<table>
<thead>
<tr>
<th>RD-R754 595</th>
<th>AN AUTOMATIC CONTROL DESIGN FOR THE MARINER CLASS SHIPS 1/2</th>
<th>(U) NAVAIR POSTGRADUATE SCHOOL MONTEREY CA T TANSAN DEC 84</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNCLASSIFIED</td>
<td>F/G 13/10</td>
<td>NL</td>
</tr>
</tbody>
</table>
AN AUTOMATIC CONTROL DESIGN FOR THE MARINER CLASS SHIPS
by
Tayfun Tansan
December 1984

Thesis Advisor: George J. Thaler

Approved for public release; distribution is unlimited
An Automatic Control Design for the Mariner Class Ships

Tayfun Tansan

Naval Postgraduate School
Monterey, California 93943

Naval Postgraduate School
Monterey, California 93943

December 1984

December 1984

99

Approved for public release; distribution is unlimited.


The increase in fuel prices has initiated considerable interest by ship operators in new ship autopilots which minimize the propulsion losses due to steering.

This research presents the results of work on a steering design for the Mariner class ship based on a computer simulation. A model of the Mariner class ship was coupled to a function minimization subroutine to minimize the added resistance (cont.)
ABSTRACT (Continued) 

caused by rudder activity and hull drag of inertial origins 
caused by periodic yawing of ship in seaway.

The Mariner class ship computer model was tested in calm 
water and in seaway. The optimal controller parameters are 
shown in look up tables as functions of ship speed, sea state, 
encounter angle and encounter frequency. This technique can be 
used as an adaptive controller.
An Automatic Control Design for the Mariner Class Ships

by

Tayfun Tansan
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1978

Submitted in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the
NAVAL POSTGRADUATE SCHOOL
December 1984

Author: Tayfun Tansan

Approved by: George J. Thaler, Thesis Advisor

A. Gerba, Second Reader

Harriett B. Rigas, Chairman, Department of Electrical and Computer Engineering

John A. Dyer, Dean of Science and Engineering
ABSTRACT

The increase in fuel prices has initiated considerable interest by ship operators in new ship autopilots which minimize the propulsion losses due to steering.

This research presents the results of work on a steering design for the Mariner class ship based on a computer simulation. A model of the Mariner class ship was coupled to a function minimization subroutine to minimize the added resistance caused by rudder activity and hull drag of inertial origins caused by periodic yawing of ship in seaway.

The Mariner class ship computer model was tested in calm water and in a seaway. The optimal controller parameters are shown in look up tables as functions of ship speed, sea state, encounter angle and encounter frequency. This technique can be used as an adaptive controller.
# TABLE OF CONTENTS

I. INTRODUCTION ........................................... 11

II. NONLINEAR EQUATIONS OF MOTION .......................... 13

III. OPEN LOOP SHIP'S BEHAVIOR IN CALM WATER AND SOME SEA STATES ............................... 20  
   A. CALM WATER CASE ..................................... 20  
   B. SEA STATE CASE ....................................... 20

IV. NOMOTO MODEL OF THE MARINER .......................... 24  
   A. MATHEMATICAL APPROACH ............................... 24  
   B. COMPUTER APPROACH .................................. 25

V. DERIVATION OF A COST FUNCTION FOR THE MARINER SHIP ............................................. 28

VI. CONTROLLER DESIGN FOR THE MARINER, USING FORTRAN PROGRAM ................................. 34  
   A. CALM WATER CASE ..................................... 34  
   B. REGULAR SEAS CASE ................................... 36

VII. AN APPROACH TO AN ADAPTIVE AUTOPILOT ............... 64

VIII. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE STUDY ........................................... 67  
   A. CONCLUSIONS ........................................... 67  
   B. RECOMMENDATIONS FOR FUTURE STUDY .................. 68

