<td>Results for vessel's cost from Matship and Echidna.</td>
<td>52</td>
</tr>
<tr>
<td>14</td>
<td>% changes in Length due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>53</td>
</tr>
<tr>
<td>15</td>
<td>% changes in beam due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>54</td>
</tr>
<tr>
<td>16</td>
<td>% changes in draft due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>55</td>
</tr>
<tr>
<td>17</td>
<td>% changes in block coefficient due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>55</td>
</tr>
<tr>
<td>18</td>
<td>% changes in costs due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>56</td>
</tr>
<tr>
<td>19</td>
<td>% changes in total weight due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>56</td>
</tr>
<tr>
<td>20</td>
<td>Percentage changes in required power due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>57</td>
</tr>
<tr>
<td>21</td>
<td>% changes in RMS heave amplitude due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>58</td>
</tr>
<tr>
<td>22</td>
<td>% changes in RMS pitch amplitude due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>58</td>
</tr>
<tr>
<td>23</td>
<td>% changes in peak heave ratio due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>59</td>
</tr>
<tr>
<td>24</td>
<td>% changes in peak pitch ratio due to Kimura's Crew Safety Criteria (Rule-III)</td>
<td>60</td>
</tr>
</tbody>
</table>
List of Figures continues....

Figure 25: % changes in Kimura's threshold value due to Kimura's Crew Safety Criteria (Rule-III) 60
Figure 26: Effects of varying threshold value on length 62
Figure 27: % change in length due to the changed threshold value 62
Figure 28: Effects of varying threshold value on beam 63
Figure 29: Effects of varying threshold value on draft 63
Figure 30: Effects of varying threshold value on block coefficient 64
Figure 31: Effects of varying threshold value on displacement 64
Figure 32: % change in displacement due to the changed threshold value 65
Figure 33: Effects of varying threshold value on cost 65
Figure 34: % change in cost due to the changed threshold value 66
Figure 35: Effect of Rules on Ship's Cost 67
Figure 36: Effect of Rules on Ship's Displacement 67
Figure 37: Effect of Rules on Ship's Length 68
Figure 38: Effect of Rules on Ship's Beam 69
Figure 39: Effect of Rules on Ship's Draft 69
Figure 40: Effect of Rules on Ship's Block Coefficient. 70
Figure 41: Effect of Rules on Kimura's Threshold Value 71
Figure 42: Effect of Rules on Ship's Required Power 71
Figure 43: Kimura's Threshold values at the forward of the vessel [250 tons hold capacity] at different seastates and ship speeds. 72
Figure 44: Kimura's Threshold values at the midship of the vessel [250 tons hold capacity] at different seastates and ship speeds. 72
Figure 45: Kimura's Threshold values at the aft of the vessel [250 tons hold capacity] at different seastates and ship's speeds. 73
Figure 46: Maximum Attainable Speed for the vessel 74
List of Files

File 1: designopt.m
File 2: we2w.m
File 3: sspec.m
File 4: myfun.m
File 5: mycon.m
File 6: cons_bt1.m
File 7: cons_bt2.m
File 8: cons_lb1.m
File 9: cons_lb2.m
File 10: cons_disp.m
File 11: cost.m
File 12: displace.m
File 13: tweight.m
File 14: hv_amp.m
File 15: pt_amp.m
File 16: peak_hv.m
File 17: peak_pch.m
File 18: rms_hv.m
File 19: rms_pch.m
File 20: s_acc_h1.m
File 21: s_accv1.m
File 22: s_acc_h2.m
File 23: s_acc_v2.m
File 24: s_acc_h3.m
File 25: s_accv3.m
File 26: rmsveraccact.m
File 27: stab.m
File 28: stablite.m
File 29: tm01.m
List of Files continues....

File 30: tm02.m 120
File 31: tm03.m 121
File 32: pwerfish.m 122
File 33: Sminteg.m 124
File 34: tmintp.m 124
File 35: tmmax.m 124
File 36: vcrit.m 125
File 37: okul.m 126
File 38: maintrgb.m 131
## List of Equations

| Equation 1 | 27  |
| Equation 2 | 32  |
| Equation 3 | 32  |
| Equation 4 | 32  |
| Equation 5 | 32  |
| Equation 6 | 33  |
| Equation 7 | 33  |
| Equation 8 | 35  |
| Equation 9 | 37  |
| Equation 10 | 37  |
| Equation 11 | 46  |
| Equation 12 | 61  |
Acknowledgements

It is not possible for me to weigh on a scale and compare the contributions that different personnel have made towards the project undertaken by the undersigned. But out of the many people who deserves my most sincere gratitude, Dr.S.M.Calisal's name comes the first, who was my supervisor and has always been behind me giving the most essential elements behind the successful completion of the project, that is the "mental support", "encouragement" and "guidance".

Next to him is Dr.Ayhan Akinturk, my co-supervisor, who virtually taught me how to crawl, then stand on my feet and finally walk with the project. He has guided me in learning optimization techniques, usage of an expert system and various other fields in my project work.

I would also like to thank Mr. Jon Mikkelsen for his continual suggestions and Mr. Jeffery Liggins and Mr. Methias Borstad for their assistance in my learning computer software, needed for the project work.

Mr.Jerry Stengaard, from the Ocean Engineering Center of BCRI deserves special thanks from me for his assistance in providing me with some data of the model tests, which played an important role in my project work.

My friends and colleagues also deserve my gratitude for their encouragement and technical contribution through discussions. I am very much thankful to my sister who has always been behind me to go for higher studies and wants me to continue with further studies.

Finally, my gratitude goes to my wife and two kids, who supported me with everything they could do.
## Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>(B)</td>
<td>Beam of a ship</td>
</tr>
<tr>
<td>(C_b)</td>
<td>Block coefficient</td>
</tr>
<tr>
<td>(C_p)</td>
<td>Prismatic coefficient</td>
</tr>
<tr>
<td>(D)</td>
<td>Depth of a ship</td>
</tr>
<tr>
<td>(F_n)</td>
<td>Froude number ((F_n = V/\sqrt{g/L}))</td>
</tr>
<tr>
<td>(g)</td>
<td>Acceleration due to gravity ((9.807\text{m/sec}^2\text{ or }32.2\text{ft/sec}^2))</td>
</tr>
<tr>
<td>(H)</td>
<td>Average wave height</td>
</tr>
<tr>
<td>(H_{1/3})</td>
<td>Significant wave height</td>
</tr>
<tr>
<td>((H_{1/3})_S)</td>
<td>Significant wave height for sea state 5</td>
</tr>
<tr>
<td>(k)</td>
<td>Wave number ((2\pi/\lambda))</td>
</tr>
<tr>
<td>(L_{bp})</td>
<td>Length between perpendiculars of a ship</td>
</tr>
<tr>
<td>(L_{oa})</td>
<td>Length overall of a ship</td>
</tr>
<tr>
<td>(L_{wl})</td>
<td>Length waterline of a ship</td>
</tr>
<tr>
<td>(LT)</td>
<td>Long tons ((1016\text{ [kg]}))</td>
</tr>
<tr>
<td>(NC)</td>
<td>No sea-keeping constraints</td>
</tr>
<tr>
<td>(NH_{1/3})</td>
<td>Non-dimensional significant wave height (NH_{1/3} = H_{1/3}/(H_{1/3})_S)</td>
</tr>
<tr>
<td>(PC)</td>
<td>Peak pitch &amp; heave constraints</td>
</tr>
<tr>
<td>(RMS)</td>
<td>Root Mean Square</td>
</tr>
<tr>
<td>(T)</td>
<td>Mean draft of a ship</td>
</tr>
<tr>
<td>(K)</td>
<td>Kimura’s Threshold factor used in Rule III</td>
</tr>
<tr>
<td>(tmOl)</td>
<td>Kimura’s threshold constraint at aft</td>
</tr>
<tr>
<td>(\bar{T})</td>
<td>Average wave period for a given sea state</td>
</tr>
<tr>
<td>(\bar{T}_{\text{max}})</td>
<td>The wave period that maximizes wave energy spectrum for a given sea state</td>
</tr>
<tr>
<td>(V)</td>
<td>Ship speed ((\text{knots}))</td>
</tr>
<tr>
<td>(z)</td>
<td>Heave amplitude</td>
</tr>
<tr>
<td>(z_{rms})</td>
<td>RMS heave amplitude</td>
</tr>
<tr>
<td>(\dot{z})</td>
<td>Heave acceleration</td>
</tr>
<tr>
<td>(\theta)</td>
<td>Pitch amplitude</td>
</tr>
<tr>
<td>(\dot{\theta})</td>
<td>Pitch acceleration</td>
</tr>
<tr>
<td>(\theta_{rms})</td>
<td>RMS pitch amplitude</td>
</tr>
<tr>
<td>(\lambda)</td>
<td>Wave length</td>
</tr>
<tr>
<td>(\bar{\lambda})</td>
<td>Average wavelength for a given sea-state</td>
</tr>
<tr>
<td>(\omega_e)</td>
<td>Encounter frequency ([\text{radian/s}])</td>
</tr>
<tr>
<td>(\omega_n)</td>
<td>Non-dimensional encounter frequency ((\omega_n = \omega_e/\sqrt{g/T}))</td>
</tr>
<tr>
<td>(\omega_{\text{max}})</td>
<td>Frequency at the maximum spectral energy ([\text{radian/s}])</td>
</tr>
</tbody>
</table>
1 Background

1.1 Introduction:

In October 1997, the Naval Architecture department under the School of Mechanical Engineering of the University of British Columbia received a proposal from Dr. Gordon Thomas of Kelwona, BC, a world famous explorer, for the design of a ferry for their transglobal expedition. In this proposed expedition, Dr. Thomas, along with Dr. Shappiro of Russia, a famous Arctic explorer were planning to drive a truck from the east most point on the map to the west most point or vice versa. They will cross four major waterways, namely, the Irish Sea, English Channel, Bering Strait and Yukon river and use a ferry which is to be run by the truck’s power to cross these waterways. The ferry will have to be as small as possible for easy transportation, but very safe and must be comfortable for the explorers, who are not professional mariners. This is where the need for optimization came into the picture, where cost control along with the safety and the crew comfort got the highest importance. The Transglobal Expedition Project is being dealt by undergraduate students as a project and they are using dimensions availed through the application of the present process.

To take it one step further, it was thought if Matship could be made available to all designers for any ship design purpose. Other work done along the same line was studied and an easy platform was sought, which would be easy to use, available in the world market and more versatile than the existing computational platforms used in preliminary design stage.

1.2 History:

There is no firm date or year when human being invented a floating system that was used for their transportation. Until a few decades back it was thought that the Egyptians invented boats around 5,000 years ago. New evidences from archaeological findings confirms the age of some of the Egyptian Pyramids to be more than 12,000 years old and surprisingly they also found that boat-like structures had been used at that time. Though no body is 100% sure that
this is the first civilization on the face of the earth, it is accepted that within this present civilization, a boat is one of the earliest inventions of mankind.

Ancient civilizations knew nothing about moment of inertia or any mathematical concept, but from their practical experience, definitely knew about increased stability when they extended their arms while walking on a fallen log to cross the creeks. The clever ones used this concept in making their boats safer by adding outriggers, and developed the first concept of a catamaran boat.

In the last few centuries, when watercraft started playing a major role in transportation, colonialization and war, extensive research and development had taken place in regards to their design, construction and varieties of usage. Safety, stability and sea-keeping qualities of the vessel and the comfort of the crew and passengers are presently being given priority in ship design, which used to be the "carrying capacity" in the past. Safety is a perceived quality that determines to what extent the management, engineering and operation of a system is free of danger to life, property and the environment.

It is also implied that there is no absolute level of safety. It is virtually impossible and/or not cost effective to make anything 100% safe for any situation. The design is usually based on an optimum level of safety, usually guided by different international standards.

It is also interesting to note that requirements for each category towards a safer vessel may dictate conflicting goals. For example, increased metacentric height (GM), which is a parameter related to initial stability, could be desirable from a vessel's safety point of view, though it may prove otherwise for the crew on-board as the frequency of roll motion may fall in the uncomfortable region. An increase in GM usually calls for wider and/or deeper vessel, which in turn could increase the cost of the vessel.

Unlike other land-based industries, a ship is a moving platform. In the big ships, even with a small rolling or pitching angle, the linear acceleration level, at points away from the axis of rotation of the vessel may be quite significant and may be of great discomfort to the personnel
on board. In the case of small vessels, especially workboats and fishing vessels, where the crew on board has to work on the exposed decks most of the time; lack of balance makes the job much more difficult and dangerous for them.

Thus, the safety criterion has to be included into at the preliminary design stage, as nothing much can be done after the vessel has been constructed. The safety criteria can be used as constraints in the optimization and design parameters may be achieved for desired safety level. For example, in the first category, improvements in a vessel's safety could mean improved design features, e.g. increased initial stability, watertight deckhouses or a better understanding of the factors that degrade safety in the vessel's lifetime. For the second category i.e. occupational accidents, improved safety may mean improved working conditions on-board the vessels. This could involve reduction in noise levels, as well as vibration and motions i.e. motion amplitudes and acceleration levels, better deck and bridge design, training and education of crew, etc. This is the second category, which has more influence on the human comfort and safety and is looked after throughout this work.

Traditionally, even at present, ship design is mainly based on existing data from similar vessels, without going very much into the detail of their initial and operating cost, sea-keeping quality, crew comfort etc. Usual practice is to reproduce successful ships or some designs similar to it for a blind success. Designing a vessel based upon previous experience is satisfactory if there are sufficient number of similar type vessels that have experienced a complete spectrum of operating conditions. However, with the rapidly changing operating requirements of today, it is rare to find more than a few similar vessels. Of these similar vessels, none possibly have met the sea conditions that the present vessel may be designed for.

Though following a "Design Spiral" concept is quite a common practice in marine design, "Optimization" techniques have not been very much used by ship designers. Optimization has not been used as a tool in ship design, in any of the known ship design software in the market. Most of the ship designers do not have enough background to develop their own computer program for design optimization and rather will be happy to use available design software with optimization tools. This study is a help to marine designers to comfortably use the
advantage of optimization tools in their design in achieving both a safer and more comfortable ship.

In recent years, expert systems have been developed by virtue of super-speed computational facilities with modern-day computers and development of applied mathematical science. This project will introduce two new objectives. First, the usage of Kimura’s threshold value in preliminary ship design, for the safety and comfort of the crew and passengers, and second, a design platform which is easily available and easy to use in terms of mathematical conception. Optimization technology is used for the most optimum and reliable result. Human comfort factors like Rule-I and Rule-II (Please see Chapter 4 for details of these rules) have been used by Akinturk [1] within the expert system environment “Echidna”, a software, developed at Simon Fraser University, but Rule-III (Please Chapter 4 for details of these rules) was not used before. In addition to these three rules, a fourth rule, named, Rule-IV (see Chapter 4 for details of the Rule-IV) is used in this study, as constraints used in this rule is available from different studies.

The software “Echidna” is not easily available in the world market and needs extensive knowledge in computer programming languages. With the present process (Matship), the ship designers will have more confidence and comfort in using this non-traditional but scientific system of design within a user-friendly platform. If there are members of the community that are not taking advantage of these modern analytical capabilities, it is hoped that this thesis will help them appreciate the merits of appropriate application to their problems.
2 An overview of Ship Design Processes

For decades ships have been designed using the well known “basis ship approach” of following an existing design, that is, reproduction of an existing so-called successful design, just by scaling it up or down with minimal alterations. Ship design involves numerous uncertainties. Some of them are related to ship’s operational environment, such as, it’s zone of operation (route), market conditions etc. and the others are related to the uncertainties involving the estimation of the ship’s parameter, weight, cost etc.

Ship design usually consists of four steps, viz., conceptual design, preliminary design, contract design and detail design [23]. Conceptual design involves a technical feasibility study and finding of different alternatives. Preliminary ship parameters like, length, beam (draft is usually restricted by the operational constraints and are usually proposed by owners), power etc., are usually determined in this stage. In the preliminary design phase, the above parameters are further refined. Model tests, if any, are usually performed during the third phase (contract design) when the design is further improved. After the acceptance of the contract design, the detailed design for the construction of the vessel is prepared.

Different concepts in scientific ship design processes have been proposed, developed and modified, but the design spiral concept had been the most extensively used process till the development of the fast computer computational system and optimization techniques.

In 1959, Evans [24] first visualized and modeled the process of ship design, now known as the “Ship Design Spiral” (Figure 1). The principle elements of this approach are iterative, sequential application of analyses based on the Newtonian principles of reductionism, mechanism and synthesis utilizing the intuitive skills of the designer. The design starts with the mission requirements and proceeds through the nodes of the spiral towards a detailed design. It is also laborious and expensive. With the refinements made to the “Design Spiral” over time, the basic features remained unchanged. Buxton [21] introduced economic issues into the spiral and time was added as a third dimension by Andrew [22] (Figure 2). Andrew
stated that the advantage of his representation is that many dialogues and constraints on a
designer can be shown as fundamental to the process.

![Design Spiral](image)

**Figure 1: Design Spiral**

Mistree et al [14] states that there are two salient features of the design spiral. Firstly, while
spirals represent converging towards a product, the process is divergent with respect to
information and the increasing detail of definition. Secondly, it is recognized that when the
spiral was first formalized and in the years following, represented a descriptive model that
portrayed how design was done. It represented both the state-of-art and the state-of-industry.
Two principal limitations of the spiral are that the process of design is assumed to be sequential and the opportunity to include life cycle considerations is limited. Though the design spiral has been further developed and modified by Andrew [22], into a better iterative helical "corkscrew" to introduce constraints related to the design process and constraints originating from the design environment, directly into the design at various stages, the process still remained sequential. Using this concept directly with the computer for design is not that feasible. Though the spiral approach may result in a satisfactory design it does not promote the identification of superior solutions.

Mistree et al [14] criticized the design spiral approach mentioned above as a process where it is difficult to incorporate life cycle considerations. They proposed the so-called decision based design in 1990. The actual design process, according to this representation, takes place inside the frustum in contrast to the spiral approaches and proceeds around the frustum. Each phase of the design is depicted as an irregular disk-like shape inside the frustum. In their representation, converging to a solution means the irregular shaped disc becomes circular. A representation of the design process proposed by Mistree et al is shown in Figure 3.

Modern design process developers are using artificial intelligence (AI) of the computer and information and data available through the Internet. Design until now is limited to the designer's skill, experience and the limited available data within a designer's reach.
In their paper, "Decision-Based Design: A Contemporary Paradigm for Ship Design" [14], F. Mistree et al described different developments in ship design process. They highlighted two different model types for the design process in classical engineering design, namely, perspective models and descriptive models, to improve the efficiency and effectiveness of designers and the process they use. These are as follows:

The perspective model of design consist of the following basic activities [14]:

- **Analysis**: A process of partitioning or decomposing any system into its subsystems and components to determine their individual and collective nature, proportion, functions, relationships, etc.
- **Synthesis**: A process of integrating a collection of sub-systems, so as to create a system with emergent properties.
- **Evaluation**: A process of assessing the degree to which a solution satisfies the goals that were originally stated.

According to the opinion of De Boer [15], there is basically a three-step pattern of diverging, systemizing and converging is generally recognized in each phase and sub-phase of a perspective model. In the first step of divergence, the designer generates a large number of ideas, alternatives and possible solutions. These ideas are then analyzed and synthesized in the second step, into forms that may represent acceptable solutions to the problem. And, finally, the refined ideas are further synthesized, analyzed and evaluated to achieve the most acceptable solution.

The descriptive model of design consist of the following four basic activities [17]:

- **Problem Analysis**: A problem statement is developed.
- **Conceptual design**: The statement of the problem is used to generate a collection of broad solutions.
- **Embodiment design**: The solutions in the collection are refined for the purpose of eliminating the less satisfactory ones, till the final most satisfactory one remains.
• **Detail design**: All the details of the final design are specified and the final design and documentation are produced.

There is no certain rules or sequence that all the designers always have to follow, but monitoring a number of designers, Wallace and Hales [18, 19] concluded that the designers:

- develop the functional aspects of the design in stages throughout the problem solving effort rather than during an initial functional development stage;
- use functional considerations that remained qualitative while often the form considerations become quantitative during the problem solving effort;
- make decisions based on qualitative, subjective reasoning during all phases of design;
- use their knowledge to influence the generation of ideas for problem solving;
- use their knowledge to influence the ideas to be evaluated;
- use their knowledge to influence their problem solving method;
- perform mental, visual and physical simulations as an aid to understand problems and evaluating solutions;
- attempt to find a satisfactory solution rather than an optimal one.

Deviation from scientific approach might sometimes result in a non-optimal and very often an unscientific solution or design. Influence of human factors on a design may be reduced by computerization of design using optimization and other scientific techniques.

Mistree et al [14] comments that, in the **Decision-Based Design** approach, the main role of the designer is to make decisions. The designer using this approach may use computers, as opposed to design that is assisted by the use of computers, optimization methods or other tools like finite element analysis (FEA). Some basic features of this approach are:

- The designer is the principal decision-maker.
- The design involves a combination of some sequential and some concurrent decisions.
- Decision involves information that comes from different sources and disciplines. In many cases, all information for making a decision may not be available.
• The decision making is hierarchical and interactions between the decisions are a must.
• Use of aids like computers in numerical calculations and analysis and data processing increases the productivity.

A more scientific design process is called the **Concurrent Engineering Design Process**, which is described by Winner [20] as, "*a systematic approach to the integrated, concurrent design of products and their related processes, including manufacturing and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirements*". This is characterized by a focus on the customer's requirements and priorities, which is a common case in ship design. In the early stage of concurrent design, major design decisions are made with information based on designer's judgement and experience, called soft information.

![Figure 4: Reduction of freedom of making changes to the design and increase in design knowledge with the progress](image)

Mistree et al [14] also highlights the fact that as the design process progresses, the freedom to make changes reduces and the knowledge about the object of the design increases as shown in Figure 4. Concurrent engineering helps in dragging the knowledge curve Figure 4 to the left, resulting in an increase of the ratio of hard to soft information that is available in the early
stages of design. The gradual improvement in the quality and reliability of information and data may lead to the design which may be less costly, but more scientific than achieved through traditional sequential methods. The benefits of concurrent engineering (applied to the production systems) in rough quantitative representation are [14],

(a) improvement in design resulting in more than 50% reduction of engineering change orders in early production,
(b) reduction in product-development cycle by about 40–60% compared to sequential process,
(c) reduction of manufacturing cost by 30–40% with multifunction teams, integrated in product and process design and
(d) reduction by 75% of the cost of the scrap and rework through the use of product and process design optimization [20].

Report on the benefits of concurrent engineering applied by few companies like McDonald Douglas, Boeing Ballistic System Division, AT&T, Deere & Company, Hewlett-Packard Co. Instrument Division and IBM are shown in ref [20].

It is obvious from the above discussions that tomorrow’s engineering designs should follow the concurrent engineering design approach to its best extent for the life cycle. It should also be added that it might not be only one process or approach for all designs. The elements, processes, sequences and approaches within concurrent engineering may vary for the benefit of the design goal.

The meaning of using the concurrent system of design in the case of ship design is to have an option of including, modifying and improving the design components within the design program. Concurrent design process is not a common practice in ship design, like in other engineering design. The reason might be the number of uncertainties and variables involved in ship design. One of these design variables is the seastate for a particular rout, which varies much with the location and time of the year. The seastate data are statistical and are time dependent and sometime very occasional peak values are neglected. The designer can not be sure that the designed vessel will survive those peak conditions. Creating a database for all
different types and series of vessels need extensive lab work and tank tests and centralized data acquisition, processing and storing option with worldwide accessibility. This is a very expensive option. Some of the professional societies maintain database for different types of vessels and the designers can buy these data. What is important is to update these data and use them as a design tool in computerized ship design. The design environment used in this paper is open to this option of development with acquired knowledge with time. Files containing the database and equations for calculating different ship parameters may be added with different options into the design process, and existing files may be modified from time to time to suit different types and sizes of vessels and their lifecycle.
3 Optimization

3.1 Introduction:

In a classical sense, optimization is defined as the technique of obtaining the best results according to certain objectives, while satisfying some fixed requirements. Recent advancement in applied mathematics, operations research, and digital-computer technology enable the solution of many complex industrial problems in engineering and economics to be optimized successfully by the application of logical and systematic techniques. Nowadays, many crucial decisions are made by choosing quantitative measures of effectiveness and then optimizing it.

Douglas J. Wilde and Charles S. Beightler [32] in their book, "Foundation of Optimization", described optimization analogous to mountain climbing. They write: "In mountain climbing, whatever route a mountain climber takes, he recognises the peak when he arrives there. Equally important, the climber can tell when he is not at the top and must therefore continue his climb". To justify the different numbers of ways to achieve optimization, they also write, "Despite strong similarities, the diverse optimization schemes have many intriguing differences and special characteristics. After all, differently shaped mountains require different climbing strategies, and one must base his approach not only on the obstacles in his path, but also on the climate surrounding the mountain and on the equipment at hand. Similarly in selecting a specific optimization technique one should take account not only of the mathematical topography, with its computational glaciers, cliffs, and crevasses, but also of the intellectual climate surrounding the problem and the computing facilities available".

Deciding how to design, build, regulate, or operate a physical or economic system ideally involves three steps: First, one should know accurately and quantitatively, how the system variables interact. Second, one needs a single measure of a system effectiveness expressible in terms of the system variables, that is, one should be able to form a set of equations involving all the dependent and independent variables of the system which have influence on that single
measure. Finally, one should choose those values of the system variables yielding optimum effectiveness. That is, optimization and choice are closely related.

The most important step is the first step, for, it is where the decision-maker's professional skill and training is proved essential to achieve a reliable result. A true and clear picture of the system is essential for formulating the equations, boundary conditions and commands used in optimization. For example, in ship design, the ranges of ratios between different parameters and the coefficients etc. are very much dependent on the type and size of vessel and while designing with optimization using bounds and constraints, these are to be very carefully dealt with.

In most engineering systems, the measures of system effectiveness are, profit, cost or efficiency. The second step of determining which of these measures is to be optimized, is also very important for optimization process, as one of these measures may be adversely affected by the optimization of the other. In most of the cases a more efficient design will be a more expensive design.

Optimization techniques can be applied successfully to problems of a sophisticated and complex situation. Two factors made this possible- the development of high-speed digital computers, and the application of mathematical analysis to the development of numerical techniques for obtaining maxima and/or minima. These numerical procedures bypass many difficulties associated with calculus and the contribution of computer's fast computing capabilities, enables the practical utilization of these numerical techniques in virtually no time and at a tolerable expense.

In order to consider attaining certain goals in an optimal manner, we must first define our objective quantitatively. We recognize that the desired objective must depend upon the proper choice of certain variables \( x_1, x_2, \ldots, x_n \), which are sometimes referred to as policy variables or decision variables. In an ordinary optimization problem, an objective function \( y(x_1, x_2, \ldots, x_n) \), (sometimes called as performance index, profit function, or return function) which represents some quantity, such as profit or cost or even speed etc. is to be optimized. More
specifically, a particular point \((x_{10}, x_{20}, \ldots, x_{m0})\) in a closed region, i.e., a domain plus its boundaries) that causes \(y(x_1, x_2, \ldots, x_n)\) to be either a maximum or minimum is sought.

In many cases the objective of an ordinary optimization problem cannot be stated very easily as it involves minimization or maximization of some functions with dissimilar entities. In the case of a boat design, there are so many variables, ratios and physical factors of different weights involved that it is virtually impossible to take care of all of them in the optimization. Order of magnitude analysis can be used to simplify the design algorithms to achieve faster result output.

The other problem in optimization is the occurrence of more than one extremums (maximas and minimas) within the function’s boundary and determination of the location and values of all the relative extrema, with the subsequent determination of the absolute extrema. Unfortunately, this goal is not attainable for all optimization problems.

### 3.2 Constraints:

The auxiliary conditions governing the upper and/or lower limits of the different variables and/or their functional relationship are known as the *constraints*. These constraints usually represent either system limitations or the physical and economic laws that the variables must satisfy.

In ship design there are a number of constraints that have to be used. Different types of vessels have different ranges of the ratios of their principal dimensions based on model tests and taken from existing vessels. If the optimization is run with no constraints, with power or cost to be minimised, it will probably produce a vessel with it's principal dimensions unacceptable by the designer, as the focus will be cost or power minimization and not the expected ratios of the ship dimensions. Say, for a simple example, we want to design a cylindrical tank. For a certain carrying capacity, the minimum surface area and in turn the minimum cost is achieved when the diameter is equal to the depth. This might lead to a dimension of the tank that might not be acceptable by the designer because of space problems.
or available material size restriction. This is why we must define the constraints very carefully and clearly. These constraints are usually defined in this study as lower and upper bounds.

One of the most common constraints used in ship design is that the weight is to be equal to the displacement. There are two types of constraints, namely, equality and inequality constraints. An example of an equality constraint is the weight-displacement relation, mentioned earlier. When the objective function must satisfy side conditions given as equations relating the independent variables, the optimum necessarily lies within the boundary of the feasible region. This final optimum result output may not be the minimum value in the feasible region, as it also has to satisfy the equality constraint/s. Inequality constraints are generally used for fixing the upper and lower bounds for optimization.

In short, most optimization problems require the maximization or minimization of a given objective function subject to a set of constraints and a set of bounds on the policy variables. Mathematical representation of this is as follows [22]:

\[
\text{Extremize } y (X) \text{ with respect to } X, \\
\text{Subject to} \\
g_i(X) \leq 0, \quad i=1,2, \ldots, k, \\
g_i(X) \geq 0, \quad i=k+1, k+2, \ldots, l, \\
g_i(X) = 0, \quad i=l+1, l+2, \ldots, m, \\
\text{and} \\
A \leq X \leq B,
\]

Where A and B are vectors whose components represent the lower and upper bounds, respectively, of the components of X.

Optimization finds a combination of the variables that would provide an optimum solution, while fulfilling a number of constraints, which in the case of ship design, may be stability, ratios of principle dimensions, draft restriction, maximum available propulsion power, stress level, human comfort, etc. In a mathematical formulation, the constraints are expressed as follows [22]:

\[
h_i(x_1, x_2, x_3, \ldots, x_p) = 0 \text{ ; equality constraints with } p \text{ variables} \\
g_i(x_1, x_2, x_3, \ldots, x_q) = 0 \text{ ; inequality constraints with } q \text{ variables}
\]
It is the inequality constraints that tend to be of more interest in ship design.

Optimization techniques need solving simultaneous linear equations and thus, the actual work involved in obtaining solutions increases enormously with a modest increase in the number of independent variables referred to as dimensionality. It may be virtually impossible to get a solution to a problem if the dimensionality exceeds four or five independent variable, even with a sophisticated computer.

3.3  **Sensitivity Analysis:**

According to most textbooks on optimization it is commented that, it is rarely possible to know all the constraint functions with absolute precision. There are a number of constant terms in design problems, which are related to the actual performance and are not predictable before operations begin. This study shows that there are a number of factors that influence the output result. Some of these factors are the upper and the lower bounds, the definition of the point around which the solution is sought etc. One of the reasons for this, could be the fact that, there may be more than one minima or maxima of a function that the optimization is dealing with. It is essential to check the relative weight of the influence of these individual uncertainties upon the calculated optimum results, called "**Sensitivity Analysis**".

Because of the time constraint, the sensitivity analysis was not done in this study. It will be suggestive to carry out a sensitivity analysis in the future to assess the affects of different constraints, bounds and initial point selection, on the output from the present study.

3.4  **Optimization in Ship Design:**

3.4.1  **Introduction:**

Out of the two most common ship design approaches, the *first* approach is based on the assumption that all design variables can be expressed in terms of vessel’s length and it
employs a procedure in which the length is increased sequentially until a feasible design is found. The second approach is the use of an optimization technique with the advantage of formulating a design problem as an optimization problem. It reduces the computation and opens the way to accommodate the required number of design variables and constraints, and the best solution, as defined by the objective function, is found automatically.

Murphy, Sabat and Taylor [1963], developed a systematic procedure to determine the optimum combination of design variables for a cargo ship that would satisfy the owner's requirements using the least cost criterion. Paterson [1984] used this least cost criterion to determine the best design variables for fishing vessels on the Pacific Coast. Mandel and Leopold [1966] used the least cost criterion, and adopted it to a computerized optimization technique that would determine the best vessel dimensions, based on the owner's requirements and the operating scenario for a cargo ship.

The first optimization study for fishing vessel design was done by Kupras in 1966. His model was based on Polish Factory Tankers and used methods similar to those previously mentioned. Hamelin [1971] used net parameters, vessel parameters, and least time in port as the criterion to determine the best vessel to be used for ground fishing on the Atlantic Coast.

3.4.2 Optimization Criterion:

It is very important to decide the criterion for optimization. It may be based on the operating cost, construction cost, service or maximum speed etc. For the Transglobal Expedition Project, the speed maximization could be the criterion for optimization with inclusion of power as equality constraint, as the truck power at the power takeoff, may be availed from the truck manual. The seakeeping constraints will have to be included as constraint for a safer and comfortable vessel design. The resulting design may be costlier than the one availed with cost minimization criteria. But the designer may justify the result highlighting the fact that the vessel is a special purpose vessel and is not intended to go into commercial service and have to cope up with the prescribed operational conditions. In the case of most other commercial designs, cost minimization may be the most commonly used criteria. In naval ship design, not
the cost, however, achieving it’s other military goals, such as speed and seaworthiness may be the prime factors for optimization.

RMS Pitch and Heave, Peak Pitch and Heave, Stability, minimum weight criterion may be entered as constraints, individually or in a group to check for their affects. With too many constraints, even a highly capable computer fails to find a feasible solution. Care has to be taken to select the lower and upper bounds and the initial point around which a solution is sought, as they have significant effect on the optimum result.

3.4.3 Optimization Software:

Because of the easy availability, user-friendliness, affordability and capability of Matlab software for mathematical operations, it has been found to be the most suitable software for optimization for the kind of requirements that the present study has. A brief on the Matlab Optimization toolbox and its options is given in Appendix-B.

One of the important advantages of using Matlab is its worldwide accessibility through Internet and option for contribution by different researchers in adding different optimization functions and tools. It also has help line through its web site.

To obtain the best commercial vessel, profit maximization and cost (both capital and running) minimization is the best criteria in the optimization technique. The most exciting aspect of this technique is that, it offers the possibility to design a ship, which possesses the best features directly from the factors selected, to make the economic assessment. In this process, the economic criterion is first defined, which is usually called the “Merit Function”, M, expressed as a function of a set of n number of variables as follows:

\[ M = f(x_1, x_2, x_3, \ldots x_n) \]

Where the variables \( x_1, x_2, x_3, \ldots \ldots x_n \) can influence the behavior of the function M.
3.5 Optimization procedures:

In this section, the different optimization procedures are discussed.

3.5.1 Optimization of Linear Function:

The solution of a linear set of equations with the computer is known as ‘linear programming’. Computer usually uses the “Simplex” technique for the solution, which usually involves an ingenious iterative procedure. The target is to maximize or minimize the merit function M (depending on its definition), while ensuring that the variables satisfy the constraint equations. Usually, the solution is started from an extreme point, called a basic feasible solution and moved to an adjacent extreme point so that the new value of the merit function can be compared with the basic feasible solution. This procedure is then repeated over the entire region within the feasible zone till a true extreme value is found. This procedure is comparatively simple if both the merit function and the constraint functions are linear. In the case of non-linear functions different methods have to be applied.

3.5.2 Optimization of Non-linear Function:

As most of the optimizations involve non-linear functions, the following procedures are most popular:

1. Use of derivatives: In the method of Determination of Exact Derivatives, the merit function can be directly differentiated. The root of the equation is found by numerical techniques, such as Newton and Raphson’s method. The main drawback of this approach is the need to obtain the first and second derivatives, but the method is very efficient when it can be used.

2. Gradient Techniques: The method of optimization requires knowledge of the second derivative and thus either the availability of an analytic function or of numerical differencing techniques to calculate the value. To avoid this drawback, a technique using only first derivatives has been developed. These techniques are called “Gradient
Techniques''; the most common ones are the “Steepest Descent”, “Partan” and “Conjugate Gradient” methods. Convergence using these techniques has been found to be very poor.

3.5.3 Searching Techniques:

There are two types of searching methods as follows:

1. Random Search: This technique uses a ‘random number generator’ and achieves the solution in a random manner.

2. Direct Search: The search is conducted in a more systematic manner that takes into account the ‘experience’ gained in the searching process. The main advantages of these methods are that no derivative need to be determined and all the evaluations are performed with the merit function. This process is less efficient in arriving at the desired answer.

3.5.4 Mathematical Programming Methods:

The following methods are used:

1. Linear Programming: In this process the merit function and all the constraint functions have to be linear functions. The constraints are treated as an essential part of the problem.

2. Non-linear Programming: With this process non-linear merit function and constrained functions can be handled.

3.6 Features of Matlab’s Optimization Toolbox

Most of the literature in this section has been taken from the manual for Matlab’s optimization toolbox. Optimization concerns the minimization or maximization of functions. It is used to solve complex design problems in order to improve cost, reliability, and performance in a wide range of applications. The Optimization Toolbox contains routines that implement the most widely used methods for performing minimization or maximization on general nonlinear
functions. Additional routines are provided for linear programming, quadratic programming, nonlinear least squares, and solving nonlinear equations. Combined, these routines provide a comprehensive set of tools for performing optimization.

3.7 Key Features of Matlab Optimization Toolbox:

The followings are the key features of Matlab’s Optimization Toolbox:

- Nonlinear equation solving
- Linear programming
- Quadratic programming
- Constrained nonlinear minimization
- Constrained linear least squares
- Minimax
- Multi-objective optimization
- Semi-infinite minimization

3.8 Overview of Matlab Optimization:

3.8.1 Flexible Optimization Environment:

With its extensive numerical capabilities and its fully extensible environment, Matlab is one of the ideal systems for setting up and solving optimization problems. The interactive nature of Matlab allows optimization problems to be easily refined and adapted, providing the user with valuable feedback and insight into a problem’s so-called "best" solution.

3.8.2 Nonlinear Optimization Routines:

The toolbox features a variety of nonlinear optimization routines that are designed to work with scalars, vectors, and matrices. The function to be optimized can be written as a Matlab function or as an expression. Default optimization parameters are used extensively, but can be
changed through an optional parameter variable. User-defined parameters can be passed directly to the functions, eliminating the need for global variables.

Gradients are calculated automatically using an adaptive finite-difference method, unless they are supplied in a function. You can check supplied gradients against those calculated via finite differences.

### 3.8.3 Matrix Minimization:

The optimization toolbox also includes routines for solving standard matrix minimization problems including linear programming, quadratic programming, and constrained least squares.

### 3.8.4 State-of-the-art Algorithms:

The toolbox provides the latest implementations of leading optimization algorithms:

- **For unconstrained minimization**: Nelder-Mead simplex search method and BFGS quasi-Newton method.

- **For constrained minimization, minimax, multi-objectives, and semi-infinite optimization**: variations of the sequential quadratic programming method.

- **For nonlinear least-squares problems**: Gauss-Newton and Levenberg-Marquardt methods.

- **For linear and quadratic programming, and constrained linear least-squares problems**: projection method.

The optimization routines offer a choice of algorithms and line search strategies.
3.9 Application of Matlab Software in the present study:

After setting up all the necessary equations (see) and steps for appropriate results, a Matlab design environment is prepared. Thirty-eight files were created in Matlab format to obtain desired results. The Matlab program for the calculation of different ship’s parameters uses these files. They are total weight, displacement, cost, power requirement, metacentric height, horizontal and vertical accelerations at fore, aft and midship, RMS heave, RMS pitch, ratios of peak heave and pitch, sea spectrum, threshold values at fore, aft and midship locations etc. (equations used from the references 1 to 11).

A file called, designopt.m (File-1) was then created along with an objective function file myfun.m (File-4) for cost minimization, and a constraint function file mycon.m (File-5) to represent the constraints. The cost functions are listed in Table 1.

The file myfun.m, defining the objective function can be of single goal objective function or multi-goal objective function. In a single-goal objective function, only one function or objective is minimized. In this study, a single objective function Cost.m (File 11) has been used. In multi-goal objective function, more than one objective function may be minimized or maximized. These multi-objective functions are defined as $F(1), F(2), \ldots, F(N)$ etc.

In the constraints file mycon.m (File-5), any combination of constraints may be used. Lower and upper bounds on different design parameters, such as $L/B$, $B/T$, $C_b$, $C_p$ etc., were also added in the file to get the optimal result within a desired range of these ratios and coefficients. The goal was to maximize the peak heave [File 16] and pitch ratios [File 17] for reduced motion response levels, and minimize the RMS heave [File 18] and pitch [File 19] and threshold values [File 28 to File 30] for crew comfort.

3.9.1 Optimization Model:

For the results presented in this paper, the following optimization model has been adopted. All of the other estimation formulae are identical to those used by Akinturk [1].
Objective function: Minimize cost subject to:

1. Displacement should be equal to Total weight
2. Design should be stable, i.e. GM > 0
3. RMS heave motion amplitude ≤ 0.7 m
4. RMS pitch motion amplitude ≤ 5° or 3° (for rule-IV)
5. Peak heave and pitch ratios ≥ 3.0 or ≥ 2.4 or ≥1.8, starting at the higher value for a feasible solution.

3.9.2 The cost function used is as follows:

The cost function is as collected by Bower and Akinturk and may not reflect current costs. The function is used as a criteria relative for the design. This is because the exact construction cost is not available from shipyards. The unit for cost is C$ and that for the ship’s parameters is feet. The cost function is given in Table 1.

The Matlab Optimization Toolbox is very user-friendly and may be manipulated with the design requirements (like cost or power minimization) and constraints very easily.

One of the objectives of this study is to use Matlab in ship design, using optimization techniques and bring this closer to the designer for their usage with confidence. This new platform will be referred to as Matship.

In the available software for ship designs, designers have to input the principal parameters from existing data or experience and the software is used to design the boat without optimization. Addition of design based on optimization would be a great benefit to the designers to achieve a certain goal like cost, power or speed optimization. It is obvious that the designer will have to have the motion and other design algorithms to use this process. In case there is no motion algorithm available, the designer may produce the same, using motion prediction programs like SHIPMO or PANMO at the preliminary design stage and finally can produce more accurate ones from the model test, if the option is available.
Table 1: Cost function

<table>
<thead>
<tr>
<th>Total cost</th>
<th>= capital cost + depreciation cost + insurance cost</th>
</tr>
</thead>
</table>

Where,

- **capital cost** = 1.3221 (hull cost + outfit cost + machinery cost + gear cost)
- **hull cost** = hullweight*11064 (for aluminum), hull weight in Long Tons
- **hull weight** = \( k \cdot 0.00002813(x^2 - 173.52x + 41880x - 0.0000158) \); where \( k = 1 \) (for steel) & .55 (for aluminum) and \( x = 100\left(\frac{L(B+D)}{3000}\right)^{4/3} \)
- **outfit cost** = 120.0*L^2
- **fuel cost** = (7500/speedmax)*gal/hr*2.5; Speed max in Knots
- **machinery cost** = 125*shp*2; where ship is shaft horse power
- **gear cost** = C$200,000
- **depreciation cost** = Capital cost*0.045
- **insurance cost** = \( \begin{cases} (\text{capital cost}) \times 0.02, & \text{if} \ (\text{capital cost}) < 500,000; \\ (\text{capital cost}) \times 0.015, & \text{if} \ 500,000 \geq (\text{capital cost}) \leq 2,000,000; \\ (\text{capital cost}) \times 0.01, & \text{if} \ (\text{capital cost}) > 2,000,000. \end{cases} \)

It is found that Matship gives quicker output (5 to 10 minutes) when no integration is required. Results for any parameters, like the Kimura’s value or RMS values, which needs integration, depending on the number of iterations required for an optimal solution, in some cases took more than 4/5 hours for the output. Time required for the output depends on the number of variables used, outputs desired, maximum number of iterations allowed, tolerance factor, the number of integration etc.
4 Previous Work Done at UBC

Various researchers have worked on ship design, based on the classical design spiral and using a nonlinear optimization technique. The procedure is based on solving an objective function and a series of equalities and inequalities with available computer codes.

At UBC, Bower in 1985 [11] used an optimization method to design an optimum fishing vessel for the West’s coast, to replace the aging vessels on the Pacific Coast of Canada. He focused on small trawlers, under 39m in length. The optimization criteria used by Bower for the design of a fishing vessel was based on an operational scenario, which means that the optimization criteria included, in some form, the income and expenses associated with the vessels. For any given set of design variables, a specific set of ship dimensions was calculated, which define the volume and displacement of the vessel. The sum of the weight groups was then subtracted from the displacement of the vessel at the desired draft, which gave the carrying capacity of the vessel including the payload. This payload, in terms of cubic space is termed as the fish hold capacity (FHC) and is used as a measure of a vessel’s earning capability. The total operating costs for the given sets of design variables were also calculated, which are the measure of the expenses associated with the vessel.

Bower used a cost merit function (CMF) [11] or an optimization criteria based on the ratio of the total annual operating cost divided by the fish hold capacity, the unit being dollars per cubic meter. This also implied that the best vessel will be the one with the lowest CMF, which is a vessel with the lowest operating cost and the largest fish hold. The merit function was therefore defined as:

$$CMF = f \left( \text{Ship Vector, Operational Scenario} \right)$$

Equation 1

Where, the ship vector is the ordered set \((L, B, T, C_m, C_p, V/\sqrt{L})\). Bower used the Coupler Optimization Technique, which originated in the NPL Library. This technique could work on an optimization function without requiring partial derivatives and has a guaranteed accurate convergence for optimization functions with less than 10 variables.
During the validation of the code, fuel cost from a simulated trip based on this algorithm was compared with an actual one and the difference between the actual and simulated costs was found to be less than 4%. It was further seen that this program predicted a fish hold capacity about 13% less than the actual fish hold capacity of an existing boat. It was thus concluded that the program might be used at the preliminary design stage to fairly accurately predict the design parameters of a fishing vessel.

Howard [8] tested the UBC series, fishing vessel hull forms developed at UBC, in the towing tank at BCRI Ocean Engineering Center. Table 2 shows a comparison of fishing vessel parameters of different series. The UBC series was developed by Calisal at UBC [5]. This series is specially designed for Canada's West Coast fishing industry, where there is restriction on the licensed length for a particular hold capacity. The significant feature of this series is its smaller L/B ratio and side bulb at the bow. Table 3 shows the geometric properties of the UBC series models. To study the motion of a vessel in waves, Howard used ITTC standard seakeeping hull model called the S-175 hull form. Accelero-meters, wave-probes, turbulence stimulators etc. as per necessity, were placed on the model and data were collected for 10 seconds at a sampling rate of 40 hertz. Howard did an extensive statistical analysis on these collected data and came up with algorithms to predict the heave and pitch motions and their spectral distributions. Howard compared the acquired data with that already existing for the ITTC model S-175 from University of Michigan and also with the one available from David Taylor Naval Ship Research and Development Center (DTNSRDC). Howard compared these data with the results from numerical modeling obtained by the use of both SHIPMO and PANMO, two different computer programs for the prediction of ship motion in waves (especially for slender hull form). Howard used curve fitting, sometimes more than one curve to fit to the spectral distribution of the data. From these spectra, both peak and RMS values of the ship motions, which governs the seakeeping criteria of the vessel, could be predicted.