APPENDIX A: THESIS FORTRAN ............................... 70

APPENDIX B: DATA FOR SEA STATE PROGRAM .................. 74

APPENDIX C: PROGRAM TO CALCULATE OPTIMAL GAINS ........... 78

LIST OF REFERENCES ......................................... 96
# LIST OF TABLES

I. Assessment of the Coefficients in the X_Equation ........................................ 17

II. Assessment of the Coefficients in the Y_Equation ........................................ 18

III. Assessment of the Coefficients in the N_Equation ........................................ 19

IV. The Nomoto Model Parameters for Mariner .................................................. 27

V. Values of Weighting Factor for Different Speeds of The Mariner Class Ship .......... 33

VI. Optimal Controller Parameters I ................................................................. 35

VII. Optimal Controller Parameters II ............................................................... 35

VIII. Optimal Controller Parameters III ............................................................. 35

IX. Optimal Controller Parameters IV ................................................................. 36

X. Optimal Controller Parameters V ................................................................. 37

XI. Optimal controller Parameters VI ................................................................. 37

XII. Optimal Controller Parameters VII .............................................................. 37

XIII. Optimal Controller Parameters VIII ............................................................ 38

XIV. Optimal Controller Parameters IX ............................................................... 38

XV. Optimal Controller Parameters X ................................................................. 38

XVI. Optimal Controller Parameters XI ............................................................... 39

XVII. Optimal Controller Parameters XII ............................................................ 39

XVIII. Location of the Ship .......................................................... ........................ 47

XIX. Test for LAMBDA Value I ................................................................. 49

XX. Test for LAMBDA Value II ................................................................. 50

XXI. Sea State vs Range for Wave Height ......................................................... 78

XXII. Sea State vs Wave Height ................................................................. 79
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Time Response of U, V AND R in Calm Water when D = 1 Degree</td>
<td>22</td>
</tr>
<tr>
<td>3.2</td>
<td>Time Response of U, V and R in Regular Seas</td>
<td>23</td>
</tr>
<tr>
<td>4.1</td>
<td>Determination of Third and Second Order Nomoto Models</td>
<td>26</td>
</tr>
<tr>
<td>6.1</td>
<td>Various Structure Controllers</td>
<td>34</td>
</tr>
<tr>
<td>6.2</td>
<td>Rudder Angles in Degrees for Sea State 6 and 8</td>
<td>40</td>
</tr>
<tr>
<td>6.3</td>
<td>Heading Errors in Degrees for Sea State 6 and 8</td>
<td>41</td>
</tr>
<tr>
<td>6.4</td>
<td>The Cost vs. K</td>
<td>42</td>
</tr>
<tr>
<td>6.5</td>
<td>The Cost vs. Z1</td>
<td>43</td>
</tr>
<tr>
<td>6.6</td>
<td>The Cost vs. P1</td>
<td>44</td>
</tr>
<tr>
<td>6.7</td>
<td>The Cost vs. Z1 and P1</td>
<td>45</td>
</tr>
<tr>
<td>6.8</td>
<td>Orientation of Space Axes and Moving Axes</td>
<td>46</td>
</tr>
<tr>
<td>6.9</td>
<td>Simulation Results-Steady State 600 sec</td>
<td>48</td>
</tr>
<tr>
<td>6.10</td>
<td>The Cost Curves vs. Z1 and P1 where-Parameters are Changing Around the Optimal Values for the Calm Water Case</td>
<td>51</td>
</tr>
<tr>
<td>6.11</td>
<td>The Cost Curves vs. Z1 and P1 when Parameters Changing Around the Optimal Values for the Sea State Case</td>
<td>52</td>
</tr>
<tr>
<td>6.12</td>
<td>Open Loop Steering Model</td>
<td>54</td>
</tr>
<tr>
<td>6.13</td>
<td>Open Loop System BODE and NICHOLS Plots. (Using Comp.'A')</td>
<td>55</td>
</tr>
<tr>
<td>6.14</td>
<td>Open Loop System BODE and NICHOLS Plots. (Using Comp.'B')</td>
<td>56</td>
</tr>
<tr>
<td>6.15</td>
<td>Rudder Motion in Case I with Case I and Case II Parameters</td>
<td>58</td>
</tr>
<tr>
<td>6.16</td>
<td>Heading Error in Case I with Case I and Case II Parameters</td>
<td>59</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>6.17</td>
<td>Rudder Motion in Case II with Case II and Case I Parameters</td>
<td>59</td>
</tr>
<tr>
<td>6.18</td>
<td>Heading Error in Case II with Case II and Case I Parameters</td>
<td>60</td>
</tr>
<tr>
<td>6.19</td>
<td>Rudder Motion in Case III with Case III and Case I Parameters</td>
<td>60</td>
</tr>
<tr>
<td>6.20</td>
<td>Heading Error in Case III with Case III and Case I Parameters</td>
<td>61</td>
</tr>
<tr>
<td>6.21</td>
<td>Rudder Motion in Case I with Case I and Case III Parameters</td>
<td>61</td>
</tr>
<tr>
<td>6.22</td>
<td>Heading Error in Case I with Case I and Case III Parameters</td>
<td>62</td>
</tr>
<tr>
<td>7.1</td>
<td>Adaptive Control Scheme</td>
<td>65</td>
</tr>
<tr>
<td>C.1</td>
<td>Algorithm to Calculate Optimal Gains</td>
<td>77</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENT

First I would like to thank the Turkish Navy and the United States Navy and the special relationship between these navies that permitted the completion of my research. I want to thank Lt. Vincente Garcia for his cooperation and teamwork that existed throughout my thesis project was most essential to its final completion. And, most important, I want to thank Professor George J. Thaler for his invaluable assistance.
I. INTRODUCTION

Claims by many researchers indicate that a carefully designed controller could reduce fuel consumption by minimizing the propulsion losses which are caused by added drag due to steering of the ship.

The goal of this thesis was aimed at developing and demonstrating the utility of an improved steering control for the Mariner class ship. The immediate goal of this research was to develop design methodology for an adaptive autopilot that would provide effective steering control with associated cost savings for a full range of seaway and stability conditions.

To simulate the ship in the computer program, ship's nonlinear equations of motion were needed. Chapter 2 addresses the Mariner class ship nonlinear equations of motion.

Chapter 3 addresses the work on testing the ship simulation model and open loop ship's behaviors in calm water and in a seaway.

The basic Nomoto models give an adequate description of ship steering dynamics for design. The Nomoto second and third order models were developed from the ship's linear equations of motion Chapter 4 addresses the Mariner class ship Nomoto model. Chapter 5 shows an adequate cost function which represents the added drag due to steering and includes derivations for evaluating the weighting factor. Also Chapter 5
\[ X = A^*X + B^*U \text{ then, } X(s) = (s^*I-A)^t*B^*U \text{ (assuming initial conditions are zero.)} \]

\[
\begin{bmatrix}
V \\
R
\end{bmatrix} = \begin{bmatrix}
.2101^s + .0534 \\
-.0038^s - .0002
\end{bmatrix}
\]

\[ \frac{D}{(7.67^s+1)(116.93^s+1)} \]

\[
R(s) = s^*\text{YAW}(s)
\]

\[
\text{so, then } \frac{\text{YAW}(s)}{\text{D}(s)} = \frac{-0.189(18.34^s+1)}{s(7.67^s+1)(116.93^s+1)}
\]

result is \( K=0.189 \), \( z=18.34 \), \( P1=7.67 \) and \( P2=116.93 \)

Proceeding to the second order Nomoto equation:

\[ \frac{\text{YAW}(s)}{\text{D}(s)} = \frac{K}{s(P1^s+1)} \]

Deriving the second order Nomoto transfer function from the yaw equation only, the result is \( K=0.03 \) and \( P1=10 \).

**B. COMPUTER APPROACH**

We used a function minimization subroutine to obtain parameters of the transfer functions. Figure 4.1 shows the scheme used to obtain the third and second order Nomoto transfer functions. The results are given in Table IV.
IV. NOMOTO MODEL OF THE MARINER

To find a model which can be used in computer simulation the Mariner's linear equations of motion and its hydrodynamic coefficients were used and third and second order Nomoto transfer functions derived. Values used were for ship speed of 15 knots.

Mariner's linear equations of motion are

\[
(m - X_u)\ddot{U} = X_u\dot{D}U \\
(m - Y_v)\ddot{V} - Y_v\dot{V} = (Y_R - m^*X_G)\dot{R} + (Y_R - m^*U)\dot{R} \\
(I - N_R)\ddot{R} - (N_R - m^*X_G^*U)\dot{R} = (N_R - m^*X_G)\dot{V} + N_v\dot{V}
\]

A. MATHEMATICAL APPROACH

Proceeding to the third order Nomoto equation:

\[
\frac{YAW(s)}{D(s)} = \frac{K(s)(Z^2s + 1)}{s(P_1^2s^2 + 1)(P_2^2s + 1)}
\]

Deriving the third order Nomoto transfer function from the sway and yaw equations, we show them in matrix notation as follows.

\[
\begin{bmatrix}
V \\
R
\end{bmatrix} = \begin{bmatrix}
-0.0372 & -8.42 \\
-0.0003 & -0.10
\end{bmatrix} \begin{bmatrix}
V \\
R
\end{bmatrix} + \begin{bmatrix}
0.210 \\
-0.003
\end{bmatrix} \cdot D
\]

24
Figure 3.2  Time Response of U, V and R in Regular Seas.
Figure 3.1  Time Response of U, V AND R in Calm Water when D = 1 Degree.