A ship motion response algorithm was then developed at UBC, based on the seakeeping testing of the UBC series. This algorithm, which is probably the first algorithm for ship motion response for a series, was tested to see if the effects of variations in beam or draft on ship motions could be predicted for similar fishing vessels, not necessarily derived from the
UBC series [4]. The variations in motion response values obtained by the algorithm were compared with the experimental studies done for hull improvements in a towing tank. The same algorithm was also used for the design of fishing vessels by Akinturk [2] using an *Expert System shell*, called *Echidna*, developed at the Simon Fraser University.

Table 2: Comparison of fishing vessel parameters of different series

<table>
<thead>
<tr>
<th>Series</th>
<th>( C_b )</th>
<th>( L_m/B )</th>
<th>( B/T )</th>
<th>( V^{1/3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBC Series</td>
<td>0.53 - 0.61</td>
<td>2.6 - 4.0</td>
<td>2 - 3</td>
<td>3 - 4.47</td>
</tr>
<tr>
<td>BSRA</td>
<td>0.53 - 0.63</td>
<td>4.3 - 5.8</td>
<td>2 - 3</td>
<td>4.35 - 5.1</td>
</tr>
<tr>
<td>ITU</td>
<td>0.35 - 0.56</td>
<td>3.3 - 5.0</td>
<td>2 - 3.2</td>
<td>3.4 - 6.1</td>
</tr>
<tr>
<td>Webb</td>
<td>0.42 - 0.53</td>
<td>3.2 - 5.75</td>
<td>2.3</td>
<td>3.85 - 5.22</td>
</tr>
</tbody>
</table>

Table 3: Geometric Properties of the UBC Series Models

<table>
<thead>
<tr>
<th>Model</th>
<th>( L_m ) (m)</th>
<th>B (m)</th>
<th>T (m)</th>
<th>( L_m/B )</th>
<th>( B/T )</th>
<th>( C_b )</th>
<th>( C_p )</th>
<th>( C_M )</th>
<th>( L_m/V^{1/3} )</th>
<th>LCB/L ( m )</th>
<th>LCF/L ( m )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.552</td>
<td>0.507</td>
<td>0.204</td>
<td>3.06</td>
<td>2.49</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>3.36</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>2</td>
<td>1.319</td>
<td>0.507</td>
<td>0.204</td>
<td>2.60</td>
<td>2.49</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>3.01</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>3</td>
<td>2.017</td>
<td>0.507</td>
<td>0.204</td>
<td>3.98</td>
<td>2.49</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>4.00</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>4</td>
<td>1.552</td>
<td>0.507</td>
<td>0.170</td>
<td>3.06</td>
<td>2.99</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>3.57</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>5</td>
<td>1.552</td>
<td>0.507</td>
<td>0.255</td>
<td>3.06</td>
<td>1.99</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>3.12</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>6</td>
<td>2.017</td>
<td>0.507</td>
<td>0.204</td>
<td>3.98</td>
<td>2.99</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>4.25</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>7</td>
<td>1.552</td>
<td>0.507</td>
<td>0.204</td>
<td>3.06</td>
<td>2.49</td>
<td>0.531</td>
<td>.653</td>
<td>.813</td>
<td>3.53</td>
<td>3.47%</td>
<td>6.16%</td>
</tr>
<tr>
<td>8</td>
<td>2.017</td>
<td>0.507</td>
<td>0.204</td>
<td>3.98</td>
<td>2.49</td>
<td>0.531</td>
<td>.653</td>
<td>.813</td>
<td>4.20</td>
<td>3.47%</td>
<td>6.16%</td>
</tr>
<tr>
<td>9</td>
<td>1.552</td>
<td>0.507</td>
<td>0.170</td>
<td>3.06</td>
<td>2.99</td>
<td>0.531</td>
<td>.653</td>
<td>.813</td>
<td>3.75</td>
<td>3.47%</td>
<td>6.16%</td>
</tr>
<tr>
<td>10</td>
<td>1.552</td>
<td>0.507</td>
<td>0.255</td>
<td>3.06</td>
<td>1.99</td>
<td>0.531</td>
<td>.653</td>
<td>.813</td>
<td>3.27</td>
<td>3.47%</td>
<td>6.16%</td>
</tr>
<tr>
<td>11</td>
<td>2.017</td>
<td>0.507</td>
<td>0.255</td>
<td>3.98</td>
<td>1.99</td>
<td>0.615</td>
<td>.700</td>
<td>.878</td>
<td>3.71</td>
<td>3.74%</td>
<td>7.10%</td>
</tr>
<tr>
<td>12</td>
<td>2.017</td>
<td>0.507</td>
<td>0.204</td>
<td>3.98</td>
<td>2.99</td>
<td>0.531</td>
<td>.653</td>
<td>.813</td>
<td>4.46</td>
<td>3.47%</td>
<td>6.16%</td>
</tr>
<tr>
<td>13</td>
<td>2.017</td>
<td>0.507</td>
<td>0.255</td>
<td>3.98</td>
<td>1.99</td>
<td>0.531</td>
<td>.653</td>
<td>.813</td>
<td>3.90</td>
<td>3.47%</td>
<td>6.16%</td>
</tr>
</tbody>
</table>
While in principle any nonlinear problem solver that handles inequalities and equalities could be used to design a ship, *Echidna* with its built-in logic to handle constraints, and the ability to reduce the design space using constraint propagation, offered a superior programming environment. The main difference between an approach, based on design spiral or a sequential solution and a knowledge-based solution is that, in the latter procedure, there was no assigned solution path. The solution path is internally selected by the program based on the available information and constraints supplied to the knowledge base. The *Echidna* system enables the user to define constraints, ranges, tolerances and relationships for vessel design parameters such as length, beam, draft, displacement, etc. The built-in constraint processing facilitates the information flow between the parameters during each iterative cycle of the design, thus narrowing the feasible domains of the parameters.

The objective of Akinturk’s study was to reduce the motion effects on the crew or to increase crew comfort and safety. Akinturk studied the occupational safety on board fishing vessels around the world and performed statistical analysis on the accidents and related factors, in an attempt to understand the working conditions on board fishing vessels and expose the problem areas and detecting the pattern leading to an accident. Table 4 and Table 5 is the result from data [25] collected by him reflecting the types of vessels and contributing factors in casualties on board of vessels. It was concluded by Akinturk that the working condition on board fishing vessels need improvement and to do that, the primary factor, namely, ‘the motion of the

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Cargo</td>
<td>37</td>
<td>47</td>
<td>65</td>
<td>63</td>
<td>65</td>
<td>46</td>
<td>61</td>
<td>82</td>
<td>80</td>
<td>83</td>
</tr>
<tr>
<td>OBO</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Tanker</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>9</td>
<td>9</td>
<td>4</td>
<td>5</td>
<td>13</td>
<td>21</td>
<td>16</td>
</tr>
<tr>
<td>Tug</td>
<td>11</td>
<td>6</td>
<td>2</td>
<td>8</td>
<td>14</td>
<td>9</td>
<td>5</td>
<td>9</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Barge</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>Offshore</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>19</td>
<td>14</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>Fishing</td>
<td>59</td>
<td>66</td>
<td>66</td>
<td>43</td>
<td>75</td>
<td>83</td>
<td>105</td>
<td>109</td>
<td>112</td>
<td>80</td>
</tr>
<tr>
<td>Passenger</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>10</td>
<td>5</td>
<td>6</td>
<td>5</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>Ferry</td>
<td>7</td>
<td>8</td>
<td>3</td>
<td>6</td>
<td>5</td>
<td>17</td>
<td>9</td>
<td>4</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>Others</td>
<td>12</td>
<td>14</td>
<td>11</td>
<td>16</td>
<td>29</td>
<td>33</td>
<td>45</td>
<td>37</td>
<td>101</td>
<td>115</td>
</tr>
</tbody>
</table>
vessel", should be taken care of, which is influenced by factors like, the principal dimensions and speed of the vessels.

Two rules were developed by Akinturk [0] to reduce the heave and pitch motion RMS levels at various forward speeds and sea state conditions. Akinturk concluded that, it is possible to reduce the ship motion RMS values by adding specific rules to the design process. The effect of these rules on the principal dimensions of fishing vessels and the cost of the vessels (including the cost of construction) were also investigated for various hold volume requirements. The results suggested that these rules could be used for any small craft and the design method could also be included in an optimization procedure.

Table 5: 10 year averages of percentage values of primary contributing factors in casualties aboard all vessel types.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Human Factor</td>
<td>98</td>
<td>93</td>
<td>87</td>
<td>57</td>
<td>46</td>
<td>92</td>
<td>194</td>
<td>215</td>
<td>324</td>
<td>286</td>
</tr>
<tr>
<td>Environmental Conditions</td>
<td>13</td>
<td>14</td>
<td>9</td>
<td>7</td>
<td>1</td>
<td>6</td>
<td>11</td>
<td>8</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>Vessel Conditions</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>2</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>Equipment/Structural</td>
<td>11</td>
<td>13</td>
<td>18</td>
<td>13</td>
<td>5</td>
<td>9</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>Others</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td>21</td>
<td>36</td>
<td>36</td>
<td>95</td>
<td>168</td>
<td>98</td>
<td>22</td>
<td>33</td>
<td>17</td>
<td>11</td>
</tr>
</tbody>
</table>

Kimura et al [10] developed relationships between the maintenance of human posture and ship oscillatory motions. They suggested a threshold value, K, for the maintenance of human balance on a ship.

These rules, which were developed by Akinturk [1] and Kimura [10], have been used in this study as constraints in the design process and the results have been compared with available results for validation of the design process. This study applies an optimization technique in a Matlab environment and uses these ergonomic rules as constraints in the design process.

In the optimization model, the algorithms given by Howard are used for the prediction of heave and pitch motions. A UBC series' response function [6] is given as:
\[ \eta = 1 + A/(\exp(\alpha - \delta \omega_c) - 1)/(1 + \exp(\beta - \gamma \omega_c)) \]

Equation 2

Where \( \eta \) is the non-dimensional heave or pitch amplitude, given as \( (z_a/\xi_a) \) and \( (\Omega_a/k\xi_a) \) respectively, and \( A, \alpha, \beta, \delta, \gamma \) are the coefficients given for heave and pitch motions separately in appendix-A.

Based on the above formulation by Howard, Akinturk [0] generated a number of heave and pitch response functions by systematically varying the parameters used in Equation 2, and developed the following empirical formulae for non-dimensional circular frequencies of Peak Heave and Pitch Responses:

\[ \omega_{nz}^2 = \frac{(38.8668 - 11.5286 Cb - 5.2002 Lbp/B - 1.1912 B/T)^2}{(0.8652 - 0.2816 Fn)^2 (49.6354 - 20.4321 Cb - 6.8076 Lbp/B + 0.6441 B/T)^2} \]

Equation 3

\[ \omega_{n\theta}^2 = \frac{(2.4541 - 0.7280 Cb - 0.3283 Lbp/B - 0.0752 B/T)^2}{(0.9481 - 0.3086 Fn)^2 (2.8599 - 1.1773 Cb - 0.3922 Lbp/B + 0.0371 B/T)^2} \]

Equation 4

Where \( \omega_{nz} \) and \( \omega_{n\theta} \) are non-dimensional circular maximum response frequencies for heave and pitch motions respectively.

\[ \omega_{nz} \text{ or } \omega_{n\theta} = (\omega_z \text{ or } \omega_n)/\sqrt{(g/T)} \]

Equation 5

Where \( \omega_z \) or \( \omega_n \) are dimensional [radian/second] circular peak frequencies that maximize a ship's response functions for heave and pitch motions respectively.

In order to estimate some of the RMS motion characteristics of UBC series models, Akinturk also used a ship motion estimation program, called SHIPMO [13]. Using a regression analysis
on the data obtained from the runs of \textit{SHIPMO} for thirteen UBC series models he developed the following empirical formulae for the RMS Heave and Pitch response of the series:

\[
Z_{\text{rms}}/H_{1/3} = F_z + 0.74616424 F_z^2 - 0.46955309 F_z^3
\]

\textit{Equation 6}

\[
\theta_{\text{RMS}}/H_{1/3} \omega_{\text{max}}^2/\gamma = F_\theta - 0.46697339 F_\theta^2 + 0.56576473 F_\theta^3
\]

\textit{Equation 7}

Where, \(F_z = F_{z1} F_{z2} F_{z3} F_{z4} F_{z5} F_{z6}\) (see Appendix-A for \(F_{z1}, F_{z2}, \ldots, F_{z6}\)) and \(F_\theta = F_{\theta1} F_{\theta2} F_{\theta3} F_{\theta4} F_{\theta5} F_{\theta6} F_{\theta7}\) (see Appendix-A for \(F_{\theta1}, F_{\theta2}, \ldots, F_{\theta7}\)).

\(z_{\text{rms}}\) and \(\theta_{\text{rms}}\) are the RMS heave and pitch amplitudes respectively. \(\theta_{\text{rms}}\) is in radians. \(H_{1/3}\) is the significant wave height for a given sea state, \(\omega_{\text{max}}\) is the circular frequency of the peak energy in the energy spectrum.

Akinturk reported the statistics of accidents on board of different types of vessels around the world and concluded that accidents correlate well with platform acceleration levels. Based on the statistics, Chaplin and Burney [27] believe that the hazardous conditions on board of fishing vessels are partly because a fishing vessel is a moving platform to work on and exposes fishermen to the elements which are not generally experienced in other industries. Aboulazm [28], Andrew and Lloyd [29], Amagi and Kimura [30] and Pingree [31] agree that, ship motions are one of the main contributory factors for occupational accidents and/or adversely affect the crew performance on board. To reduce motion-related accidents in vessels, Akinturk included a seakeeping criterion in the preliminary design stage. McGreer, and Calisal [7], have shown that knowledge-based design can be a useful tool for the design of a fishing vessel. Akinturk used the motion algorithms from the work of Howard and Kimura et al and used an expert system environment called \textit{Echidna}, a logic programming language embedded in an object oriented framework developed at the Simon Fraser University’s Expert Systems Laboratory. He then calculated ship parameters for improved seakeeping qualities. In the first stage, he validated his program by using a monohull seiner, named Kynoc, operating off the pacific coast of Canada. The results validated well \textit{Echidna}’s knowledge based design.
system, as the procedure produced results similar to the Kynoc's actual data. As a second stage, he extended his study on a number of steel and aluminum vessels, and reproduced the trend reported by Calisal and McGreer [7].

This study takes this process a step further by using the ergonomic criterion directly in an easily available and user-friendly design platform. It highlights the basics of human factors in small boat design and proposes a method to include crew and/or passenger safety at the preliminary stage of the design. Four rules for ergonomic factors were used in this thesis. Out of these, first three rules were mentioned in Akinturk's [1] paper. Akinturk [1] used the first two rules in his design process. This study will include the third rule from Akinturk’s paper and add a fourth rule in the design process. These rules are termed as Rule I, Rule II, Rule III and Rule IV respectively and are described below:

4.1 Rule I: Shift of peak response frequencies for heave and pitch motions:

Rule Set I is based on the idea that when a dynamic system is excited near its natural frequencies the amplitudes of motion are expected to be large. Similarly, under normal circumstances, for a given sea state, a ship will experience large amplitude motions when the frequencies of the high energy waves coincide with the peak response frequencies of ship motions. With the implementation of this first rule, design of the vessel will be done in such a way that the peak frequencies of motions will not coincide with the frequencies of the high energy waves in the sea, for a specific design sea state (Figure 5). However, the amount of shift in the peak resonance and the peak energy is an important design factor and should be determined in conjunction with cost, ergonomic design rules and other physical rules of design such as the ones related to stability. A ratio has been defined to quantify the amount of spread between the frequency of the sea state’s highest energy waves, and the natural frequency of the vessel.
Figure 5: Shift of peak frequencies of sea waves and ship’s response

\[
\text{Ratio of Peak Frequencies} = \frac{\text{Ship response peak frequency}}{\text{Sea spectrum's peak frequency}}
\]

Equation 8

In this equation, the peak frequency for a sea state corresponds to the frequency of peak wave energy, whereas for the vessel it is the peak response frequency for a particular motion, e.g. pitches motion.

Three alternative rules for the “ratios” are given as:

1st alternative : \( \text{Ratio} \geq 3.0 \)
2nd alternative: \( \text{Ratio} \geq 2.4 \)
3rd alternative: \( \text{Ratio} \geq 1.8 \)

The first alternative some times proved to be infeasible, then the other alternatives were used in order of appearance in the above set of rules. Shift of the vessel’s natural frequency to higher frequencies resulted in somehow smaller vessels in general.
Application of Rule-I tends to produce a design with shallower draft, which makes the vessel stiff in heave motion. In Matship, constraint may be applied to the lower bound for the draft to avoid stiffness. New rules can be added to avoid such problems. This study uses Rule-I to validate the present platform for ship design, as results from previous studies are available with the application of this rule in ship design.

4.2 Rule II: Constraints on RMS heave and pitch motions:

This algorithm was developed to estimate the RMS values of heave and pitch motions of a vessel for a given sea state and impose constraints on some motion RMS values in the preliminary design. The rules incorporated are given below. The first rule is related to RMS values of heave motion amplitude and the second to the pitch motion amplitude. The aim is to reduce both of these values as much as possible. A RMS value is related to the total energy of the motion spectra.

\[
\text{RMS heave amplitude} < 0.7 \text{ [m]}
\]
\[
\text{RMS pitch amplitude} < 5.0 \text{ [degrees]}
\]

As expected, in most of the cases studied, Rule II increased design lengths and costs of the vessels.

4.3 Rule III: Constraints on K (Kimura's) values:

With the view to maintaining human balance on board a ship, Kimura et al [10] reported a relationship between a crewmember losing his/her balance on board and the ship's oscillatory motions. Based on experiments, they developed a criterion correlating the motion conditions on board of fishing vessels and the stability of a standing crewmember. It is assumed that the stability of human posture is strongly influenced by the experienced motions, and hence, frequency, amplitude and the relative direction of the ship in waves. After a series of experiments on full-scale vessels and a number of subjects, they suggested that when the threshold value K, defined below is greater than 0.22, crewmembers are no longer able to maintain their balances.
\[ K = M_1x + 0.1659M_1y + 0.1133M_1z < 0.22, \]

Equation 9

Where \( M_1 \) is the first moment of the power spectrum of the acceleration on the deck and the \( x \) (longitudinal), \( y \) (transverse) and \( z \) (vertical) are coordinate directions.

\( M_1 \) is defined as,

\[
M_1 = \int_0^\infty \omega S(\omega) d\omega
\]

Equation 10

Where, \( S(\omega) \) = acceleration power spectrum density function and, \( \omega \) = circular frequency.

In the formulation used in this study, the acceleration components due to gravity and in \( y \) direction, i.e. \( 0.1695M_y \), are neglected and the accelerations are in the ship frame of reference. The omitted terms in the \( y \) direction correspond to the accelerations due to sway, yaw and roll motions. This is based on the consideration that roll motion depends on the vertical location of center of gravity of the vessel and therefore it is difficult to deal at this stage of the design. Furthermore, the example designs were done for head seas conditions; the contribution to acceleration from roll, sway and yaw motions were also considered to be small and are neglected, although this may not be true for roll motion in real situations.

In the calculation of the acceleration terms, the following equations have been used:

Vertical acceleration

\[
\begin{cases}
\ddot{z} \cos \theta + \left( \frac{\text{Loa}}{2} \right) \ddot{\theta} \\
\ddot{z} \sin \theta
\end{cases}
\]

at the bow

at the midship

at the stern

(perpendicular to the plane of deck)

Horizontal acceleration

\[
\begin{cases}
\ddot{z} \cos \theta - \left( \frac{\text{Loa}}{2} \right) \ddot{\theta} \\
\ddot{z} \cos \theta
\end{cases}
\]

at the bow

at the midship

at the stern

(along the plane of deck)
Where \( \ddot{z} \) is the heave acceleration, \( \theta \) is pitch amplitude and \( \dot{\theta} \) corresponds to the amplitude of pitch acceleration.

Kimura’s Threshold Value concept is criticized for the fact that, human being usually orients themselves to have less effect of the motion in their body. In Kimura’s experiment the subject did not have the freedom to change their orientation. Also, though the effect of the surge and sway motions is greater compared to the effects of the other motions in Kimura’s Threshold Value equation, this two motions have been neglected in the present formulations. In the experiment for the statistical prediction of motions by Howard [8], the surge brake on the carriage was released before taking the readings and thus the effect on surge on the motion prediction algorithm is taken care of. In the formulation for the calculation of the first moment of the power spectrum the surge and sway motions are neglected. This is a flaw of the present formulations but this can be rectified or improved through further experimentation.

Akinturk et al [1,2] reported that inclusion of Rules I in the design produced, approximately 20% reduction on the average of the K values at the three locations of fore, midship and aft, for a speed of 0.01 [kn.] and sea state of 2. Whereas Rules II produced an approximately 10% reduction on the average for the same speed and sea conditions. For the same speed 0.01 [kn.] but sea state 5, the average reductions in K values were around 15% for Rules Set I and 2.5% for Rule Set II for all of the three locations on deck.

### 4.4 Rule-IV: Constraints on RMS heave acceleration and RMS pitch motion:

A new rule, named Rule-IV, has been added in this study to reduce the motions of the vessel for human comfort. This rule implements the limiting values on the RMS heave acceleration and that of pitch amplitude from the following study.

Outside UBC, Pingree [31] did a different study on the effect of ship motions on human comfort. He reported that, for personnel performing tasks of a continuous nature onboard vessels, the upper limits for tolerable RMS vertical acceleration in g (acceleration due to gravity), are used as 0.2 (at bridge) in USA, 0.16 (at 0.2L from forward perpendicular) in
Netherlands, 0.18 (at fore perpendicular) in Germany, 0.14 (weighted average) in the U.K., and 0.20 (at 0.2L from forward perpendicular) in Canada. It was also noted that there is an agreement of a seakeeping design criteria applicable to personnel for continuous exposure of 4.5 degrees roll, 3 degrees pitch and 0.18g vertical acceleration, all RMS. As, we are not considering the roll motion in our study, a constraint of 3 degrees on RMS pitch amplitude and 0.18g on RMS vertical acceleration will be used as Rule-IV.

That is Rule-IV can be expressed as follows:

\[ \text{RMS Vertical Acceleration} < 0.18g, \ g \text{ being acceleration due to gravity}, \]
\[ \text{RMS Pitch Amplitude} \ < 3.0 \text{ degrees} \]
5 Summary of the files used in "Matship"

A number of files have been created in this study, some of which estimates certain parameters of the design and the others are created to define the objective functions and constraints etc. Various Matlab Commands have been used to write these files. Constants and vectors are used in the different files to define parameters like density, gravity, free-board coefficient, hold capacities, sea states, significant wave heights, wave lengths and wave periods at different sea states from ITTC table etc. In addition, a number of empirical equations, like that for specific fuel consumption and weight group calculation etc. has been used. A number of algorithms produced by the previous researchers have also been used in the files.

The file designopt.m [File 1], is the main file in the design process, which is used for optimization purpose to get the values of different design variables based on the objective function ['myfun', File 4] and the constraint file ['mycon', File 5] containing the set of constraint functions.

The file we2w.m [File 2] returns w, wave frequency for a given encountered frequency, we.