22
The disturbance forces, moments, and added mass, added inertia terms were found by running the sea state program that is presented in [Ref. 4]. Data for a Mariner class ship and chosen sea conditions that were used in the sea state program are shown in APPENDIX B.

For observation purposes 'FX' (disturbance force in surge), 'FY' (disturbance force in sway) and 'MZ' (disturbance moment in yaw) were added into the surge, sway and yaw equations that were used in the simulation program.

For regular seas the program has been run a few times. Every time different FX, FY, MZ and coefficients were used to represent different ship characteristics such as ship speed, loading etc. and environmental conditions such as sea state, encounter angle, encounter frequency. Results show that U, V and R are sine waves with amplitude and phase depending on ship and environmental conditions.

Time response of U, V and R in 200 seconds are presented in Figure (3.2), for ship speed 15 knots, encounter angle 030.0 degree, encounter frequency 0.60 radian/second and sea state 6.

\[\text{During this research, displacements (Molded) up to 32 ft were chosen as a loading condition for the Mariner.}\]
III. OPEN LOOP SHIP'S BEHAVIOR IN CALM WATER AND SOME SEA STATES

A. CALM WATER CASE

Using a ship's nonlinear equations of motion and Mariner class ship coefficients, a simulation program THESIS FORTRAN was developed and run to observe $U$, $V$ and $R$. The computer program THESIS FORTRAN is shown in Appendix A.

First run $D=0$, $\text{YAWC}=0$, $U_l=15$ knots were applied and it was seen that the ship stays on its initial course and speed. $U=15$ knots, $V=R=0$.

The program was rerun a few times, changing the rudder angle to 2.8 degrees to both sides (port and starboard) and it was observed that increasing the rudder angle changes the ship's course and also $U$ decreased, the absolute values of $V$ and $R$ increased. After a few hundred seconds $U$, $V$ and $R$ reached steady values independently. These steady values depend on rudder angle; large rudder angles decrease $U$ and increase $V$ and $R$. For a rudder angle of one degree and constant speed ($U_l$) of 15 knots, the first 200 seconds of time response of $U$, $V$ and $R$ are shown in figure 3.1 as an example.

B. SEA STATE CASE

To observe the behavior of the ship in a sea state, disturbance forces and moments are needed that depend on sea state, ship speed, encounter angle and encounter frequency. Also in sea state hydrodynamic parameters are changed, i.e., the added mass and added inertia are functions of encounter frequency and sea state.
TABLE III
Assessment of the Coefficients in the N_Equation

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Value of Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \dot{V} )</td>
<td>((m \times X_G - N_V))</td>
<td>-0.00478</td>
</tr>
<tr>
<td>( \dot{R} )</td>
<td>((I_z - N_R))</td>
<td>0.0175</td>
</tr>
<tr>
<td>( V )</td>
<td>(N_V)</td>
<td>-0.0555</td>
</tr>
<tr>
<td>( V^2 )</td>
<td>((1/6) \times N_{vvv})</td>
<td>0.345</td>
</tr>
<tr>
<td>( V \times R^2 )</td>
<td>(0.5 \times N_{vrr})</td>
<td>0</td>
</tr>
<tr>
<td>( V \times D^2 )</td>
<td>(0.5 \times N_{vdd})</td>
<td>0.00264</td>
</tr>
<tr>
<td>( V \times D U )</td>
<td>(N_V)</td>
<td>-</td>
</tr>
<tr>
<td>( V \times D U^2 )</td>
<td>(0.5 \times N_{vvv})</td>
<td>-</td>
</tr>
<tr>
<td>( R )</td>
<td>((N_R - m \times X_G \times U1))</td>
<td>-0.0349</td>
</tr>
<tr>
<td>( R^2 )</td>
<td>((1/6) \times N_{rrr})</td>
<td>0</td>
</tr>
<tr>
<td>( R \times V^2 )</td>
<td>(0.5 \times N_{rvv})</td>
<td>-1.158</td>
</tr>
<tr>
<td>( R \times D^2 )</td>
<td>(0.5 \times N_{rdd})</td>
<td>0</td>
</tr>
<tr>
<td>( R \times D U )</td>
<td>(N_R)</td>
<td>-</td>
</tr>
<tr>
<td>( R \times D U^2 )</td>
<td>(0.5 \times N_{rvv})</td>
<td>-</td>
</tr>
<tr>
<td>( D )</td>
<td>(N_D)</td>
<td>-0.0293</td>
</tr>
<tr>
<td>( D^2 )</td>
<td>((1/6) \times N_{ddd})</td>
<td>0.00482</td>
</tr>
<tr>
<td>( D \times V^2 )</td>
<td>(0.5 \times N_{dvv})</td>
<td>0.1032</td>
</tr>
<tr>
<td>( D \times R^2 )</td>
<td>(0.5 \times N_{drr})</td>
<td>0</td>
</tr>
<tr>
<td>( D \times D U )</td>
<td>(N_D)</td>
<td>0</td>
</tr>
<tr>
<td>( D \times D U^2 )</td>
<td>(0.5 \times N_{dvv})</td>
<td>-</td>
</tr>
<tr>
<td>( V \times R \times D )</td>
<td>(N_{vrd})</td>
<td>0</td>
</tr>
<tr>
<td>( D U )</td>
<td>(N_D)</td>
<td>0.00059</td>
</tr>
<tr>
<td>( D U^2 )</td>
<td>(N_D)</td>
<td>-</td>
</tr>
</tbody>
</table>
TABLE II
Assessment of the Coefficients in the $Y$ Equation