The file sspec.m [File 3] reproduces the standard ITTC sea-spectrum for given seastate. Use of algorithm for a particular seastate of a specific route, if available, is better for more reliable results.

The file myfunc.m [File 4] is used to define the objective functions. F(1) and F(2) are the sub-functions of this objective functions, which are to be minimized by the optimization procedure. The function myfun, F(1) and F(2) are all functions of the variables listed within the bracket.

The file mycon.m [File 5] contains a list of constraints that the Matlab Optimization toolbox uses, to produce the results. There can be a group of equality and inequality constraints for the optimization procedure. The constraints listed as functions c(1), c(2).....ceq etc.(except c(7)) are different files, created separately to define the constraints. These constraints, usually
defines the lower and upper bounds, the range within which a result is targeted. Constraint c(7) is a constraint based on licensed length.

The file `cons_bt1.m` [File 6] defines the upper bound of the beam-draft ratio as a constraint in optimization, while, the file `cons_bt2.m` [File 7] defines the lower bound of the same.

The file `cons_lb1.m` [File 8] defines the upper bound of the length-beam ratio as a constraint in optimization, while, the file `cons_lb2.m` [File 9] defines the lower bound of the same.

The file `cons_disp.m` [File 10] defines the condition of equality, of the displacement and total weight of the vessel as an equality constraint in optimization.

File 11 to File 13 calculate the cost, displacement and total weight of the vessel, respectively.

File 14 to File 19 calculate the heave amplitude, pitch amplitude, peak heave ratio, peak pitch ratio, RMS heave and RMS pitch, respectively.

File 20 to File 25 produce the spectrum for horizontal and vertical accelerations at fore, midship and aft deck of the ship.

The file `rmsveraccaft.m` [File 26] calculates the vertical acceleration at the aft of the vessel and this is used for the constraint in rule-IV.

The file `stab.m` [File 27] estimates the transverse metacentric height of a loaded vessel, while the file `stablite.m` [File 28] computes the transverse metacentric height of a light vessel.

The files `tm01.m`, `tm02.m` and `tm03.m` [File 29 to File 31] compute Kimura's threshold value at the aft deck, midship deck and fore deck of the vessel respectively, from the first moment of power spectrum of the acceleration in longitudinal (x) and vertical (z) directions.

File 32 calculates the required power for the UBC series fishing boat.
The file Sminteg.m [File 33] defines the technique of integration by Simpson’s method.

File tmintp.m [File 34] is used in the optimization to find the minimum threshold value at different seastates and speed.

The file tmmax.m [File 35] is used in the optimization to find the maximum threshold value.

The file vcrit.m [File 36] calculates the maximum attainable velocity at any location (fore, midship or aft) at which Kimura’s threshold value will just be .22. This is useful in deciding the location of the workspace onboard the vessel for maximum comfort and safety of the crews.

The file okul.m [File 37] reads Kimura’s threshold value \( K \) calculated in the tm01, tm02 and tm03 files and produces plots of \( K \) versus speed for each sea state. This also calculates maximum attainable speed for a given operational sea state and produces plots for the same.

The file maintrgb.m [File 38] is the main file in the design process for Transglobal Expedition, which is used for optimization purpose to get the values of different design parameters for the ferry, based on the objective function and the constraint function. The objective function and the constraints in this process of optimization are written within this file, unlike the designopt.m file, where the objective and constraint functions are separate files.

The results from optimization in this study are based on cost minimization. Some of the important commands used in the above mentioned files are described in Appendix- C and some of the files created for Matship are shown in Appendix- D.

Figure 6 shows flow diagram of Input and Output variables in the Matship Process. The input variables may be grouped into owner’s requirements, objective functions, and initial points around which the solution is expected and the constraints to be satisfied by the output result.
**Input List:** (Owner Requirements)
- Hold Capacity
- Material type
- Design speed
- Design sea state

**Objective function:**
- **Single goal optimization**
  - minimize cost, or
  - etc.
- **Multi goal optimization**
  - minimize (RMS heave and pitch amplitudes and cost)
  - minimize (Cost and peak heave and pitch responses)
  - etc.

**Constraints:**
- Parameter Ranges
  - Length, beam, draft, Cb, L/B, B/T
- Displacement = Total weight
- Stability
- Optional Constraints
  - Rule I (peak heave and pitch response)
  - Rule II (RMS heave and pitch amplitudes)
  - Rule III (Threshold value)
  - Rules IV (RMS pitch amplitude and vertical acceleration)
  - RMS heave and pitch accelerations

**Ship Vector** (Initial Values)

**MATLAB Environment**

**Optimized ship vector**
L,B,T,Cb,Cp,Cm ...

Figure 6: Flow diagram of Input and Output variables in the *Matship* Process
6 Results and Discussions

6.1 Introduction:

In this chapter, the results obtained in this study are presented and discussed. As it is shown in the figures given below, the earlier works done by Bower [11], Calisal and McGreer [7] and Akinturk [1] were successfully carried a step further. Kimura’s rule that directly relates human balance to the platform motions was explicitly included into the preliminary ship design.

The results related to the validation of the code, developed in this study, are presented first. This new Matlab code was able to duplicate reasonably previous results reported by Akinturk [1,2]. Effects of Kimura’s rule on the various aspects of the designs, is then discussed. Finally, the effects of using different rules on the ship’s parameters are discussed.

It should be mentioned here that the results in this study are based on local optimization. Application of global optimization technique, if available and applied, may give different results.

It is important at this stage to discuss the differences in the design platform and features of Echidna and Matship, to better understand their result output and comparison with each other. Following section gives the important highlights of these two design platforms.

6.2 Differences Between the results obtained from Echidna and Matship:

Echidna’s solutions are based on constraint propagation and satisfaction. Every variable used in Echidna has a pre-assigned domain to it by the user. During the execution of Echidna, these domains are divided into sub-domains. A solution is then the union of the sub-domains selected for each variable, which satisfies the constraints. Hence, Echidna’s solutions are shown as intervals, representing these sub-domains. These selections are usually done randomly, unless the user specifies otherwise. Furthermore, there may be many and discrete
sub-domains for some of the variables, which constitute different solutions. *Echidna* results shown in the figures presented in this thesis do not always represent the all of the solution intervals, particularly if a variable has discrete sub-domains as solutions. By adding new constraints to the design knowledge base in *Echidna*, one can shrink these sub-domains to such an extent that it may eventually converge to a so-called minimum or maximum (optimum) solution.

There are no concepts of an objective function and an initial point in *Echidna*, the way they are defined and used in *Matship* optimization. Changing the number of variables and constraints during the flow of program execution in *Echidna* is possible and can be implemented easily as alternative models of the problem at hand. Rule Set I used in this study is a good example of defining alternative constraints. In a given time during a run, only one of the rules in Rule Set I will be active. If *Echidna* is not able to obtain a solution with that rule (or constraint), the other rules in the set will be tried one at a time. Therefore, in *Echidna*, it is difficult to know exactly, which of the alternative rules (constraints) were used in obtaining a solution. This is in contrast to optimization, in which the constraints can not be changed during optimization process.

Some of the design algorithms used in this study are empirical relationships among the certain parameters of the existing vessels. For instance,

\[
\text{length/beam} = 0.017742 \times \text{length} + 1.690323
\]

Such a relationship is very useful in obtaining an initial point for the optimization process to start in *Matship* or can be used as a constraint. On the other hand, if this relation is used in *Echidna*, it will always be considered as a constraint. Hence, any solution given by *Echidna* will satisfy this relationship. Therefore, although the equations used both in *Echidna* and *Matship* are the same, the way those equations are being used may be different sometimes.

With the above in mind, when formulating the design problem defined in *Echidna* as optimization (*Matship*), some of the design information (e.g. the above equation) is used to
define an initial point for optimization in Matship. The design philosophy was that the existing designs/vessels represent a working solution, hence, information similar to the above equation is used as an initial point for optimization. The algorithm used in Matship finds a local minimum in the neighborhood of the initial point, which satisfies the constraints. The number of constraints and the location of the initial point remain a subject for further research, as they may greatly affect the optimum solution found by Matship. However, Matship has the flexibility that the designer can change the initial point and the constraints to suit his/her requirements. In order to illustrate this point further, the constraint given in Equation 11 is removed and the results are compared in Figure 7 and Figure 8.

The constraint to be removed relates ship length to its hold capacity as follows:

\[ \text{Length} = 3.3 \times (28.5 \times \text{HoldCapacity})^{0.3369} \]

Equation 11

Where HoldCapacity is in long tons and Length is in feet. Figure 7 shows that by excluding this constraint, optimization results converged to a much shorter ship length. Since the cost function used in this study is proportional to length, a similar trend was obtained in the cost comparison (see Figure 8). Although such shorter vessels may not be desirable for different reasons, the final choice is still in the hands of the designer.

In the following sections, a more detailed comparison between Echidna and Matship results will be presented. As for the present discussion, both Echidna and Matship results with this constraint (Equation 11) on are similar to each other. Matship results for higher hold capacities converged to the lower bounds of the solutions given by Echidna (see Figure 7 and Figure 8). As mentioned above, Echidna solutions are intervals, thus, in the figures, there are two curves associated with those solutions: lower and upper bounds of the intervals.
Figure 7: Influence of including *Equation 11* as a constraint on the lengths of the designs. (Matship–A: Matship results with *Equation 11* included as a constraint into the design; Matship–B: Matship results without *Equation 11* included as a constraint into the design; Echidna–lower bound: lower bounds of Echidna results with *Equation 11* included as a constraint into the design; Echidna–upper bound: upper bounds of Echidna results with *Equation 11* included as a constraint into the design).

Figure 8: Influence of including *Equation 11* as a constraint on the costs of the designs. (Matship–A: Matship results with *Equation 11* included as a constraint into the design; Matship–B: Matship results without *Equation 11* included as a constraint into the design; Echidna–lower bound: lower bounds of Echidna results with *Equation 11* included as a constraint into the design; Echidna–upper bound: upper bounds of Echidna results with *Equation 11* included as a constraint into the design).
6.3 Validation of the process:

For the purpose of validation of the Matship platform, the fishing vessel “Kynoc” and the UBC Series Fishing Vessels were redesigned with this new process. The results were then compared with the available data for these two vessels, obtained from actual data and that from the studies by Bower [11] for “Kynoc” and Akinturk [1 and 2] for the UBC Series. These two vessels were selected because these studies were done in UBC and sufficient data is available for a proper comparison.

6.3.1 Redesigning of Kynoc with Matship:

As a first step of the validation process, a real fishing vessel, Kynoc was redesigned in Matship Optimization design program. The comparison is shown in Table 6. The Matship design version of Kynoc is very similar to the real vessel. In obtaining these results, two additional constraints were used in the design, which were L/B and B/T ratios set equal to the values of real Kynoc. Otherwise, Matlab Optimization would converge to a smaller vessel. Results from Matship and Echidna for the dimensions for “Kynoc” can be compared with those of real “Kynoc” in Table 6.

Table 6: Comparison of the results from Echidna and Matlab to the real Kynoc.

<table>
<thead>
<tr>
<th></th>
<th>Real Kynoc</th>
<th>Echidna</th>
<th>Matlab</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fishing Method</strong></td>
<td>Seiner</td>
<td>Seiner</td>
<td>Seiner</td>
</tr>
<tr>
<td><strong>Material</strong></td>
<td>Aluminum</td>
<td>Aluminum</td>
<td>Aluminum</td>
</tr>
<tr>
<td><strong>Design sea state</strong></td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td><strong>Hold capacity [LT]</strong></td>
<td>25</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td><strong>Speed [kn.]</strong></td>
<td>10.3</td>
<td>10.3</td>
<td>10.3</td>
</tr>
<tr>
<td><strong>Lwl [ft]</strong></td>
<td>56.7</td>
<td>56.7</td>
<td>58.71</td>
</tr>
<tr>
<td><strong>B [ft]</strong></td>
<td>20</td>
<td>20</td>
<td>20.71</td>
</tr>
<tr>
<td><strong>T [ft]</strong></td>
<td>4.92</td>
<td>4.92</td>
<td>5.09</td>
</tr>
<tr>
<td><strong>Cb</strong></td>
<td>0.519</td>
<td>0.519</td>
<td>0.52</td>
</tr>
<tr>
<td><strong>Cm</strong></td>
<td>0.728</td>
<td>0.728</td>
<td>0.728</td>
</tr>
<tr>
<td><strong>Cp</strong></td>
<td>0.712/.711/.717</td>
<td>0.714</td>
<td></td>
</tr>
<tr>
<td><strong>Displacement [LT]</strong></td>
<td>82</td>
<td>83.5</td>
<td>92.01</td>
</tr>
<tr>
<td><strong>Required power [HP]</strong></td>
<td>420/134/378</td>
<td>259</td>
<td></td>
</tr>
<tr>
<td><strong>Cost [Can $]</strong></td>
<td>N/A</td>
<td>1.20E+06</td>
<td>1.29E+06</td>
</tr>
</tbody>
</table>
In *Matship* process for “Kynoc” speed was taken as 10.3 knots, similar to the speed of the actual “Kynoc”. Other inputs in the design process were Cm, seastate, hull-material etc., similar to that of real “Kynoc” [Table 6]. Upper bound for Cb was taken as 0.52. There is a large discrepancy in the required horsepower value between the Matship calculated one and that of the real “Kynoc”. *Matship* result is within the range of *Echidna* result, while the main propulsion unit in real “Kynoc” is around 38% more than the *Matship* value. In the real Kynoc, part of the power from the main propulsion unit might have been utilized for electric power generation, running pumping, fishing gears and other systems or rated for operating condition such as fishing gear operation.

6.3.2 *Redesigning of UBC Series Fishing Vessels with Matship:*

The UBC Series was chosen for the availability of its vast database of motion characteristics, from a number of studies done at UBC.

The work done by Calisal and McGreer [7], Howard [8] and Akinturk [1] were based on UBC Series. As a continuation of their work, this study also used the UBC Series design algorithms. However, the concept developed here is independent of databases coded and can be applied to any other series.

For the figures presented in this chapter, a set of design parameters were reported for hold capacities of 10, 25, 50, 75, 100, 125, 150, 175, 200, 250 and 300 long tons. Whenever applicable, the design and operational sea states are Beaufort Seastate 5, unless otherwise mentioned.

Figure 9 compares lengths of the designs obtained by Akinturk to those obtained by *Matship*. In both cases, there were no sea keeping criteria included in the design procedure. In *Matship*, license length constraint was not implemented. In fact, this constraint is effective only after 75 long ton hold capacity for *Echidna*-designs and 175 long ton hold capacity for *Matship* designs. *Matship* designs are shorter in length compared to *Echidna* results at 10 tons hold capacity and from 75 long tons hold capacity through around 150 tons hold capacity and are
very similar in other ranges. Unlike Echidna results, the variation in length for Matship results is smoother.

Figure 9: Comparison of length values obtained by Matlab and Echidna for UBC Series.

Figure 10: Comparison of beam values obtained by Matlab and Echidna for UBC Series.

Comparison of the beam values of Echidna designs and Matship designs is given in Figure 10. The differences between the two beam values are very small, except within the range of hold capacity from 250 long tons to 300 long tons.
Figure 11 Comparison of draft values obtained by Matlab and Echidna for UBC Series.

Figure 11 shows the comparison of drafts obtained from the Echidna and Matship. For Matlab Optimization, a smoother variation of draft with hold capacity is observed. The differences between the two draft values are much smaller compared to the differences for length values. This figure shows that both the results match closely with each other, especially in the range of 50 to 250 long tons hold capacity. The solutions from optimization are fairly within the ranges given by Echidna.

Figure 12 shows comparison between displacement from Matship to those obtained by Echidna. The results from Matship matches very closely to the lower bound of the Echidna results. Cost optimization applied in Matship results in smaller vessel designs. Up to 150 tons hold capacity, the Matship results are within the bounds of Echidna results and beyond 175 tons hold capacity its slightly below the lower bound of Echidna.

Figure 13 shows comparison between costs from Matship to those obtained by Echidna. There is a slight reduction in cost produced by Matship below the lower bound of Echidna results up to hold capacity of 150 long tons. Beyond 300 tons of hold capacity Matship produced values higher than the upper bound of the result from Echidna.
6.4 Effects of Kimura’s Crew Safety Criteria on the designs

In this section, the effect of the direct inclusion of Kimura’s Crew Safety Criteria on different ship parameters is analyzed and discussed. In the comparisons in this section, the percentage differences were calculated using the following formula:
Percentage difference = 100 * (Constrained value - Unconstrained value) / Constrained value

The control value is the unconstrained value.

Figure 14 reflects the percentage increase in waterline length at different hold capacities, comparing the results with no constraint and the one with Rule-III. Average increase in length is 9.92%. Graph shows percentage increase in length increases with increase in hold capacity. This is because of the equation for vessel’s length, used for UBC Series fishing vessels, is a nonlinear function of hold capacity.

Figure 14: % changes in Length due to Kimura’s Crew Safety Criteria (Rule-III)

Figure 15 shows the percentage difference in beam values, the average being about 10%. Note the non-linearity of the curve. As ship’s beam is one of the major contributing factors in the cost calculation, the optimization procedure always tries to minimize the cost, as far as possible. In case the procedure can not converge to a result for a certain hold capacity, Matship increases the beam to get to an acceptable solution, still minimizing the cost.

Two significant graphs are Figure 16 and Figure 17, showing the percentage differences for draft, T and the block coefficient, Cb. Note the opposite nature of the two graphs. While the
inclusion of crew safety increases draft, the Cb decreases and vice-versa (exception is for 75 tons hold capacity for reasons unknown). The optimization shows the tendency to decrease draft for cost minimization and to compensate for the total buoyancy Matship increases the Cb of the vessel. On the other hand, it was noted that, if power minimization is the objective function, it optimizes all the variables that affect the power requirement for the propulsion, such as draft, Cb etc.

Figure 15: % changes in beam due to Kimura’s Crew Safety Criteria (Rule-III)

Figure 18 shows percentage increase in the cost of the vessel versus hold capacity with and without crew safety constraint. Figure 19 shows the increase in total weight, for the above inclusion. One can see that the increase in cost is directly related to the increase in weight. Increase in the cost of machinery, out fitting etc. are other contributing factor to the total increase in the vessel’s cost. Average increase in cost is 7.45%. The graph shows that the cost increases from 2% at 25 tons hold capacity to a maximum of 14.4% at 75 tons hold capacity; then again decreasing to 2% at 300 tons hold capacity.
Figure 16: % changes in draft due to Kimura's Crew Safety Criteria (Rule-III)

Figure 17: % changes in block coefficient due to Kimura's Crew Safety Criteria (Rule-III)
Figure 18: % changes in costs due to Kimura’s Crew Safety Criteria (Rule-III)

Figure 19: % changes in total weight due to Kimura’s Crew Safety Criteria (Rule-III)

Figure 20 shows the percentage decrease in power requirement for the inclusion of crew safety constraint. Power requirement is a major contributing factor and a variable in cost calculation. Thus, cost minimization criteria seems to minimize the power as a secondary factor. The power requirement largely depends on the length, draft and Cb of the vessel. Thus, the increase or decrease of these parameters at any holds capacity will affect the power
requirement of the vessel. It should be noted that the power calculation is based on calm water resistance. Highest decrement in calm-water power requirement of 35% is marked at 100 tons hold capacity and the lowest 15.64% being at that of 300 tons. The average decrease is 23.8%, which is a very significant decrease. Lowering of ship motions with seakeeping constraints should also increase the propulsive efficiency of the vessel and in turn decrease the power requirement, at sea. These results are based on cost minimization. As the cost of propulsion unit, which is directly proportional to the required power, is one of the factors in the total cost of the vessel, Matship tries to minimize the power also, in turn.

Figure 21 displays the percentage decrease in RMS heave motion, highest value of 8.46% being at 75 tons hold capacity and the lowest decrease of .56% is at 150 tons hold capacity. The higher decrement of RMS values means lower energy in its motion spectrum.

Figure 22 reveals that the RMS pitch motion is reduced up to a maximum of 259% at 75 tons hold capacity with the inclusion of Rule-III. When constraint on the Kimura’s threshold value is used, the present procedure tries to minimize the energy level in the motion spectrum, largely contributed by the RMS values, through the selection of optimum ship’s parameters.

Figure 20: Percentage changes in required power due to Kimura’s Crew Safety Criteria (Rule-III)
Figure 21: % changes in RMS heave amplitude due to Kimura’s Crew Safety Criteria (Rule-III)

Figure 22: % changes in RMS pitch amplitude due to Kimura’s Crew Safety Criteria (Rule-III)
Figure 23 and Figure 24 show the percentage difference produced in Peak Heave and Peak Pitch Ratios by the inclusion of Rule-III. Both graphs are very similar in shape and values at different data points. The definition of peak ratio is given in Equation 8. The higher these Peak Ratios, better is the design in terms of seakeeping characteristic. These two figures show that, though inclusion of Rule-III reduces the acceleration level in the vessel, it decreases the peak ratios till about 175 tons hold capacities, and then increases the ratios for hold capacities higher than 175 tons.

Figure 25 reflects the effect of the inclusion of Rule-III on the Kimura’s Threshold Value, K. The constraint on K (Rule-III) is chosen to be less then 0.22. Kimura’s study suggests that at K value higher than .22, it is difficult for the crew on board to maintain their posture and work safely and comfortably. Reduction in Kimura’s constant varied from 18.62% at hold capacity of 300 tons to about 100% at that of 75 ton. It should be marked in Figure 20, Figure 21 and Figure 22, that at the hold capacity of 75 tons, the percentage decrease in RMS values are maximum, along with highest saving in power requirement. This is made with the sacrifice of cost, as is evident from Figure 18. Using this kind of analysis, it is also possible to predict the most economical vessel as a standard for a series, in terms of initial cost or operational cost. It
seems that though the vessel with 75 tons is relatively more expensive in initial cost, it could be the most economical one in terms of operating cost from the point of power savings.

Figure 24: % changes in peak pitch ratio due to Kimura’s Crew Safety Criteria (Rule-III)

Figure 25: % changes in Kimura’s threshold value due to Kimura’s Crew Safety Criteria (Rule-III)
6.5 Implications of varying Kimura’s Crew safety Threshold Value:

Kimura et al [10] suggested a value of 0.22 for the threshold, after which humans lose their balance. However, in their experimental data, the value of 0.22 was not a clear boundary. Thus, in this section, the effects of varying threshold value to 0.15 and 0.30 on ship parameters are presented.

As the threshold value is decreased, for a given owner requirements such as hold capacity and design sea state, vessel’s length is increased (see Figure 26). The curves in this figure represent how design length varies with hold capacity for a given threshold value. The three curves follow a similar trend and are almost parallel to each other. In Figure 27, the percentage changes are plotted against hold capacity. In calculating the percentage values, the following equation is used and the control design corresponds to the threshold value of .22.

\[
\%\text{change} = \frac{100 \times \text{Parametervalue for threshold 0.15 or 0.30} - \text{Parametervalue for threshold 0.22}}{\text{Parametervalue for threshold 0.22}}
\]

Equation 12

For hold capacities larger than 100 [LT], percentage change in length is almost constant for both threshold values of 0.15 and 0.30 resulting in approximately 5% increase and approximately 4% decrease, respectively.

Figure 28 shows the changes in beam values due to the changes in crew safety threshold value. A similar trend in Figure 26 is observed that is increasing threshold values result in narrower beams. However, the same is not valid for the draft values, as seen in Figure 29. For threshold value of 0.3, all of the designs produced by Matship have greater drafts than those designs corresponds to the threshold value of 0.22. Reducing the threshold value to 0.15, on the other hand, resulted in greatest draft values (this is a common rule of thumb, used by Naval Architects to have “good seakeeping” characteristic), among the three threshold cases, up to approx. 150 [LT] hold capacity. Past 150 [LT] hold capacity, the designs for threshold value of 0.15 have the shallowest drafts among the three groups (see Figure 29).
Figure 26: Effects of varying threshold value on length

Figure 27: % change in length due to the changed threshold value
Effect of $K$ constraint on Ship's Beam

Figure 28: Effects of varying threshold value on beam

Effect of $K$ constraint on Ship's Draft

Figure 29: Effects of varying threshold value on draft

Increasing threshold value did not seem to affect the block coefficient greatly in Figure 30. However, one should also keep in mind that there was an upper bound on the block coefficient ($C_b$) such that $C_b \leq 0.7$. Decreasing the threshold value from 0.22 to 0.15 resulted in lower block coefficient values for hold capacities up to approx. 150 [LT].
Figure 30: Effects of varying threshold value on block coefficient

Figure 31: Effects of varying threshold value on displacement

Figure 31 shows the resultant changes in displacement due to the changes in the threshold value. The same data are presented in
Figure 32 as percentages. Increasing the threshold value slightly decreases the displacements over the whole hold capacity range considered, from approx. 7.5% for smaller hold capacities to approximately 2% for larger hold capacities. On the other hand, reducing the threshold value increased the displacements from approx. 12% to 3% as the hold capacity increases.

Figure 32: % change in displacement due to the changed threshold value

Figure 33: Effects of varying threshold value on cost
Decreasing the threshold values, that is improving the comfort level due to motions, has implications in terms of the vessel cost (see Figure 33). According to Figure 34, which shows the percentage changes in cost due to changes in the threshold value, improving the motion levels for better crew safety has a cost penalty of approximately 6% to 10% the average.

![Figure 34: % change in cost due to the changed threshold value](image)

6.6 **Comparison of various Crew Safety Rules:**

This section will discuss about the effects of the different crew safety rules on different ship parameters. It should be marked that the objective function in this study is cost minimization only, with seakeeping criterion as a constraint, along with bounds on ratios of ship’s principal dimensions and the process can select the ship’s parameters within these limits and constraints. The process may be run with any objective function or group of functions to be minimized or maximized.

Figure 35 shows the effect of the constraints on the cost. For the UBC Series, it is seen that the Rule I, and IV did not have any effect on the cost when compared to the cost of the design with no constraint, while inclusion of Rule III increased the cost of the vessel by about 8.61 percent only. This small increase in cost with the Rule-III constraint may justify its inclusion.
in the design process for the safety and comfort of the crews on board and more economical overall operation of the vessel.

Figure 35: Effect of Rules on Ship’s Cost

Figure 36 shows the effect of the constraints on the displacement. For the UBC Series, it is seen that the Rule I, and IV did not have any major effect on the cost compared to the one with no constraint, while inclusion of Rule III increased the displacement of the vessel by average of 6.79 percent only.

Figure 36: Effect of Rules on Ship’s Displacement
Figure 37 through Figure 40 show the affect of the constraints on length, beam, draft and coefficient of buoyancy (Cb). The results with Rule-I and IV merged with the one with no constraint for Length, Beam and Draft, with relatively small variation (1.15%) within 50 tons to 150 tons hold capacity range, while the Rule-III produced an average increase in length by 9.52%. It is also seen from Figure 38 that the results for beam with no constraint and Rule-I and IV show similar trends as Figure 37, with 8.02% increase for the inclusion of Rule-III. Figure 40 shows that values obtained with no rules merged with that obtained with Rule-I and Rule-IV for Block Coefficient and for most of the region has converged at the upper bound of 0.7, defined in the optimization process. Run with Rule-III, that is, threshold value less than 0.22, at the aft section of the vessel resulted in smaller vessels up to hold capacity of 175 tons and beyond this hold capacity the values of Block Coefficient converged at its upper bound of 0.7, defined in the optimization process. It is significant to note the sudden change in draft [Figure 39] and Cb [Figure 40] at hold capacity of 175 tons. This is due to the process of optimization with cost minimization. It increased the Cb to get the desired buoyancy and reduced the draft for cost minimization. Both these changes (to buoyancy and draft) might have adverse effect on seakeeping characteristics of the vessel. This may have happened due to the restriction on the length of the designed vessel based on the hold capacity. From this results it may be stated that the relaxation on the license length based on hold capacity under some fisheries regulations at different part of the world may help in design of a safer vessel.
Effect of Rules on Ship’s Beam

Figure 38: Effect of Rules on Ship’s Beam

Effect of Rules on Ship’s Draft

Figure 39: Effect of Rules on Ship’s Draft
Inclusion of Rule IV resulted in reduction in K values for the hold capacities from 50 to 125 tons as shown in Figure 41 as compared to the result from no rules and that with the inclusion of Rule-I and merged for other hold capacities. Results from the inclusion of Rule-I merged with that without the inclusion of any rules. Inclusion of Rule-III, that is, Kimura’s threshold value less than 0.22 at the aft section of the vessel resulted in upper limit of 0.22 of the threshold value for all ranges of hold capacity. Inclusion of Rule-III seems to be essential for safer and comfortable working condition. Using this constraint one can obtain, for a given operational sea state, what the maximum operating speed is, so that at a certain location on the ship, K will be less than 0.22. Similarly, at the design stage, certain work places can be located at those locations on the ship, where K is less than 0.22 for a given operational sea state and speed scenario. This will indicate to the designer the most suitable location for a working area in the vessel.

Figure 42 shows the effect of the constraints on the Required Propulsive Power of the vessel. For the UBC Series, it is seen that the Rule I and IV did have very insignificant effects (slight difference in the range of 75 tons to 150 tons hold capacity) on required power compared to the one with no constraint. Inclusion of Rule III decreased the required power of the vessel by an average value of 18 percent, for this particular series. The percentage of power reduction is seen to be increasing with the hold capacity. As, the machinery cost is a major contributory
factor of the total cost (which is being minimized by the process), and is dependant on the required power, the power minimization is also achieved indirectly, because of cost optimization.

![Effect of Rule-III on Threshold Value at aft of vessel](image)

Figure 41: Effect of Rules on Kimura’s Threshold Value

![Effect of Rules on Ship’s Required Power](image)

Figure 42: Effect of Rules on Ship’s Required Power

Figure 43 to Figure 45 show Kimura’s Threshold Values, $K$ at different sections of the vessel, namely, fore, midship and aft of the vessel [250 tons hold capacity] at different seastates and ship’s speeds. The Kimura’s constraint (Rule III) is shown as “$K_{\text{limit}}$".
Figure 43: Kimura's Threshold values at the forward of the vessel [250 tons hold capacity] at different seastates and ship speeds.

Figure 44: Kimura's Threshold values at the midship of the vessel [250 tons hold capacity] at different seastates and ship speeds.
Figure 45: Kimura’s Threshold values at the aft of the vessel [250 tons hold capacity] at different seastates and ship’s speeds.

It is clear from these graphs that with the lower sea condition (seastate), K value gets lower, that is, the vessel becomes more comfortable and safer for the crews and passengers onboard the vessel. It is also seen that at a particular seastate and ship’s speed, the aft section of the vessel shows lesser K value compared to the other two sections. This obviously tells the designer and operator about the extent of motion and acceleration levels at different parts of the vessel as a guideline, for the placement of work areas. These results are desired results from the present design procedure.

Figure 46 shows the maximum attainable speed (maximum safe speed envelop) with crew safety for the fore, mid ship and aft sections of the optimized design for 250 ton hold capacity at different sea states. This graph is basically produced from the previous graphs (Figure 43 to Figure 45). It may be observed that the maximum attainable speed at the aft section of this particular vessel is greater than the other sections. This also tells like the previous graphs that, for this particular vessel, it will be desirable to locate the working area (fish landing, sorting etc.) at amidships or the aft part of the vessel for the crew safety and comfort.
From the above results and discussions, it is clear that Rule-III and Rule IV, along with the other seakeeping criteria, can be comfortably used in the design process for the design of a safer and a comfortable vessel, with marginal increase in cost. In the long run, from a better operating condition on board the vessel due to this new design process, higher return than the additional initial investment, due to inclusion of crew safety criteria, may result. The difficult part is that, this indirect return is very difficult to assess in monetary terms.
7 Conclusions

After this extensive study with a new design process and platform, the following conclusion may be drawn:

a. An ergonomic factor is directly included in a computer aided boat design program. This study shows that the design of a small craft can be computerized and design nodes, such as crew-safety or acceleration levels can be included within the preliminary design procedure, using a Matlab like environment.

b. In this study the ergonomic factors are applied in the design process as constraints to get optimal dimensions of the vessel. In addition safe operational speed-seastate ceiling is derived.

c. Through the study on different series of vessels in terms of their ratios of principal dimensions and form coefficients, algorithms may be calculated for use with any type of vessel.

d. Minimization of cost or power requirements may be included in the design, using optimization techniques. Cost minimization with ergonomic constraints included in the optimization process, in some cases generated relatively less expensive but ergonomically improved boats. In general, ergonomic constraints suggest a larger vessel.

e. On work platforms, complete inclusion of human factors in design may affect the total cost and the principal dimensions of the vessel. However, human factors may be a very important factor to be considered in the design process, especially for workboats, where many manual operations are carried out on the deck.

f. User friendly, readily available and affordable design software may help boat designers include the human factors in the design and provide optimization techniques to design possibly more efficient and safer vessels.
g. The results obtained can be extended to the design of any small craft and the above rules could also be implemented in a design process based on a general optimization procedure.

h. The present length restrictions on fishing vessels seem to be against crew safety and may result in increased accidents on board.
8 References:


32. Wilde, D. J. and Beighter, C.S. "Foundation of Optimization".
9 Appendix- A

Coefficients for non-dimensional Heave and Pitch motions:

For heave:

\[ A = 0.8591 \times \frac{F_n}{L_{pp}} \left( \frac{T}{B}\right)^2 + 1.5804 \times \frac{F_n}{Cb} \left( \frac{T}{B}\right) \]
\[ \alpha = 1.816 + 31.42 \frac{Cb}{T/L_{pp}} \]
\[ \delta = 16.48 \frac{T}{L_{pp}} - 0.7485 \frac{T}{Cb} \left( \frac{T}{B}\right) \]
\[ \beta = 6.436 \frac{Cb}{L_{pp}} \left( \frac{T}{B}\right)^2 + 163.1 \frac{Cb}{T/B} \]
\[ \omega = 2.87 \frac{Cb}{L_{pp}} \left( \frac{T}{B}\right)^2 + 40.36 \frac{Cb}{T/B} \]

For pitch:

\[ A = 1.1313 \frac{F_n}{L_{pp}} \left( \frac{T}{B}\right)^2 + 0.9264 \frac{F_n}{T/Cb} \left( \frac{T}{B}\right) \]
\[ \alpha = 0.7904 + 45.47 \frac{Cb}{T/L_{pp}} \]
\[ \delta = 16.82 \frac{T}{L_{pp}} - 0.8621 \frac{T}{Cb} \left( \frac{T}{B}\right) \]
\[ \beta = 18.34 \frac{Cb}{L_{pp}} \left( \frac{T}{B}\right)^2 + 37.35 \frac{Cb}{T/B} \]
\[ \gamma = 7.211 \frac{Cb}{L_{pp}} \left( \frac{T}{B}\right)^2 + 3.039 \frac{Cb}{T/B} \]

Coefficients for non-dimensional RMS Heave and Pitch Responses:

\[ F_{z1} = 0.17244257 - 0.10902408 \frac{B}{L_{bp}} - 0.062345772 \left( \frac{B}{L_{bp}}\right)^2 \]
\[ F_{z2} = 0.92116235 + 9.26182543 \frac{T}{L_{bp}} - 17.01219503 \left( \frac{T}{L_{bp}}\right)^2 \]
\[ F_{z3} = 1.25445302 + 0.015068303 \]
\[ F_{z4} = 0.37459542 + 0.33475563 \frac{F_n}{L_{bp}} + 0.69543449 \left( \frac{F_n}{L_{bp}}\right)^2 - 0.62937284 \left( \frac{F_n}{L_{bp}}\right)^3 \]
\[ F_{z5} = 0.51156287 + 1.67664716 \frac{\lambda}{L_{bp}} - 0.60058403 \left( \frac{\lambda}{L_{bp}}\right)^2 + 0.065781060 \left( \frac{\lambda}{L_{bp}}\right)^3 \]
\[ F_{z6} = 1.27331482 - 1.02535654 \left( \frac{NH}{L_{bp}}\right)^{1/3} + 0.56388486 \left( \frac{NH}{L_{bp}}\right)^{1/3} \]
\[ F_{z7} = 0.465143 - 2.343704 \left( \frac{\lambda}{L_{bp}}\right)^{1/3} + 3.482758 \left( \frac{\lambda}{L_{bp}}\right)^{1/3} \]
\[ F_{z8} = 1.323205 + 13.947877 \left( \frac{NH}{L_{bp}}\right)^{1/3} - 26.389156 \left( \frac{B}{L_{bp}}\right)^2 \]
\[ F_{z9} = 0.346817 - 24.620275 \left( \frac{T}{L_{bp}}\right) - 37.557687 \left( \frac{T}{L_{bp}}\right)^2 \]
\[ F_{z10} = 0.044729 + 1.003151 \]
\[ F_{z11} = -0.182212 - 0.296472 \frac{F_n}{L_{bp}} + 0.059477 \left( \frac{F_n}{L_{bp}}\right)^2 - 0.245965 \left( \frac{F_n}{L_{bp}}\right)^3 \]
\[ F_{z12} = -0.181446 + 4.924184 \frac{\lambda}{L_{bp}} - 0.834965 \left( \frac{\lambda}{L_{bp}}\right)^2 + 0.074877 \left( \frac{\lambda}{L_{bp}}\right)^3 \]
\[ F_{z13} = 2.16213325 - 2.78020823 \left( \frac{NH}{L_{bp}}\right)^{1/3} + 1.54756589 \left( \frac{NH}{L_{bp}}\right)^{1/3} \]
It consists of:
(a) unconstrained optimization example
(b) constrained optimization example
(c) constrained example using gradients
(d) bounded example
(e) equality constrained example
(f) unconstrained example using user-supplied settings

All the principles outlined in this demonstration apply to the other routines: attgoal, minimax, leastsq, and fsolve.

---

**a: Unconstrained Problem:**

Consider initially the problem of finding a minimum to the function:

\[ f(x) = \exp(x(1)) \cdot (4x(1)^2 + 2x(2)^2 + 4x(1) \cdot x(2) + 2x(2) + 1) \]

The function to be minimized is:

\[ \text{fun} = '\exp(x(1)) \cdot (4x(1)^2 + 2x(2)^2 + 4x(1) \cdot x(2) + 2x(2) + 1)' \; ; \]

Take a guess at the solution

\[ x0 = [-1 1] ; \]

Use default parameters

\[ \text{options} = [] ; \]

\[ [x, \text{options}] = \text{fminu} (\text{fun}, x0, \text{options}) ; \]

The optimizer has found a solution at: \( x \)

The function value at the solution is:

Options (8)
The total number of function evaluations was:
Options (10)

Note: by entering the expression explicitly in a string using the variable 'x' we avoid having to write an M-file.

####################################################################

b: Constrained Problem:

Consider the above problem with two additional constraints:

\[
\begin{align*}
\text{minimize} & \quad f(x) = \exp(x(1)) \cdot (4x(1) + 2x(2)^2 + 4x(1)^2 \cdot x(2) + 2x(2) + 1) \\
\text{subject to} & \quad 1.5 + x(1) \cdot x(2) - x(1) - x(2) \leq 0 \\
& \quad -x(1) \cdot x(2) \leq 10
\end{align*}
\]

Objective Function and constraints:

funf = 'f = exp(x(1))*(4*x(1)^2+2*x(2)^2+4*x(1)*x(2)+2*x(2)+1);';
fung = 'g = [1.5 + x(1)*x(2) - x(1) - x(2); -x(1)*x(2) - 10];';
fun = [funf fung];

Take a guess at the solution

x0 = [-1 1];

Use default parameters

options = [];

[x, options] = constr(fun, x0, options);

A solution to this problem has been found at: x

The function value at the solution is:
Options (8)

Both the constraints are active at the solution:

\[g = [1.5 + x(1)*x(2) - x(1) - x(2); -x(1)*x(2) - 10] \]

The total number of function evaluations was:
Options (10)
c: Bounded Example:

Consider the above problem with additional bound constraints:

minimize \( f(x) = \exp(x(1)) \cdot (4x(1)^2 + 2x(2)^2 + 4x(1)x(2) + 2x(2) + 1) \)

subject to
\[
\begin{align*}
1.5 + x(1) \cdot x(2) - x(1) - x(2) & \leq 0 \\
-x(1) \cdot x(2) & \leq 10 \\
\text{and} & \\
x(1) & \geq 0 \\
x(2) & \geq 0
\end{align*}
\]

Objective Function and constraints

\[
\begin{align*}
\text{funf} &= \ 'f' = \exp(x(1)) \cdot (4x(1)^2 + 2x(2)^2 + 4x(1)x(2)+ 2x(2) + 1);' \\
\text{fung} &= \ 'g' = \ [1.5 + x(1) \cdot x(2) - x(1) - x(2); -x(1) \cdot x(2) - 10];' \\
\text{fun} &= \ [\text{funf} \text{ fung}];
\end{align*}
\]

Again, make a guess at the solution

\( x_0 = [-1 1]; \)

Use default parameters

\( \text{options} = []; \)

This time we will use the bounded syntax:
\( X = \text{CONSTR('FUN', X0, OPTIONS, VLB, VUB)} \)

\( \text{vlb} = \text{zeros}(1,2); \) Lower bounds \( X \geq 0 \)
\( \text{vub} = []; \) No upper bounds

\( [x, \text{options}] = \text{constr(fun, x0, options, vlb, vub}); \)

The solution to this problem has been found at: \( x \)

The function value at the solution is:
Options (8)

The constraint values at the solution are:

\( g = \ [1.5 + x(1) \cdot x(2) - x(1) - x(2); -x(1) \cdot x(2) - 10] \)

The total number of function evaluations was:
Options (10)
**d: User-Supplied Gradients:**

The above problem can be solved more efficiently and accurately if gradients are supplied by the user. This demo shows how this may be performed.

Again the problem is:

minimize \( f(x) = \exp(x(1)) \ast (4x(1)^2 + 2x(2)^2 + 4x(1)x(2) + 2x(2) + 1) \)

subject to \( 1.5 + x(1).x(2) - x(1) - x(2) \leq 0 \)
\[-x(1).x(2) \leq 10\]

----------------[

**Objective function and constraints:**

funf = 'f = \exp(x(1)\ast(4\ast x(1)^2 + 2\ast x(2)^2 + 4\ast x(1)\ast x(2) + 2\ast x(2) + 1));';
fung = 'g = [1.5 + x(1)\ast x(2) - x(1) - x(2); -x(1)\ast x(2) - 10];';
fun = [funf fung];

Make a guess at the solution:

x0 = [-1 1];

Use default parameters

options = [];

Partial derivatives

dfdx1 = 'df(x(1))\ast(4\ast x(1)^2 + 2\ast x(2)^2 + 4\ast x(1)\ast x(2) + 2\ast x(2) + 1) + 4\ast \exp(x(1))\ast(2\ast x(1)\ast x(2));';
dfdx2 = '4\ast \exp(x(1))\ast(x(1)\ast x(2) + 0.5);';
dfdg = '[x(2)-1, -x(2); x(1)-1, -x(1)];';
grad = ['df=\'dfdx1, dfdx2, \']\; dg=\'dfdg;]

[x, options] = constr(fun, x0, options, [], [], grad);

As before the solution to this problem has been found at: x

The function value at the solution is:

options(8)

Both the constraints are active at the solution:

g = [1.5 + x(1)\ast x(2) - x(1) - x(2); -x(1)\ast x(2) - 10]
The total number of function evaluations was:

\[ \text{options}(10) \]

e: *Equality Constrained Example:*

Consider the above problem with an additional equality constraint:

\[
\text{minimize } f(x) = \exp(x(1))\left(4x(1)^2 + 2x(2)^2 + 4x(1)x(2) + 2x(2) + 1\right)
\]

subject to,

\[
\begin{align*}
x(1) + x(2) &= 0 \\
1.5 + x(1)x(2) - x(1) - x(2) &\leq 0 \\
-x(1)x(2) &\leq 10
\end{align*}
\]

Objective function and constraints

\[
\text{funf} = 'f' = \exp(x(1))\left(4^{*}x(1)^{2}+2^{*}x(2)^{2}+4^{*}x(1)^{1}\text{*}x(2)+2^{*}x(2)+1\right)';
\]

\[
\text{fung} = 'g' = [x(1)+x(2);1.5+x(1)^{1}\text{*}x(2)-x(1)-x(2);-x(1)^{1}\text{*}x(2)-10]';
\]

\[
\text{fun} = [\text{funf} \, \text{fung}];
\]

Again, make a guess at the solution

\[ \text{x0} = [-1 \, 1]; \]

This time we will indicate the number of equality constraints by setting \text{options}(13) to the number of constraints. The equality constraints must be contained in first few elements of \text{g}.

*One equality constraint:*

\[
\text{clear options} \\
\text{options}(13) = 1;
\]

\[ \text{[x, options]} = \text{constr(fun, x0, options);} \]

The solution to this problem has been found at: \text{x}

The function value at the solution is:

Options (8)

The constraint values at the solution are:

\[ \text{g} = [x(1) + x(2); 1.5 + x(1)x(2) - x(1) - x(2); -x(1)x(2) - 10] \]

The total number of function evaluations was:

Options (10)
f: Changing The Default Settings

Consider the original unconstrained problem:

minimize \( f(x) = \exp(x(1)) \cdot (4x(1)^2 + 2x(2)^2 + 4x(1)x(2) + 2x(2) + 1) \)

This time we will solve it more accurately by overriding the default termination criteria (OPTIONS (2) and OPTIONS (3)).

The function to be minimized is

\[
\text{fun} = '\exp(x(1)) \cdot (4 \cdot x(1)^2 + 2 \cdot x(2)^2 + 4 \cdot x(1) \cdot x(2) + 2 \cdot x(2) + 1)' \;
\]

Again, make a guess at the solution:

\( x_0 = [-1 \ 1]; \)

Override the default termination criteria:

\[
\text{clear options} \;
\text{options (2) = le-8; Termination tolerance on X} \;
\text{options (3) = le-8; Termination tolerance on F} 
\]

\[ [x, \text{options}] = \text{fminu}(	ext{fun}, x_0, \text{options}); \]

The optimizer has found a solution at: \( x \)

The function value at the solution is:

Options (8)

The total number of function evaluations was:

Options (10)

Note: warning messages may be displayed because of problems in the calculation of finite difference gradients at the solution point. They are an indication that tolerances are overly stringent, however, the optimizer always continues to make steps towards the solution.