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Value of Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\dot{Y}$</td>
<td>$(m - Y)$</td>
<td>0.327</td>
</tr>
<tr>
<td>$\dot{\dot{Y}}$</td>
<td>$(m^2X_g - Y)$</td>
<td>-0.0018</td>
</tr>
<tr>
<td>$Y$</td>
<td>$Y_v$</td>
<td>-0.244</td>
</tr>
<tr>
<td>$Y^3$</td>
<td>$(1/6)*Y_{vvv}$</td>
<td>-1.702</td>
</tr>
<tr>
<td>$Y^2R^2$</td>
<td>$0.5*Y_{vrr}$</td>
<td>0</td>
</tr>
<tr>
<td>$Y^2D^2$</td>
<td>$0.5*Y_{vdd}$</td>
<td>-0.0008</td>
</tr>
<tr>
<td>$Y^2DU$</td>
<td>$Y_{vuu}$</td>
<td>-</td>
</tr>
<tr>
<td>$Y^2DU^2$</td>
<td>$0.5*Y_{vuu}$</td>
<td>-</td>
</tr>
<tr>
<td>$R$</td>
<td>$(Y - m)$</td>
<td>-0.105</td>
</tr>
<tr>
<td>$R^3$</td>
<td>$(1/6)*Y_{r^3r}$</td>
<td>0</td>
</tr>
<tr>
<td>$R*V^2$</td>
<td>$0.5*Y_{r^2vv}$</td>
<td>3.23</td>
</tr>
<tr>
<td>$R*D^2$</td>
<td>$0.5*Y_{r^2dd}$</td>
<td>0</td>
</tr>
<tr>
<td>$R*DU$</td>
<td>$Y_{ruu}$</td>
<td>-</td>
</tr>
<tr>
<td>$R*DU^2$</td>
<td>$0.5*Y_{ruu}$</td>
<td>-</td>
</tr>
<tr>
<td>$D$</td>
<td>$Y_d$</td>
<td>0.0586</td>
</tr>
<tr>
<td>$D^3$</td>
<td>$(1/6)*Y_{d^3d}$</td>
<td>-0.00975</td>
</tr>
<tr>
<td>$D*V^2$</td>
<td>$0.5*Y_{d^2vv}$</td>
<td>0.25</td>
</tr>
<tr>
<td>$D*R^2$</td>
<td>$0.5*Y_{d^2rr}$</td>
<td>0</td>
</tr>
<tr>
<td>$D*DU$</td>
<td>$Y_{duu}$</td>
<td>0</td>
</tr>
<tr>
<td>$D*DU^2$</td>
<td>$0.5*Y_{duu}$</td>
<td>-</td>
</tr>
<tr>
<td>$Y^2R*D$</td>
<td>$Y_{vrd}$</td>
<td>0</td>
</tr>
<tr>
<td>$-Y^2$</td>
<td>$Y_{uu}$</td>
<td>-0.0008</td>
</tr>
<tr>
<td>$DU$</td>
<td>$Y_{du}$</td>
<td>0</td>
</tr>
<tr>
<td>$DU^2$</td>
<td>$Y_{duu}$</td>
<td>-</td>
</tr>
</tbody>
</table>
TABLE I
Assessment of the Coefficients\(^1\) in the X_Equation