If we want a tabular display of each iteration we can set options (1) = 1 as follows:

\[
\text{options = 1; Set display parameter to on, the others to default.} 
\]

\[ [x, \text{options}] = \text{fminu}(	ext{fun}, x_0, \text{options}); \]

The optimizer has found a solution at: \( x \)
The function value at the solution is:
Options (8)

The total number of function evaluations was:
Options (10)

At each major iteration the table displayed consists of:

- number of function values
- function value
- step length used in the line search
- gradient in the direction of search

-----------------------------------End-----------------------------------
### Appendix- C

#### Some of the Commands used in the Matship files

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>global X Y Z</code></td>
<td>defines X, Y, and Z as global in scope.</td>
</tr>
<tr>
<td><code>fid = fopen ('file name', w)</code></td>
<td>Opens a file or obtains information about open file; w command deletes the contents of an existing file or create a new file, and open it for reading and writing.</td>
</tr>
</tbody>
</table>
| `count = fprintf (fid, exitflag, A,B,C,...)` | fid is an integer file identifier obtained from fopen. A value `exitflag` describes the exit condition:  
  - > 0 indicates that the function converged to a solution x.  
  - 0 indicates that the iteration count was exceeded. Increasing the tolerance (TolX parameter in options) may lead to a solution.  
  - < 0 indicates that the function did not converge to a solution. A,B,C,... are matrix elements to be printed. |
| `m = size(X,dim)` | returns the size of the dimension of X specified by scalar dim. |
| `oldopts = optimset('optimfun')` | creates an options structure options with all parameter names and default values relevant to the optimization function optimfun (fmincon in our study). |
| `options = optimset(oldopts,'param1',value1,..)` | MaxFunEvals [ positive integer ]: Maximum number of function evaluations allowed. 
MaxIter [ positive integer ]: Maximum number of iterations allowed. 
TolFun [ positive scalar ]: Termination tolerance on the function value. 
TolX [ positive scalar ]: Termination tolerance on x. |
<p>| <code>vlb [a b c .. n]</code> | defines lower bound for a solution |
| <code>vub [a b c .. n]</code> | defines upper bound for a solution |</p>
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>x, fval, attainfactor, exitflag</code></td>
<td><code>fgoalattain(...)</code> solves the goal attainment problem, which is one formulation for minimizing a multi-objective optimization problem. Returns a value <code>exitflag</code> that describes the exit condition of <code>fgoalattain</code>.</td>
</tr>
<tr>
<td><code>fun</code></td>
<td>The function to be minimized. <code>fun</code> takes a vector <code>x</code> and returns a vector <code>F</code> of the objective functions evaluated at <code>x</code>. Alternatively, <code>fun</code> can be a string containing the name of a function (an M-file, a built-in function, or a MEX-file). If <code>fun=myfun'</code> then the M-file function <code>myfun.m</code> would have the form <code>function F = myfun(x)</code> <code>F = ... % Compute function values at x</code></td>
</tr>
<tr>
<td><code>nonlcon</code></td>
<td>The function that computes the nonlinear inequality constraints <code>c(x) &lt;= 0</code> and nonlinear equality constraints <code>ceq(x)=0</code>. <code>nonlcon</code> is a string containing the name of a function (an M-file, a built-in, or a MEX-file). <code>nonlcon</code> takes a vector <code>x</code> and returns two arguments, a vector <code>c</code> of the nonlinear inequalities evaluated at <code>x</code> and a vector <code>ceq</code> of the nonlinear equalities evaluated at <code>x</code>. For example, if <code>nonlcon = 'mycon'</code> then the M-file <code>mycon.m</code> would have the form <code>function [c, ceq] = mycon(x)</code> <code>c = ... % Compute nonlinear inequalities at x</code> <code>ceq = ... % Compute the nonlinear equalities at x</code></td>
</tr>
<tr>
<td><code>goal</code></td>
<td>Vector of values that the objectives attempt to attain. The vector is the same length as the number of objectives <code>F</code> returned by <code>fun</code>. <code>fgoalattain</code> attempts to minimize the values in the vector <code>F</code> to attain the <code>goal</code> values given by <code>goal</code>.</td>
</tr>
</tbody>
</table>
| `weight` | A weighting vector to control the relative under-attainment or over-attainment of the objectives in `fgoalattain`. When the values of `goal` are all nonzero, to ensure the same percentage of under- or over-attainment of the active objectives, set the weighting function to `abs(goal)`. (The active objectives are the set of..."
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disp(X)</td>
<td>( disp(X) ) displays an array, without printing the array name. If ( X ) contains a text string, the string is displayed.</td>
</tr>
<tr>
<td>fclose(fid)</td>
<td>closes the specified file, if it is open.</td>
</tr>
<tr>
<td>roots(p)</td>
<td>gives roots of a polynomial function</td>
</tr>
<tr>
<td>k = isreal(A)</td>
<td>returns logical true (1) if all elements of ( A ) are real numbers, and logical false (0) if either ( A ) is not a numeric array, or if any element of ( A ) has a nonzero imaginary component. Since strings are a subclass of numeric arrays, ( isreal ) always returns 1 for a string input.</td>
</tr>
<tr>
<td>sminteg</td>
<td>a file created for integration by Simpson's method.</td>
</tr>
<tr>
<td>fun(funname, x1, x2, varargin)</td>
<td>varargin pass variable numbers of arguments; accepts a variable number of arguments into function ( fun.m ).</td>
</tr>
<tr>
<td>[y1,y2..] = feval(function,x1,..,xn)</td>
<td>If ( function ) is a string containing the name of a function (usually defined by an M-file), then ( feval(function,x1,..,xn) ) evaluates that function at the given arguments.</td>
</tr>
<tr>
<td>yi = interp1(x,Y,xi,method)</td>
<td>interpolates using alternative methods:</td>
</tr>
<tr>
<td></td>
<td>• 'nearest' for nearest neighbor interpolation</td>
</tr>
<tr>
<td></td>
<td>• 'linear' for linear interpolation</td>
</tr>
<tr>
<td></td>
<td>• 'spline' for cubic spline interpolation</td>
</tr>
<tr>
<td></td>
<td>• 'cubic' for cubic interpolation</td>
</tr>
<tr>
<td>t = strcat(s1,s2,s3,..)</td>
<td>horizontally concatenates corresponding rows of the character arrays ( s1, s2, s3, ) etc.</td>
</tr>
<tr>
<td>[A,count] fscanf(fid,format,size)</td>
<td>reads the amount of data specified by size, converts it according to the specified format string, and returns it along with a count of elements successfully read. ( size ) is an argument that determines how much data is read.</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>legend('string1','string2',...)</td>
<td>displays a legend in the current axes using the specified strings to label each set of data.</td>
</tr>
<tr>
<td>xlabel('string')</td>
<td>labels the x-axis of the current axes.</td>
</tr>
<tr>
<td>ylabel('string')</td>
<td>labels the y-axis of the current axes.</td>
</tr>
<tr>
<td>title('string')</td>
<td>outputs the string at the top and in the center of the current axes.</td>
</tr>
<tr>
<td>h = gca</td>
<td>returns the handle to the current axes for the current figure. If no axes exist, MATLAB creates one and returns its handle. The current axis is the target for graphics output when one creates axes children.</td>
</tr>
</tbody>
</table>
Appendix- D

Some of the files used in Matship

File 1: designopt.m

% Name: designopt.m
% Description: The file designopt1.m is the main file in the design process, which is used for optimization purpose to get the values of different design variables based on the objective function ('myfun') and the constraint file ('mycon') containing the set of constraint functions.

format short g

% Global parameters and variables:

global density_m density_ft g_ms g_ft
global material geartype Cm Cw coeff_freeboard ws ehp
global h13_5

% density of sea water [LT/m^3]
  density_m = 1.025/1.016;
% density of sea water [LT/ft^3]
  density_ft = (1.025/1.016)*(0.3048)^3;

% gravitational acc. ft/s
  g_ft = 32.2;
% gravitational acc. m/s
  g_ms = 9.807;

% Cm = mid ship sectional area coefficient
  Cm = 0.878;
% Waterplane area coefficient
  Cw = 0.95;

% Sea state information...
  h13_ref = [0.30 0.88 1.40 2.45 3.65];
  t_ref = [2.4 3.9 4.6 5.7 6.8];
  tmax_ref = [3.4 5.4 6.5 8.1 9.7];
  lambda_ref = [6.1 15.9 21.6 34.0 49.0];

% h13_5: significant wave height for sea state 5
  h13_5 = 3.65;  % [m]

% heading : ship's heading [deg]
  heading_angle = 180;  % sea keeping design algorithms are only valid for head seas

% Input parameters ...

- 92 -
sea_states = [5]; % sea states
%sea_states = [3,5];
hold_capacity = [10 25 50 75 100 125 150 175 200 250 300]; % hold capacity range [LT]
%hold_capacity = [200];
design_speeds = [10]; % design speeds [kn]
material = 'aluminum'; % or 'steel'
range_nm = 500; % range [nautical miles]
coeff_freeboard = 0.115; % ratio of freeboard/draft

filename = 'c:\users\ayhan\mizan\ship\runrmsmo.out';
fid = fopen(filename,'w');
count = fprintf(fid,' exitflag designseastate hold v_kn Lwl B T Cb
    displacement totalweight requiredpower stab_loaded stab_light cost rms_hv rms_pt peak_hv
    peak_pt tmO 1 tm02 tm03 rmsHVacc rmsPTacc amphvrms ampptrms
    
    lb = 3.06; % L/B ratio for UBC Series main hull
    bt = 2.49; % B/T ratio for UBC Series main hull

% Initial points ...
x00(l,:) = [39 18 6.75 0.615 ];
x00(2,:) = [45 19.3 7.1 0.615 ];
x00(3,:) = [52 20.5 7.5 0.615 ];
x00(4,:) = [57 21 6.4 0.615 ];
x00(5,:) = [62 22.3 6.8 0.615 ];
x00(6,:) = [72 25 9 0.615 ];
x00(7,:) = [72 24 8.5 0.615 ];
x00(8,:) = [74 24 8.7 0.615 ];
x00(9,:) = [75 24 9 0.615 ];
x00(10,:) = [82 25.5 11 0.615 ];
x00(11,:) = [85 25.5 10 0.615 ];
for i_ss = 1:size(sea_states,2)
designss = sea_states(i_ss);
    h13_ms = h13_ref(designss); % significant wave height for design sea state
    lambda_ss = lambda_ref(designss); % average wave length for the design sea state
    wnmmax = 2*pi/tmax_ref(designss); % wave frequency that maximizes sea state's energy spectrum
for i_hold = 1:size(hold_capacity,2)
    i_hold = size(hold_capacity,2) + 1 - i_hold;
for i_v = 1:size(design_speeds,2)
    v_kn = design_speeds(i_v);
    heading = (pi/180)*heading_angle;
    v_ms = 0.5144*v_kn;
    wave_amp = 1.0;
    % waterline length estimation based on the hold capacity
    L_0 = 3.3*(28.5*hold)^0.3669;
    % This restriction is applicable for westcoast
    L_licence = (v_kn^2)/1.4;
    % ship vector x consist of [ Lwl B T Cb] and is in feets
    % for optimization models
    L0 = L_0;
    B0 = L0 / (0.017742*L_0+1.690323);
    if strcmp(material,'aluminum') == 1
        T0 = B0 / ((1+coeff_freeboard)*2.4); % for aluminum

elseif strcmp(material,'steel') == 1
    TO = B0 / ((1+coeff_freeboard)*2.2); % for steel
else
    disp('Error! Unknown material type!!!');
    TO = 0.0; % unknown material! Not implemented!
end
Cb0 = 0.5; % 0.615;

%x0 = [L0 B0 T0 Cb0] % starting point
x0 = x00(i_hold,:)

% Parameter ranges ...

%disp(x0) %density_ft*prod(x0);
%

%total_weight = tweight(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
%power = pwerfish(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
%stabLoaded = stab(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
%stabLite = stablite(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
%rms_hv(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%rms_pch(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%peak_hv(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%peak_pch(x0, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%tm01(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%tm02(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%tm03(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)
%acchvrms(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
%accptrms(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
%amphvrms(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
%amphvrm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

 disp('optimization started...
oldopts = optimset('fmincon');
options = optimset(oldopts,'LineSearchType','quadpoly', 'MaxFunEvals',1000000,'MaxIter',200,...
    'TolCon',1.0e-002, 'TolFun',1.0e-001,'TolX',1.0e-2);

% bounds on the variables
%vlb = [10 1 0.25 0.5]; % lower bounds for x vector
%vub = [L0*1.3 x(1)/2.6 x(1)/2.6/2 0.65];
vlb = [35.39 1 0.1 0.5];
vub = [200 75 40 0.7]; % upper bounds for x vector
%vub = [120 80 40 0.7]; % upper bounds for x vector

%a(1:4) = [1 0 0 0];
%b(1:4) = [L_licence 0 0 0];

fun = 'cost'; % objective function definition
function "mycon.m"

% inequality and equality constraints defined in the
% nonlcon = 'mycon';
[x fval exitflag] = fmincon(fun, x0, [], [], [], vlb, vub,...
    nonlcon, options, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss,
    wnmax)

res(i_ss, i_hold, i_v, 1:4) = x;

[hold v_kn x]

res(i_ss, i_hold, i_v, 5) = density_ft*prod(x);
res(i_ss, i_hold, i_v, 6) = tweight(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v, 7) = pwerfish(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v, 8) = stab(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v, 9) = stablire(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,10) = cost(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,11) = rms_hv(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,12) = rms_pch(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,13) = peak_hv(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,14) = peak_pch(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,15) = tm01(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,16) = tm02(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,17) = tm03(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,18) = acpvrms(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
res(i_ss, i_hold, i_v,19) = acptrms(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);

count = fprintf(fid,' %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g
%g%g
', ...
    exitflag, designss, hold, v_kn, x,...
    res(i_ss, i_hold, i_v, 5), res(i_ss, i_hold, i_v, 6),res(i_ss, i_hold, i_v, 7),res(i_ss, i_hold, i_v, 8),...
    res(i_ss, i_hold, i_v, 9), res(i_ss, i_hold, i_v,10),res(i_ss, i_hold, i_v,11),res(i_ss, i_hold, i_v,12),...
    res(i_ss, i_hold, i_v,13), res(i_ss, i_hold, i_v,14),res(i_ss, i_hold, i_v,15),res(i_ss, i_hold, i_v,16),...
    res(i_ss, i_hold, i_v,17), res(i_ss, i_hold, i_v,18),res(i_ss, i_hold, i_v,19),res(i_ss, i_hold, i_v,20),...
    res(i_ss, i_hold, i_v,21));

disp([' i_ss, i_hold, i_v ']);

[i_ss, i_hold, i_v ]
end, end, end
fclose(fid);
File 2: we2w.m

% Name: we2w.m
% Description: This file (we2w.m) returns w, wave frequency for a given encountered
frequency, we.

function we_to_w = we2w(heading, v_ms, we)

% heading : ship's heading with respect to the waves [rad]
% v_ms : ship speed [m/s]
% we : encounter wave frequency [rad/s]
% w : wave frequency [rad/s]
% g_ms : gravity [m/s^2]

global g_ms

if (v_ms == 0);
    w = we;
else
    a = v_ms * cos(heading) / g_ms;
p = [ 1 -1/a we/a];
r = roots(p);

    if (isreal(r(1)) & r(1) >= 0);
        w = r(1);
    elseif (isreal(r(2)) & r(2) >= 0);
        w = r(2);
    else
        display('No real roots or no positive roots!!');
        w = NaN;
    end
end

we_to_w = w;
File 3: sspec.m

% Name: sspec.m
% Description: This file (sspec.m) reproduces the standard ITTC sea-spectrum for given seastate. If an algorithm for a particular seastate of a specific route is available, it will be better to use the same for more reliable results.

% ITTC standard Wave Spectrum

function sea_spectrum = sspec(h13_ms, w)

global g_ms

% h13_ms : significant wave height [m]
% w : wave frequency [rad/s]
% g_ms : gravity [m/s^2]

A = 8.10e-3 * g_ms^2;
B = 3.11 / h13_ms^2;

sea_spectrum = (A/w^5) * exp(-B/w^4);

File 4: myfun.m

% Name: myfun.m
% Description: This file is used to define the objective function. F(1) and F(2) are the sub-functions of this objective functions, which are to be minimized by the optimization procedure. The function myfun, F(1) and F(2) are all functions of the variables listed within the bracket.

function F = myfun(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)

% For single goal optimization...

% Minimize cost

F = cost(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax)/1000000;

% For multi-goal optimization...

% Minimize Peak Heave and Peak Pitch ratios and Cost

%F(1) = 1.0/peak_hv(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
% F(2) = 1.0/peak_pch(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, hl3_ms, lambda_ss, wnmax);
% F(3) = cost(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, hl3_ms, lambda_ss, wnmax)/1000000;

File 5: mycon.m

% Name: mycon.m
% Description: The file mycon.m contains a list of constraints that the Matlab Optimization Toolbox uses to produce the results. There can be a group of equality and inequality constraints for the optimization procedure. The constraints listed as functions c(1), c(2),…c(9) are different files created separately to define the constraints. These constraints, usually define the lower and upper bounds, the range within which a result is targeted. Constraint c(7) is a constraint based on licensed length.

function [c, ceq] = mycon(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, hl3_ms, lambda_ss, wnmax)
% inequality constraints:
c(1) = cons_lbm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(2) = cons_bt2m(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(3) = cons_btlm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(4) = cons_bb2m(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(5) = cons_ban(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(6) = x(1) - 0.01;
c(7) = stab(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(8) = rms_hv(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax) - 0.7;
c(9) = acchvrm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax) - 0.3;
c(10) = ampptrm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax) - 5.0;
c(11) = rmsveraccaft(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax) - 0.18*9.81;
c(12) = 3.0 - peak_hv(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(13) = 3.0 - peak_pch(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
c(14) = tm01(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax) - 0.22;
% equality constraints:

ceq = [];
% ceq(1) = cons_lbm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
% ceq(2) = cons_btlm(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
% ceq = cons_ban(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda_ss, wnmax);
File 6: cons_btl.m

% Name: cons_btl.m
% Description: The file cons_btl.m defines the upper bound of the beam-draft ratio as a constraint in optimization.

function constraint_value = cons_btl(x, speedkn, rangenm, hold, heading, v_ms, wave_amp, hl3_ms, lambda, wnmax)
    constraint_value = x(2)/x(3) - 4.0;

File 7: cons_bt2.m

% Name: cons_bt2.m
% Description: The file cons_bt2.m defines the lower bound of the beam-draft ratio as a constraint in optimization.

function constraint_value = cons_bt2(x, speed_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
    constraint_value = 1.99 - x(2)/x(3);

File 8: cons_lb1.m

% Name: cons_lb1.m
% Description: The file cons_lb1.m defines the upper bound of the length-beam ratio as a constraint in optimization.

function constraint_value = cons_lb1(x, speed_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
    constraint_value = x(1)/x(2) - 4.5;

File 9: cons_lb2.m

% Name: cons_lb2.m
% Description: The file cons_lb2.m defines the lower bound of the length-beam ratio as a constraint in optimization.

function constraint_value = cons_lb2(x, speed_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
    constraint_value = 1.99 - x(1)/x(2);
**File 10: cons_disp.m**

% Name: cons_disp.m
% Description: The file cons_disp.m defines the condition of equality, of the displacement and total weight of the vessel as an *equality constraint* in optimization.

```matlab
function constraint_value = cons_disp(x, speed_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

global density_ft

constraint_value = tweight(x, speed_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax) - density_ft*prod(x);
```

**File 11: cost.m**

% Name: cost.m
% Description: This file (cost.m) computes the total cost of a vessel based on the algorithm produced by Akinturk for the UBC series fishing vessels and may be applied for both steel and aluminum vessels.

```matlab
function total_cost = cost(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

global ws ehp material

tweight(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax);
pwerfish(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax);

length = x(1);

v_ms = 0.5144*v_kn;

if strcmp(material,'aluminum') == 1
    coeff = 11064; % for aluminum
elseif strcmp(material,'steel') == 1
    coeff = 4155; % for steel
else
    disp('Error! Unknown material!');
    coeff = -100000.0; % unknown material! Not implemented!
end

hullweight = ws;
hull_cost = coeff*hullweight;%11064(Akinturk's Thesis); % for Alumium; 4155 for steel.
outfit_cost = 120.0*length^2;

shp = ehp/(0.97*0.58);
gal_hr = shp*(0.4/8.1);
fuel_cost = (7500/v_kn)*gal_hr*2.5;
```
mach_cost = 250*shp;
gear_cost = 200000;
capital_cost = (hull_cost + outfit_cost + mach_cost + gear_cost)*1.3221;
derpreciation_cost = capital_cost*0.045;
if capital_cost < 500000
   insurance_cost = capital_cost*0.02;
elseif capital_cost >=500000 & capital_cost <=2000000
   insurance_cost = capital_cost*0.015;
else insurance_cost = capital_cost*0.01;% this case for capital_cost> 2x10^6;
end
end_total_cost= capital_cost + depreciation_cost + insurance_cost;

**File 12: displace.m**

% **Name:** displace.m
% **Description:** This file (displace.m) computes the total displacement of a vessel.

function displacement = displace(x)

global density_ft

% length = x(1); beam = x(2); draft = x(3); Cb = x(4);

displacement = density_ft*x(1)*x(2)*x(3)*x(4); %displacement in long tons
% File 13: tweight.m

% Name: tweight.m
% Description: This file (tweight.m) computes the total weight of a vessel based on the algorithm produced by Akinturk for the UBC series fishing vessels and may be applied for both steel and aluminum vessels and different types of fishing gears e.g. seiners, trawlers etc.

% This file (tweight.m) computes the total weight of a vessel

function total_weight = tweight(x, speed_kn, range_nm, hold_cap, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)
    length = x(1);
    beam = x(2);
    draft = x(3);
    cb = x(4);

    pm = pwerfish(x, speed_kn);
    sfcm = 250 - 50.0*pm/l810;  % [gallons/(kw hr)]
    sfcm = 4.5609 * sfcm;  % [LT/(kw hr)]; 1 gal U.K.=4.5609 litres

    % assume "depth = draft + freeboard"
    freeboard = coeff_freeboard * draft;
    depth = draft + freeboard;  % depth, draft and freeboard in feet.

    % WEIGHT GROUPS:
    cn = length*beam*depth/100;  %cn is called 'cubic number'.

    % Structural weight, ws [LT]: This includes the weight of hull, deck and deckhouse in long tons for both steel and aluminium construction.
    if strcmp(material, 'aluminum') == 1
        k = 0.55;  % for aluminum
    elseif strcmp(material, 'steel') == 1
        k = 1.0;  % for steel
    else
        k = 0.0;  % unknown material! Not implemented!
    end

    c=100.0*(length*(beam+draft)/3000.0)*(4.0/3.0);
ws = k*0.00002813*(c^3-173.52*c^2+41880*c-0.0000158);

% Machinery weight, wm [LT]:
wm = cn*(cn/27300.0 + 0.0537);

% Outfit weight:
wo = 0.25*x(1)*x(2);

% Stores weight (water + fuel + provision):
if x(1) < 39.03/3.618
   wp = 0;
else
   wp = 3.618*x(1) - 39.03; % From Ayhan's thesis.
end

% Gear weight, wg: For seiner vessel the gear weight includes the net
% and the skiff.
% wg = 7.23; % for seiners
% wg = 3.05 for trawlers, which includes the net and wire rope.
if strcmp(geartype,'seiner') == 1
   wg = 7.23;
elseif strcmp(geartype,'trawler') == 1
   wg = 3.05;
else
   wg = 0.0; % for Transglobal ferry.
end

% payload is hold_cap capacity and any other weight might be included in payload = hold_cap;

% Total weight, wt [LT]
wt = ws + wp + hold_cap + wg + wm + wo;

%*************************************************************************
% objective function:
%*************************************************************************
total_weight = wt;
File 14: hv_amp.m

% Name: hv_amp.m
% Description: This file (hv_amp.m) computes heave amplitude (non-dimensional) for a
given encounter frequency based on D. Howard's algorithm for UBC series fishing vessels.

function nond_hv = hv_amp(Lwl,B, T, Cb,v_ms, wave_amp, we)

% g_ms : gravity in [m/s^2]
% we : encounter frequency [rad/s]
% wen : nondimensional encounter frequency
% wave_amp : wave amplitude = 1 [m] or [ft]
% Lwl : waterline length [ft]
% Lbp : length between perpendiculars [ft]
% B : beam [ft]
% T : draft [ft]
% Cb : block coefficient
% v_ms : ship speed [m/s]

global g_ms

Lbp = Lwl*.94;
Lbp_m = 0.3048* Lbp; % Lbp in [m]

% Froude's number
Fn = v_ms / sqrt(Lbp_m*g_ms);

wen = we * sqrt(Lbp_m/g_ms);

A_hv = 0.8591*Fn*Lbp*T/B^2 + 1.5804*Fn*T/(Cb*B);
alpha_hv = 1.816 + 31.42*Cb*T/Lbp;
delta_hv = 16.48*T/Lbp - .7485*T/(Cb*B);
beta_hv = 6.436*Cb*Lbp*T/B^2 + 163.1*Cb*Fn*T/B;
gamma_hv = 2.870*Cb*Lbp*T/B^2 + 40.36*Cb*Fn*T/B;

% non dimensional heave amplitude by Howard's algorithm
nond_hv = 1 + A_hv / exp((alpha_hv-delta_hv*wen)^2) ... - 1 / ( 1 + exp (beta_hv-gamma_hv*wen)) ;

% dimensional heave amplitude
%d_hv = nond_hv * wave_amp;
File 15: pt_amp.m

% Name: pt_amp.m
% Description: This file (pt_amp.m) computes pitch amplitude (non-dimensional) for a given
encounter frequency based on D. Howard's algorithm for UBC series fishing vessels.

function nond_pt = pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, we)

% g_ms : gravity in [m/s^2]
% we : encounter frequency [rad/s]
% wen : non-dimensional encounter frequency
% wave_amp : wave amplitude = 1 [m] or [ft]
% Lwl : waterline length [ft]
% Lbp : length between perpendiculars [ft]
% B : beam [ft]
% T : draft [ft]
% Cb : block coefficient
% v_ms : ship speed [m/s]

global g_ms

Lbp = Lwl*0.94;
Lbp_m = 0.3048* Lbp; % Lbp in [m]

% Froude's number
Fn = v_ms / sqrt(Lbp_m*g_ms);

wen = we * sqrt(Lbp_m/g_ms);

A_pch = 1.1313*Fn*Lbp*T/B^2 + 0.9264*Fn*T/(Cb*B);
alpha_pch = 0.7904 + 45.47*Cb*T/Lbp;
beta_pch = 18.34*Cb*Lbp*T/B^2 + 37.35*Cb*Fn*T/B;
gamma_pch = 7.211*Cb*Lbp*T/B^2 + 3.039*Cb*Fn*T/B;

% non dimensional pitch amplitude by Howard's algorithm
nond_pt = 1 + A_pch / exp( (alpha_pch - delta_pch*wen)^2 ) - 1 / (1 + exp(beta_pch-gamma_pch*wen) );

%d_pt = nond_pt * (we^2*wave_amp/g_ms);

- 105 -
% Name: peak_hv.m
% Description: This file (peak_hv.m) computes peak heave ratio, dimensional and non-dimensional peak heave values for UBC series fishing vessels. This is ratio between the frequency at which the wave spectrum reaches its peak, to that of the same of the ship’s response spectrum for its heave motion.

function ratio_hv = peak_hv(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

global g_ms

L=x(1)*0.3048;
B=x(2)*0.3048;
T=x(3)*0.3048;
Cb=x(4);

lb = L/B;
b = B/T;
fk = v_kn*0.5144 / sqrt(g_ms*L);

f1 = (38.8668 - 11.5286*Cb - 5.2002*lb - 1.1912*bt)^2;

f2 = (0.8652 - 0.2816*fk)^2;

f3 = (49.6354 - 20.4321*Cb - 6.8076*lb + 0.6441*bt)^2;

wn_hv = sqrt(f1 / (f2*f3)); % Non dimensional

w_hv = sqrt(g_ms/T) * wn_hv; % Dimensional [rad/s]

for head seas : cos(heading angle) = -1; The following is derived from we = w - w^2*VCosx/g, Cos180 being -1 making we = w + w^2 V/g; ratio = w_hv/we;

ratio_hv = w_hv / (wnmax+wnmax^2*(0.5144*v_kn)/g_ms);
File 17: peak_pch.m

% Name: peak_pch.m
% Description: This file(peak_pch.m) computes peak pitch ratio, dimensional and non-
dimensional peak pitch values.

function ratio_pch = peak_pch(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

    global g_ms

    L = x(1)*0.3048;
    B = x(2)*0.3048;
    T = x(3)*0.3048;
    Cb = x(4);

    lb = L/B;
    bt = B/T;
    fn = 0.5144*v_kn / sqrt(g_ms*L);

    %******************************************************************************
    % objective function:
    %
    wn_pch = sqrt(f1 / (f2*f3)); % Non dimensional .
    w_pch = sqrt(g_ms/T) * wn_pch; % Dimensional [rad/s]
    %
    % for head seas : cos(heading angle) = -1; The following is derived from we=w^-w'2*VCosx/g, Cos180 being -1
    % making we=w+w'2V/g; ratio=w_pch/we;
    %
    ratio_pch = w_pch / (wnmax+wnmax^2*(0.5144*v_kn)/(g_ms));

    %******************************************************************************

    f1 = (2.4541 - 0.7280*Cb - 0.3283*lb - 0.0752*bt)^2;
    f2 = (0.9481 - 0.3086*fn)^2;
    f3 = (2.8599 - 1.1773*Cb - 0.3992*lb + 0.0371*bt)^2;

    %******************************************************************************
File 18: rms_hv.m

% Name: rms_hv.m
% Description: This file (rms_hv.m) computes rms heave response for a given sea state for the UBC series fishing vessels.

function heave = heave (x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

    global h13_5 g_ms

    L=x(1)*0.3048;
    B=x(2)*0.3048;
    T=x(3)*0.3048;
    Cb=x(4);

    bl=B/L; tl=T/L; ll=lambda/L;

    Fn=0.5144*v_kn / sqrt(g_ms*L); %Fn is Froude Number
    h5=h13_5;

    nh=h13_ms/h5; %nh1/3=Non-dimensional significant wave height.

    f1=0.17244257-0.10902408*bl-0.062345772*bl^2;
    f2=0.92116235+9.26182543*tl-17.01219503*tl^2;
    f3=1.25445302+0.015068303*Cb;
    f4=0.37459542+0.33475563*Fn+0.69543449*Fn^2-0.62937284*Fn^3;
    f5=0.51156287+1.67664716*ll-0.60058403*ll^2+0.065781060*ll^3;
    f6=1.27331482-1.02535654*nh+0.56388486*nh^2;

    f=f1*f2*f3*f4*f5*f6;

    heave=h13_ms*(f+0.74616424*f^2-0.46955309*f^3); % rms heave in meters
File 19: rms_pch.m

% Name: rms_pch.m
% Description: This file (rms_pch.m) computes rms pitch response for a given sea state for the UBC series fishing vessels.

function pitch=pitch(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

    global h13_5 g_ms
    L=x(1)*0.3048;
    B=x(2)*0.3048;
    T=x(3)*0.3048;
    Cb=x(4);
    bl=B/L; tl=T/L; ll=lambda/L;
    disp_vol=L*B*T*Cb;
    dl=(disp_vol*(1/3))/L;
    Fn=0.5144*v_kn / sqrt(g_ms*L); %Fn is Froude Number
    h5=h13_5;
    nh=h13_ms/h5; %nh1/3=Non-dimensional significant wave height.
    f1=0.465143-2.343704*dl+3.482758*dl^2;
    f2=1.323205+13.947877*bl-26.389156*bl^2;
    f3=0.346817-24.620275*tl+37.557687*tl^2;
    f4=0.044729+1.003151*Cb;
    f5=-0.182212-0.296472*Fn+0.059477*Fn^2-0.245965*Fn^3;
    f6=-0.181446+4.924184*ll-0.834965*ll^2+0.074877*ll^3;
    f7=2.16213525-2.78020823*nh+1.54756589*nh^2;
    f=f1*f2*f3*f4*f5*f6*f7;
    pitch=h13_ms*((wnmax)^2)/9.807*(f-0.46697339*f^2+0.56576473*f^3)*180/pi;
File 20: s_acc_h1.m

% Name: s_acc_h1.m
% Description: This file (s_acc_h1.m) computes the compound horizontal acceleration at aft
deck given an encounter frequency based on the formulation given in Akinturk's thesis for
the UBC series fishing vessels.

function sjiorizontalaccaft = s_acc_h1 (x, heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb)

% B : beam of the ship [ft]
% Cb : block coefficient
% D : depth of the ship [ft]
% heading : ship's heading [rad] (head seas is 180 degrees)
% g_ms : gravity [m/s^2]
% Loa : length over all [ft]
% Lwl : waterline length [ft]
% T : draft of the ship [ft]
% v_ms : ship speed [m/s]
% w : wave frequency [rad/s]
% x : encounter frequency vector [rad/s]

global g_ms

Loa = 1.05*Lwl*0.3048;
D = 1.30*T;

for i = 1 : size(x,2);
    w = we2w (heading, v_ms, x(i));

    pitch_amp = (w^2/g_ms)*pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))...
        *sqrt(sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5));

    waveamplitude = sqrt(sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5));

    horizontal_acc_aft_term = hv_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))*sin(pitch_amp)...
        -(Loa/2)*(w^4*waveamplitude/g_ms^2)*(pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))^2);

    s_horizontal_acc_aft(i) = x(i)^5 * horizontal_acc_aft_term^2 ...
        * sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5);

end % end for "for i = ...

- 110 -
**File 21: s_accvl.m**

% **Name:** s_accvl.m  
% **Description:** This file (s_accvl.m) computes the compound vertical acceleration at aft deck given an encounter frequency based on the formulation given in Akinturk's thesis for the UBC series fishing vessels.

```matlab
function s_vertical_acc_aft = s_accvl(x, heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb)

% B : beam of the ship [ft]
% Cb : block coefficient
% D : depth of the ship [ft]
% heading : ship's heading [rad] (head seas is 180 degrees)
% g_ms : gravity [m/s^2]
% Loa : length over all [ft]
% Lwl : waterline length [ft]
% T : draft of the ship [ft]
% v_ms : ship speed [m/s]
% w : wave frequency [rad/s]
% wave_amp: wave amplitude = 1 ([m] or [ft])
% x : encounter frequency vector [rad/s]

global g_ms

Loa = 1.05*Lwl * 0.3048;
D = 1.30*T;

for i = 1 : size(x,2);
    w = we2w (heading, v_ms, x(i));
    pitch_amp = (w^2*wave_amp/g_ms)*pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))...
        *sqrt(sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms).^5);
    vertical_acc_aft_term = hv_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))*cos(pitch_amp)...
        - (Loa/2)*(w^2/g_ms)*pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i));
    s_vertical_acc_aft(i) = x(i)^5* vertical_acc_aft_term^2 ...
        *sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms).^5;
end % end for "for i = ..."
```

- 111 -
File 22: s_acc_h2.m

% Name: s_acc_h2.m
% Description: This file (s_acc_h2.m) computes the compound horizontal acceleration at midship deck given an encounter frequency based on the formulation given in Akinturk's thesis for the UBC series fishing vessels.

function s_horizontal_acc_mid = s_acc_h2(x, heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb)

% B : beam of the ship [ft]
% Cb : block coefficient
% D : depth of the ship [ft]
% heading: ship's heading [rad] (head seas is 180 degrees)
% g_ms: gravity [m/s^2]
% Loa: length over all [ft]
% Lwl: waterline length [ft]
% T: draft of the ship [ft]
% v_ms: ship speed [m/s]
% w: wave frequency [rad/s]
% wave_amp: wave amplitude = 1 ([m] or [ft])
% x: encounter frequency vector [rad/s]

global g_ms
Loa = 1.05*Lwl*0.3048;
D = 1.30*T;

for i = 1 : size(x,2);
    w = we2w(heading, v_ms, x(i));
    pitch_amp = (w^2/g_ms)*pt_amp(Lwl, B, Cb, v_ms, wave_amp, x(i))... *sqrt(sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^0.5);
    horizontal_acc_mid_term = hv_amp(Lwl, B, Cb, v_ms, wave_amp, x(i))*sin(pitch_amp);
    s_horizontal_acc_mid(i) = x(i)^5 * horizontal_acc_mid_term^2 ... *sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^0.5;
end % end for "for i = ..."
File 23: s_acc_v2.m

% Name: s_acc_v2.m
% Description: This file (s_accv2.m) computes the compound vertical acceleration at midship deck, given an encounter frequency based on the formulation given in Akinturk's thesis for the UBC series fishing vessels.

function s_vertical_acc_mid = s_accv2(x, heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb)

% B : beam of the ship [ft]
% Cb : block coefficient
% D : depth of the ship [ft]
% heading : ship's heading [rad] (head seas is 180 degrees)
% g_ms : gravity [m/s^2]
% Loa : length over all [ft]
% Lwl : waterline length [ft]
% T : draft of the ship [ft]
% v_ms : ship speed [m/s]
% w : wave frequency [rad/s]
% wave_amp: wave amplitude = 1 (m) or [ft]
% x : encounter frequency vector [rad/s]

global g_ms
Loa = 1.05*Lwl * 0.3048;
D = 1.30*T;
for i = 1 : size(x,2);
    w = we2w (heading, v_ms, x(i));
    pitch_amp = (w^2*wave_amp/g_ms)*pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))... *sqrt(sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5);
    vertical_acc_mid = hv_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))*cos(pitch_amp);
    s_vertical_acc_mid(i) = x(i)^5 * vertical_acc_mid^2 ... *sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5;
end % end for "for i = ..."
File 24: s_acc_h3.m

% Name: s_acc_h3.m
% Description: This file (s_acc_h3.m) computes the compound horizontal acceleration at fore deck given an encounter frequency based on the formulation given in Akinturk's thesis for the UBC series fishing vessels.

function s_horizontal_acc_bow = s_acc_h3(x, heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb)

% B : beam of the ship [ft]
% Cb : block coefficient
% D : depth of the ship [ft]
% heading : ship's heading [rad] (head seas is 180 degrees)
% g_ms : gravity [m/s^2]
% Loa : length over all [ft]
% Lwl : waterline length [ft]
% T : draft of the ship [ft]
% v_ms : ship speed [m/s]
% w : wave frequency [rad/s]
% wave_amp: wave amplitude = 1 ([m] or [ft])
% x : encounter frequency vector [rad/s]

global g_ms
Loa = 1.05*Lwl*0.3048;
D = 1.30*T;

for i = 1 : size(x,2);
    w = we2w(heading, v_ms, x(i));
    pitch_amp = (w^(2/g_ms)) * pt_amp(Lwl, B, Cb, v_ms, wave_amp, x(i)) * sqrt(sspec(h13_ms, w) / (1 - 4 * x(i) * v_ms * cos(heading) / g_ms)^0.5);
    waveamplitude = sqrt(sspec(h13_ms, w) / (1 - 4 * x(i) * v_ms * cos(heading) / g_ms)^0.5);
    horizontal_acc_bow = hv_amp(Lwl, B, Cb, v_ms, wave_amp, x(i)) * sin(pitch_amp) + (Loa/2) * (w^4 * waveamplitude / g_ms^2) * (pt_amp(Lwl, B, Cb, v_ms, wave_amp, x(i)))^2;
    s_horizontal_acc_bow(i) = x(i)^5 * horizontal_acc_bow ...
        * sspec(h13_ms, w) / (1 - 4 * x(i) * v_ms * cos(heading) / g_ms)^0.5;
end % end for "for i = ..."
% Name: s_accv3.m
% Description: This file (s_accv3.m) computes the compound vertical acceleration at fore deck, given an encounter frequency, based on the formulation given in Akinturk's thesis for the UBC series fishing vessels.

function s_vertical_acc_bow = s_accv3(x, heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb)

% B : beam of the ship [ft]
% Cb : block coefficient
% D : depth of the ship [ft]
% heading : ship's heading [rad] (head seas is 180 degrees)
% g_ms : gravity [m/s^2]
% Loa : length over all [ft]
% Lwl : waterline length [ft]
% T : draft of the ship [ft]
% v_ms : ship speed [m/s]
% w : wave frequency [rad/s]
% wave_amp: wave amplitude = 1 (Im) or [ft])
% x : encounter frequency vector [rad/s]

global g_ms

Loa = 1.05*Lwl * 0.3048;
D = 1.30*T;

for i = 1 : size(x,2);
    w = we2w (heading, v_ms, x(i));
    pitch_amp = (w^2/g_ms)*pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))... *
        sqrt(sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5);
    vertical_acc_fore = hv_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i))*cos(pitch_amp)... + (Loa/2)*(w^2/g_ms)*pt_amp(Lwl, B, T, Cb, v_ms, wave_amp, x(i));
    s_vertical_acc_bow(i) = x(i)^5 * vertical_acc_fore^2 ...
        *sspec(h13_ms,w)/(1-4*x(i)*v_ms*cos(heading)/g_ms)^.5;
end % end for "for i = ..."
% Name: rmsveraccaft
% Description: This file computes the RMS vertical acceleration at the aft deck, given an
encounter frequency, based on the formulation given in Akinturk's thesis for the UBC series
fishing vessels.

function veraccaft = rmsveraccaft(x, v_kn, range_nm, hold, heading, v_ms, wave_amp,
h13_ms, lambda, wnmax) % threshold value at aft

% g_ms : gravity [m/s^2]
% heading : ship's heading [rad]
% v_ms : ship speed [m/s]
% w_l : lower bound of wave frequency for integration [rad/s]
% w_u : upper bound of wave frequency for integration [rad/s]
% we_l : lower bound of encounter wave frequency for integration [rad/s]
% we_u : upper bound of encounter wave frequency for integration [rad/s]

global g_ms

Lwl = x(1);
B = x(2);
T = x(3);
Cb = x(4);

w_l = 0.01;
w_u = 5.01;

we_l = w2we(heading, v_ms, w_l);
we_u = w2we(heading, v_ms, w_u);

%s = quad8('s_accvl.m',we_l,we_u,[],[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

% By Simpson's method ...
s = sminteg('s_accvl',we_l,we_u,heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

veraccaft = sqrt(s);
% Name: stab.m
% Description: This file (stab.m) computes the transverse metacentric height of a loaded vessel.

function stability = stab(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

% x = [ Waterline length beam draft block coefficient ] [ft]
% v_kn = design speed [kn]
% Cm = midship sectional area coefficient
% Cp = prismatic coefficient

global density_m g_ft coeff_freeboard Cw
global ws wm payload wo wp wg

L = x(1);
B = x(2);
T = x(3);
Cb = x(4);

depth = (1 + coeff_freeboard) * x(3);
tweight(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax);

density = density_m * 0.3048^3 / 1.016;
Cn = L*B*depth/100;

% KG for hull weight:
kg_hull = depth * (Cn/14630.0 + 0.821);

% KG for outfit weight:
kg_outfit = depth * (Cn/8820.0 + 1.19);

% KG for machinery weight: This weight is included in the payload
% in the case of Transglobal Expedition Ferry.
kg_machinery = depth * ((Cn-400)^2/130900.0 + 0.494);

% KG for provision stores weight(wp):
kg_stores = depth * 0.75;

% KG for gear weight:
k_gear = depth * 1.66;

% KG for payload or cargo:
k_payload = depth * 0.60;

kg_ship = (ws*kg_hull + wm*kg_machinery + payload*kg_payload + wo*kg_outfit... + wp*kg_stores + wg*k_gear)/(ws + wm + wo + payload + wp + wg);

KG_loaded = kg_ship; % in feet

Objective function:
stability = KG_loaded/depth - 0.243*B/depth - 0.2831; % stability should be less than zero
File 28: stablite.m

% Name: stablite.m
% Description: This file (stablite.m) computes the transverse metacentric height of a light vessel.

function stability = stablite(x, vkn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

% IMPORTANT!!!! units are in feet, LongTons !!!!!

global density_m density_ft g_ft coeff_freeboard Cw
global ws wm wo wp wg

L = x(1);
B = x(2);
T = x(3);
Cb= x(4);

depth = coeff_freeboard * x(3);
tweight(x, vkn, range_nm, hold, heading, v_ms, waveamp, hl3_ms, lambda, wnmax);
density = density_ft;
Cn = L*B*depth/100;

%******************************************************************************
% CENTER OF GRAVITIES
%******************************************************************************
% KG for hull weight:
%  kg_hull = depth * (Cn/14630.0 + 0.821);
% KG for outfit weight:
%  kg_outfit = depth * (Cn/8820.0 + 1.19);
% KG for machinery weight: This weight is included in the payload
% in the case of Transglobal Expedition Ferry.
%  kg_machinery = depth * ((Cn-400)^2/130900.0 + 0.494);
% KG for provision stores weight(wp):
%  kg_stores = depth * 0.75;
% KG for gear weight:
%  kg_gear = depth * 1.66;
% KG for payload or cargo:
%  kg_payload = depth * 0.60;

kg_ship = (ws*kg_hull + wm*kg_machinery +wo*kg_outfit)/(ws + wm + wo);

KG_light = kg_ship; % in feet

Objective Function:
stability = KG_light/depth - 0.466*B/depth + 0.046; % stability should be less than zero

File 29: tm01.m

% Name: tm01.m
% Description: This file (tm01.m) computes Kimura's threshold value at the aft deck of the vessel from the first moment of power spectrum of the acceleration in longitudinal (x) and vertical (z) directions.

function tmaft = tm01(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax) % threshold value at aft

% g_ms : gravity [m/s^2]
% heading : ship's heading [rad]
% v_ms : ship speed [m/s]
% w_l : lower bound of wave frequency for integration [rad/s]
% w_u : upper bound of wave frequency for integration [rad/s]
% we_l : lower bound of encounter wave frequency for integration [rad/s]
% we_u : upper bound of encounter wave frequency for integration [rad/s]

global g_ms
Lwl = x(1);
B = x(2);
T = x(3);
Cb = x(4);

w_l = 0.01;
w_u = 5.01;

we_l = w2we(heading, v_ms, w_l);
we_u = w2we(heading, v_ms, w_u);

% By Simpson's method:
s1 = sminteg('s_acc_hl.m',we_l,we_u,[],[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);
s2 = sminteg('s_accvl.m',we_l,we_u,[],[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

tmaft = s1 + 0.1133*s2;
File 30: tm02.m

% Name: tm02.m
% Description: This file (tm02.m) computes Kimura's threshold value at the midship deck of
the vessel from the first moment of power spectrum of the acceleration in x and z
directions.

function tm02mid = tm02(x, v_kn, range_nm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax) %

threshold value at aft

% g_ms : gravity [m/s^2]
% heading : ship's heading [rad]
% v_ms : ship speed [m/s]
% w_l : lower bound of wave frequency for integration [rad/s]
% w_u : upper bound of wave frequency for integration [rad/s]
% we_l : lower bound of encounter wave frequency for integration [rad/s]
% we_u : upper bound of encounter wave frequency for integration [rad/s]

global g_ms

Lwl = x(1);
B = x(2);
T = x(3);
Cb = x(4);

w_l = 0.01;
w_u = 5.01;

we_l = w2we(heading, v_ms, w_l);
we_u = w2we(heading, v_ms, w_u);

s3 = quad8('s_acc_h2.m',we_l,we_u,[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);
s4 = quad8('s_accv2.m',we_l,we_u,[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

% By Simpson's method:
s3 = sminteg('s_acc_h2',we_l,we_u,heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);
s4 = sminteg('s_accv2',we_l,we_u,heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

tmmid = s3 + 0.1133*s4;
File 31: tm03.m

% Name: tm03.m
% Description: This file(tm03.m) computes Kimura's threshold value at the fore deck of the vessel from the first moment of power spectrum of the acceleration in x and z directions.

function tmbow = tm03(x, vkn, rangenm, hold, heading, vms, waveamp, h13_ms, lambda, wnmax) % threshold value at aft

% g_ms : gravity [m/s^2]
% heading : ship's heading [rad]
% v_ms : ship speed [m/s]
% w_l  : lower bound of wave frequency for integration [rad/s]
% w_u  : upper bound of wave frequency for integration [rad/s]
% we_l : lower bound of encounter wave frequency for integration [rad/s]
% we_u : upper bound of encounter wave frequency for integration [rad/s]

global g_ms

Lwl = x(1);
B  = x(2);
T  = x(3);
Cb  = x(4);

w_l = 0.01;
w_u = 5.01;

we_l = w2we(heading, vms, w_l);
we_u = w2we(heading, vms, w_u);

%s5 = quad8('s_acc_h3.m',we_l,we_u,[],[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);
%s6 = quad8('s_accv3.m',we_l,we_u,[],[],heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

% By Simpson's method :
s5 = sminteg('s_acc_h3',we_l,we_u,heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);
s6 = sminteg('s_accv3',we_l,we_u,heading, v_ms, wave_amp, h13_ms, Lwl, B, T, Cb);

tmbow = s5 + 0.1133*s6;
% Name: pwerfish.m
% Description: This file (pwerfish.m) computes resistance and power requirement for a
UBC Series fishing vessel. This algorithm was produced by Akinturk.

%Resistance and Powering Calculation:

function pwr = pwerfish(x, speed_kn, rangenm, hold, heading, v_ms, wave_amp, h13_ms, lambda, wnmax)

% x = [ Waterline_length beam draft block_coefficient] [ft]
% v_kn = design speed [kn]
% Cm = mid ship sectional area coefficient

global g_ft Cm ehp
L = x(1);
B = x(2);
T = x(3);
Cb = x(4);

% Cp = prismatic coefficient
Cp = Cb/Cm;

V = 1.69*speed_kn; % 1 [ft/s] = 1.69 * [kn]

%Fn is Froude Number
Fn=V/sqrt(g_ft*L);
%Rn*Reynold's Number
Rn=V*L/.929E-5;
%S, wetted surface area [ft^2]
S = L*(2*T+B)*sqrt(Cm)*(0.453+0.4425*Cb-0.2862*Cm-0.003467*B/T+0.3696*0.8274);

% Residual resistance estimation coefficients

dl0 = 0.816643*Cb- 0.427581;
dl1 = 0.022352*Cb-0.011868;
dl2 = -0.008464*Cb + 0.004504;
dl3 =-0.593119*Cb +0.313188;
dl4= 0.114583*Cb-0.060598;
cl = dl0 + dl1*(L/B) + dl2*(L/B)^2 + dl3*(B/T) + dl4*(B/T)^2;

d20 = -1.060036*Cb + 0.728888;
d21 = 0.010643*Cb - 0.006468;
d22 = 0.06869*Cb - 0.042227;
d23 = -0.054369*Cb + 0.02819;
d24 = 0.03881*Cb - 0.023296;
c2 = d20+d21*(L/B) + d22*(L/B)^2 + d23*(B/T) + d24*(B/T)^2;

d30 = 0.796988*Cb - 0.49031;
d31 = -0.004176*Cb + 0.002643;
d32 = -0.024798*Cb + 0.016158;
d33 = -0.017131*Cb + 0.009399;
d34 = -0.025976*Cb + 0.015314;

\[ c_3 = d_{30} + d_{31}(L/B) + d_{32}(L/B)^2 + d_{33}(B/T) + d_{34}(B/T)^2; \]

\[ d_{40} = 0.348833Cb - 0.233175; \]

\[ d_{41} = -0.003756Cb + 0.00232; \]

\[ d_{42} = -0.025905Cb + 0.01605; \]

\[ d_{43} = -0.024893Cb + 0.013365; \]

\[ d_{44} = -0.000131Cb + 0.00125; \]

\[ c_4 = d_{40} + d_{41}(L/B) + d_{42}(L/B)^2 + d_{43}(B/T) + d_{44}(B/T)^2; \]

\[ m = 0.14347Cp(-2.1976); \]

% Cr, Wave Resistance coefficient
\[ Cr = c_1\exp(-m/(9(Fn^2))) + c_2\exp(-m/(Fn^2))^2 + c_3\exp(-m/(Fn^2))\sin(Fn^2) \]
\[ + c_4\exp(-m/(Fn^2))\cos(Fn^2); \]

% Cf, coefficient of friction
\[ Cf = 0.075/((\log_{10}(Rn)-2)^2); \]

% Ct, total resistance coefficients
\[ Ct = Cr + Cf + 0.004; \]

% R, total resistance
\[ R = 0.5\times Ct \	imes 1.9905 \	imes S \times V^2; \]

% ehp, effective horse power
\[ ehp = R \times V / 550; \]

% pwr, required horse power
\[ pwr = 2.0 \times ehp; \]
File 33: Sminteg.m

% Name: Sminteg.m
% Description: This file (Sminteg.m) defines the mathematics of integration by Simpson’s method.

function simpson_integration = sminteg(funname, x1, x2, varargin)

N = 2001;
sm(1) = 1;
deltaX = (x2 - x1) / N;
x(1) = x1;

for i = 2:2:N-1
    sm(i) = 4; sm(i+1) = 2;
    x(i) = x1 + deltaX*i; x(i+1) = x1 + deltaX*(i+1);
end;
sm(N) = 1;
x(N) = x2;
y = feval(funname,x, varargin{:});
simpson_integration = deltaX * sum(sm.*y) / 3.0;

File 34: tmintp.m

% Name: tmintp.m
% Description: This file (tmintp.m) is used in the optimization to find the minimum threshold value.

tmintp.m

function f = tmintp(x)

global x0crit y0crit
f = interp1(x0crit,y0crit,x, 'linear');

File 35: tmmax.m

% Name: tmmax.m
% Description: This file (tmmax.m) is used in the optimization to find the maximum threshold value.

tmax.m
function \( f = \text{tmmax}(x) \)

global \( x0 \) \( y0 \) \( \text{crit} \)

\( f = -\text{interp1}(x0, y0, \text{crit}, '\text{spline}'); \)

**File 36: vcrit.m**

% **Name:** vcrit.m

% **Description:** This file (vcrit.m) produces the maximum attainable velocity at any location (fore, midship or aft) at which Kimura's threshold value will just be .22. This is useful in deciding the location of the workspace onboard the vessel for maximum comfort and safety of the crews.

vcrit.m

function \( f = \text{vcrit}(x) \)

global \( x0 \) \( y0 \) \( \text{crit} \)

\( y = \text{interp1}(x0, y0, \text{crit}, 'linear'); \)

\( f = (y - 0.22)^2; \)
File 37: oku1.m

% Name: oku1.m
% Description: This function reads in the values of Kimura’s threshold value (tm) calculated previously and produces plots of tm versus speed for each sea state. This also calculates maximum attainable speeds for a given operational sea state and produces plots of it.

v_kn = [0 1 2 3 4 5 6 7 8 9 10]; % ship speeds [kn]
swh = [0.30 0.88 1.40 2.45 3.65]; % significant wave heights [m]
ss = [1 2 3 4 5]; % sea states
ships = [36.679 42.403 49.806 55.895 65.73 65.453 69.204 75.503 79.139 84.256 90.438]; % design number

len_ss = size(swh,2);
len_speed = size(v_kn,2);
len_ships = size(ships,2);
for ii = 1:len_speed;
    tmcri(ii) = 0.22; % Kimura’s limiting value
end; % ii

fid = fopen('mizan.out','r');

tmaft(1,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmaft(2,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmaft(3,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmaft(4,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmaft(5,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);

tmmid(1,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmmid(2,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmmid(3,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmmid(4,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmmid(5,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);

tmbow(1,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmbow(2,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmbow(3,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmbow(4,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);
tmbow(5,1:len_speed,1:len_ships) = fscanf(fid,'%10g',[len_speed len_ships]);

st = fclose(fid);

for i = 1:len_ships; % number of ships
    len_ships = 11;

- 126 -
figure;
title1 = strcat('ship no:',num2str(i),' (aft ship)'); %subplot(1,1,1),
plot(v_kn, tmcri,'k', v_kn,tmaft(1,1:len_speed,i),'k-o', v_kn,tmaft(2,1:len_speed,i),'b:*', ...
    v_kn,tmaft(3,1:len_speed,i),'g-+', v_kn,tmaft(4,1:len_speed,i),'r:x', ...
    v_kn,tmaft(5,1:len_speed,i),'k-d');

v = axis;
v(3) = -0.225; v(4) = 0.225;
axis(v); xlabel('Ship speed [kn]');ylabel('tm value');title(title1);
legend(' tm limit ', 'sea state: 1 ', 2 ', 3 ', ...
        4 ', 5 ',0);

figure;
title2 = strcat('ship no:',num2str(i),' (mid ship)'); %subplot(2,1,2),
plot(v_kn, tmcri,'k', v_kn,tmmid(1,1:len_speed,i),'k-o', v_kn,tmmid(2,1:len_speed,i),'b:*', ...
    v_kn,tmmid(3,1:len_speed,i),'g-+', v_kn,tmmid(4,1:len_speed,i),'r:x', ...
    v_kn,tmmid(5,1:len_speed,i),'k-d');

v = axis;
v(3) = -0.225; v(4) = 0.225;
axis(v);
axis(v); xlabel('Ship speed [kn]');ylabel('tm value');title(title2);
legend(' tm limit ', 'sea state: 1 ', 2 ', 3 ', ...
        4 ', 5 ',0);

figure;
title3 = strcat('ship no:',num2str(i),' (fore ship)'); %subplot(1,1,3),
plot(v_kn, tmcri,'k', v_kn,tmbow(1,1:len_speed,i),'k-o', v_kn,tmbow(2,1:len_speed,i),'b:*', ...
    v_kn,tmbow(3,1:len_speed,i),'g-+', v_kn,tmbow(4,1:len_speed,i),'r:x', ...
    v_kn,tmbow(5,1:len_speed,i),'k-d');

v = axis;
v(3) = -0.225; v(4) = 0.225;
axis(v);
axis(v); xlabel('Ship speed [kn]');ylabel('tm value');title(title3);
legend(' tm limit ', 'sea state: 1 ', 2 ', 3 ', ...
        4 ', 5 ',0);
end;

for i = 1:len_ss; % sea states
    for j = 1:len_ships; % ships

        len_ships = 11; % defines for which ship is the result being sought.

- 127 -
global x0crit y0crit
x0crit = v_kn(1:len_speed); y0crit(1:len_speed) = tmaft(i,1:len_speed,j);
vmin(i,j) = fmin('tmintp',v_kn(1), v_kn(len_speed)); % tm value is min for the speed
v_kn(7)
vmax(i,j) = fmin('tmmmax',v_kn(1), v_kn(len_speed)); % tm value is max for the speed
v_kn(7)

vc1(i,j) = fmin('vcrit', v_kn(1), vmin(i,j));
vc2(i,j) = fmin('vcrit', vmin(i,j), v_kn(len_speed));

if interp1(x0crit,y0crit,vmax(i,j), 'spline') < 0.22
vc3(i,j) = v_kn(len_speed);
elseif vc1(i,j) > vc2(i,j)
vc3(i,j) = vc1(i,j);
else
vc3(i,j) = vc2(i,j);
end; % if

clear x0crit y0crit
end; % j
end; % i

figure;
plot(ss,vc1(1:len_ss,11),g', ss,vc2(1:len_ss,11),k*,...
ss,vc3(1:len_ss,11),b-,ss,vmin(1:len_ss,11),c:o');
legend(' Maximum attainable (Region 1)', ' Maximum attainable (Region 2)',...
' Maximum attainable (Overall)', 'Corresponding to min tm value',1);
xlabel('Sea state');ylabel('Ship speed [kn]');title('aft');
h = gca;
set(h,'xtick',[1 2 3 4 5]);

for i = 1:len_ss; % sea states
for j = 1:len_ships; % ships

len_ships = 11;

global x0crit y0crit
x0crit = v_kn(1:len_speed); y0crit(1:len_speed) = tmmid(i,1:len_speed,j);
vmin(i,j) = fmin('tmintp',v_kn(1), v_kn(len_speed)); % tm value is min for the speed
v_kn(7)
vmax(i,j) = fmin('tmmax', v_kn(1), v_kn(len_speed)); % tm value is max for the speed
vmax

v_kn(7))
vc1(i,j) = fmin('vcrit', v_kn(1), vmin(i,j));
vc2(i,j) = fmin('vcrit', vmin(i,j), v_kn(len_speed));

if interp1(x0crit,y0crit,vmax(i,j), 'spline') < 0.22
vc3(i,j) = v_kn(len_speed);
elseif vc1(i,j) > vc2(i,j)
vc3(i,j) = vc1(i,j);
else
vc3(i,j) = vc2(i,j);
end; % if

clear x0crit y0crit
end; % j
end; % i

figure;
plot(ss,vc1(1:len_ss,11),'g', ss,vc2(1:len_ss,11),'k*', ...
ss,vc3(1:len_ss,11),'b-',ss,vmin(1:len_ss,11),c'o');
legend(' Maximum attainable (Region 1)', 'Maximum attainable (Region 2)', ...
'Maximum attainable (Overall)', 'Corresponding to min tm value', 1);
xlabel('Sea state');ylabel('Ship speed [kn]');title('mid');
h = gca;
set(h,'xtick',[1 2 3 4 5]);

for i = 1:len_ss; % sea states
for j = 1:len_ships; % ships

len_ships = 11;

global x0crit y0crit
x0crit = v_kn(1:len_speed);
y0crit(1:len_speed) = tmbow(i,1:len_speed,j);
vmin(i,j) = fmin('tmintp', v_kn(1), v_kn(len_speed)); % tm value is min for the speed
vmin
v_kn(7))
vmax(i,j) = fmin('tmmax', v_kn(1), v_kn(len_speed)); % tm value is max for the speed
vmax

v_kn(7])
vc1(i,j) = fmin('vcrit', v_kn(1), vmin(i,j));
vc2(i,j) = fmin('vcrit', vmin(i,j), v_kn(len_speed));
if interp1(xOcrit,yOcrit,vmax(i,j), 'spline') < 0.22
    vc3(i,j) = v_{kn}(len_{speed});
elseif vc1(i,j) > vc2(i,j)
    vc3(i,j) = vc1(i,j);
else
    vc3(i,j) = vc2(i,j);
end; % if

clear xOcrit yOcrit
end; % j
end; % i

figure;
plot(ss,vc1(1:len_ss,11),'g', ss,vc2(1:len_ss,11),'k*', ...
     ss,vc3(1:len_ss,11),'b-',ss,vmin(1:len_ss,11),'c:o');
legend(' Maximum attainable (Region 1)', ' Maximum attainable (Region 2)', ...
       ' Maximum attainable (Overall)', 'Corresponding to min tm value',1);
xlabel('Sea state');ylabel('Ship speed [kn]');title('bow');
h = gca;
set(h,'xtick',[1 2 3 4 5]);
File 38: maintrgb.m

% Name: maintrgb.m
% Description: The file maintrgb.m is the main file in the design process for Transglobal Expedition, which is used for optimization purpose to get the values of different design variables based on the objective function and the constraint function.

format short g

fid = fopen('r_pk40_3.out','a+');% 'w' deletes contents in existing file or create a new file, and open it for writing; 'a+' create and open new file for reading and writing, appending to the end of the file. The prefix 1-5 denotes seastate.

fprintf(fid,' seast hold vel L B T Cb Disp HP Cost RMS_hv RMS_pt Peak_hv Peak_pt stab stablite');
end

% Global parameters and variables:

global density gravity gms g_fts
global freeboard depth length beam draft
global Cm Cp CwCb
global h13 h13_5 lambda wnmax
global speedmax rangemax
global powermax ehp
global payload ws wo wp wm wg wf

% density of sea water [LT/ft^3]
density_ft = 1/(35.3147 / 1.016);
% density of sea water [LT/m^3]
density_m = 1.025/1.016;

g_fts = 32.2; % gravitational acc. ft/s
g_ms = 9.807; % gravitational acc. m/s
density = density_ft;
density = density_m;
gravity = g_fts;

seastate = 3; % seastate varies from 1 to 5.
h13_ref = [0.30 0.88 1.40 2.45 3.65];
t_ref = [2.4 3.9 4.6 5.7 6.8];
tmax_ref = [3.4 5.4 6.5 8.1 9.7];
lambda_ref = [6.1 15.9 21.6 34.0 49.0];
h13 = h13_ref(seastate);
h13_5 = 3.65; % [m]
%h13 = lambda_ref(seastate); % [m]; for seastate 5.

lambda = lambda_ref(seastate); % [m]
wnmax = 2*pi/tmax_ref(seastate); % [rad/s]
speedmax = 10.0; % [kn]
rangemax = 200.0; % [km]
powermax = 150.0; % [HP]
hold = [10 25 50 75 100 125 150 175 200 250 300];

% Addition by Ayhan
% begin
results = zeros(size(hold,2), 16);
results(:,1) = seastate;
%end

for igen = 1:size(hold,2);
%for igen = 5,9; % for element # 5th and 9th.
payload = hold(igen);
% Addition by Ayhan
% begin
results(igen, 2) = payload;
results(igen, 3) = speedmax;
% end
%freeboard = 2.50;
%Cm = 0.878;
%Cp = 0.7;
cargoweight = payload;

L_hold = 3.3*(28.5*cargoweight)*0.3669;
L_licence = (speedmax^2)/1.4; %this restriction is applicable for westcoast

%L0 = L_hold;
L0 = l_est(hold(igen));

x0 = [L0 (L0/3) (L0/4/3) 0.6 0.87 0.7 ]; % in feet
x = x0;
vlb = [L0*0.95 x(1)/4.0 x(2)/4 0.5 .85 .65 ];

% Addition by Ayhan
% begin
vub = [L0*1.3 L0/(0.0177*L0 + 1.690323)*1.1

x(2)/2.6 0.65 .90 .75 ]; % in feet
%x0 = vlbr + (vub-vlb)/2;
%x = x0;
length = x(1); % in feet
beam = x(2); % in feet
draft = x(3); % in feet
Cb = x(4); % block coefficient
Cm = x(5); % midship coefficient
Cp = x(6); % prismatic coefficient

pwerfish(x0); stab(x0); rms_hv(x0); tweight(x0); rms_pch(x0);peak_hv(x0);peak_pch(x0); stablite(x0);

% displacement = density*cb*length*beam*draft
% displacement = density*x(1)*x(2)*x(3)*x(4);

funf = 'f = [cost(x)];%[pwerfish(x)];% cost(x)% tweight(x)%;pwer(x); ((1/peak_hv(x))+(1/peak_pch(x)))

- 132 -
***

% Max: L/B=4.99; B/T=5.83; L/T=18.5
% Min: L/B=1.44; B/T=1.80; L/T=4.05

%fung = 'g = [(1.025/1.016)*x(1)*x(2)*x(3)*x(4)/tweight(x)-1.0; (1.025/1.016)*x(1)*x(2)*x(3)*x(4)/tweight(x)-1.015; x(1)/x(2)+1.99; x(1)/x(2)-4; -x(2)/x(3)+1.99; x(2)/x(3)-6.5];';
% fung = 'g = [(1.025/1.016)*x(1)*x(2)*x(3)*x(4)/tweight(x); x(1)/x(2)+1.99; x(1)/x(2)-2.5; -x(2)/x(3)+1.99; x(2)/x(3)-6.5];';
% (rms_hv(x)/0.3048)-2.0; (rms_pch(x))-3.0; (-peak_hv(x)/l.3659+1.4;); (-peak_pch(x)/l.3659+2);';
% fung = 'g = [x(1)*x(2)*x(3)*x(4)/(35.0045867*tweight(x))-1.0; rms_hv(x)-3.2808; rms_pch(x)-5.0];';
% fung = 'g = [x(1)*x(2)*x(3)*x(4)/(35.0045867*tweight(x))-1.0];';
% (x(2)*x(4)/x(3)-2.3; -x(2)*x(4)/x(3)+1.5; x(1)/x(2)+1.44; x(1)/x(2)-4.99; x(2)/x(3)-5.83; -x(2)/x(3)+1.8; x(1)/x(3)-18.5; -x(1)/x(3)+4.05];';
% fung = 'g = [(1.025/1.016)*x(1)*x(2)*x(3)*x(4)-tweight(x); rms_hv(x); 3.0-peak_pch(x)];';
% fung = 'g = [x(2)/x(3)-4.0; -x(2)/x(3)+1.5; -x(l)/x(2)+1.44; x(l)/x(2)-4.99; x(2)/x(3)-5.83; -x(2)/x(3)+1.8; x(1)/x(3)-18.5; -x(1)/x(3)+4.05];';
% fung = 'g = [x(2)*x(4)/x(3)-1.7];';

***

% SOME EXAMPLES OF CONSTRAINTS:
% -x(4)+.6; Cb constraint Cb>.6
% x(1)/x(2)-4.99; -x(1)/x(2)+1.44; L/B constraint 1.44<L/B<4.99
% x(2)/x(3)-5.83; -x(2)/x(3)+1.80; B/T constraint 1.80<B/T<5.83
% x(1)/x(3)-18.5; -x(1)/x(3)+4.05; L/T constraint 4.05<L/T<18.5
% x(2)*x(4)/x(3)-1.7; wetted surface minimization
% pwr(x)-150.0; Power constraint < 150 hp
% rms_hv(x)/0.3048-2.0; RMS Heave constraint
% rms_pch(x)-3.0; RMS Pitch constraint
% peak_hv(x)/0.3048-2.5; peak heave constraint
% peak_pch(x)-4; peak pitch constraint

fun = [fung fung]
disp('optimization started ...')

options(1) = 1;
options(2) = 1.0e5;  % Termination tolerance for x(default 1e-4).
options(3) = 1.0e6;  % Termination tolerance for F(default 1e-4).
options(13) = 1;  % Number of equality constraints.
options(14) = 1000000;  % Maximum number of iterations.

[x options] = constrffun, xO, options, vlb, vub);

x_feet = x
x_meter = [x(1)*0.3048 x(2)*0.3048 x(3)*0.3048 x(4) x(5) x(6)]
speedmax
disp('displacement'); disp(x)
disp('Total weight'); tweight(x)
disp('Power'); pwerfish(x)
disp('Total Cost'); cost(x)
disp('RMS Heave'); rms_hv(x)
disp('RMS Pitch'); rms_pch(x)
disp('Peak_Heave Ratio'); peak_hv(x)
disp('Peak_Pitch Ratio'); peak_pch(x)
disp('Transverse_GMt_loaded'); stab(x)
disp('Transverse_GMt_light'); stablite(x)

% Addition by Ayhan
% begin
results(igen,4) = x(1);
results(igen,5) = x(2);
results(igen,6) = x(3);
results(igen,7) = x(4);
results(igen,8) = displacex(x);
results(igen,9) = pwerfish(x);
results(igen,10) = cost(x);
results(igen,11) = rms_hv(x);
results(igen,12) = rms_pch(x);
results(igen,13) = peak_hv(x);
results(igen,14) = peak_pch(x);
results(igen,15) = stab(x);
results(igen,16) = stablite(x);

fprintf(fid,'\n%3.0f%6.1f%6.1f%8.3f%8.3f%8.3f%8.3f%8.3f%8.3f%8.3f%8.3f%10.3f
%10.3f%12.3f%12.3f',results(igen,:));

end % end of main for loop, counter is "igen"