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Value of Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \dot{U} )</td>
<td>(- X_{\dot{u}})</td>
<td>0.177</td>
</tr>
<tr>
<td>DU</td>
<td>(X_{u})</td>
<td>-0.0253</td>
</tr>
<tr>
<td>DU(^2)</td>
<td>0.5(*X_{uu})</td>
<td>0.00948</td>
</tr>
<tr>
<td>DU(^3)</td>
<td>((1/6)*X_{uuu})</td>
<td>-0.00217</td>
</tr>
<tr>
<td>V(^2)</td>
<td>0.5(*X_{vv})</td>
<td>-0.189</td>
</tr>
<tr>
<td>( R^2 )</td>
<td>0.5(<em>X_{rr} + m</em>X_{r})</td>
<td>0.00379</td>
</tr>
<tr>
<td>D(^2)</td>
<td>0.5(*X_{dd})</td>
<td>-0.02</td>
</tr>
<tr>
<td>V(^2)*DU</td>
<td>0.5(*X_{vvv})</td>
<td>-</td>
</tr>
<tr>
<td>( R^2 )*DU</td>
<td>0.5(*X_{rrv})</td>
<td>-</td>
</tr>
<tr>
<td>D(^2)*DU</td>
<td>0.5(*X_{ddv})</td>
<td>-</td>
</tr>
<tr>
<td>V*R</td>
<td>(X_{vr} + m)</td>
<td>0.168</td>
</tr>
<tr>
<td>V*D</td>
<td>(X_{vd})</td>
<td>0.0196</td>
</tr>
<tr>
<td>R*D</td>
<td>(X_{rd})</td>
<td>0</td>
</tr>
<tr>
<td>V<em>R</em>DU</td>
<td>(X_{vrv})</td>
<td>-</td>
</tr>
<tr>
<td>V<em>D</em>DU</td>
<td>(X_{vvd})</td>
<td>-</td>
</tr>
<tr>
<td>R<em>D</em>DU</td>
<td>(X_{rdv})</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>(X_{s})</td>
<td>0</td>
</tr>
</tbody>
</table>

\(^1\)All derivatives are nondimensionalized on the basis of \( \text{RHO,L,T and S.} \)
No entry in these columns means the coefficient was ignored.
These surge, sway and yaw equations can be rewritten in the form:

\[
dU/dt = g(t, U(t), V(t), R(t), D(t))
\]

\[
dV/dt = g(t, U(t), V(t), R(t), D(t)) \quad \text{(eqn 2.10)}
\]

\[
dR/dt = g(t, U(t), V(t), R(t), D(t))
\]

Where \( U(t), V(t), R(t) \) and \( D(t) \) are the instantaneous values of \( U, V, R \) and \( D \) at any time \( t \).

Equation 2.10 is a set of three first-order differential equations for which approximate numerical solutions are readily obtained on a digital computer. The key to the numerical solution is that values of \( U, V \) and \( R \) at time \( t+\Delta t \) are obtained from knowledge of the values of \( U, V, R \) and \( D \) at time \( t \) using a simple first-order expansion; that is,

\[
U(t+\Delta t) = U(t) + \Delta t \cdot U(t)
\]

\[
V(t+\Delta t) = V(t) + \Delta t \cdot V(t) \quad \text{(eqn 2.11)}
\]

\[
R(t+\Delta t) = R(t) + \Delta t \cdot R(t)
\]

This method is found to give adequate accuracy for the present type of differential equations because of the fact that the accelerations \( \ddot{U}, \ddot{V} \) and \( \ddot{R} \) vary but slowly with time, due to the large mass and inertia of a ship compared to the relatively small forces and moments produced by its control surface. Any desired accuracy of the solutions can be obtained with a computer by simply using smaller time intervals \( \Delta t \). This procedure was used for all computer programs which were developed for this thesis.
$V = D = 0.0$ are identified as $Y^0$ and $N^0$, these are likely to be speed dependent. To see the rudder effects in calm water or sea state propeller effects were ignored.

Finally, from the $X$, $Y$ and $N$ equations the ship's surge, sway and yaw equations can be written as follows.

\[ f(U,V,R,D) \]
\[ \ddot{U} = \frac{(m - X_u)}{(m - X_u)} \]  
\[ \text{(eqn 2.7)} \]

\[ (I_z - N_k) f_1(U,V,R,D) - (mX_G - Y_k) f_2(U,V,R,D) \]
\[ \ddot{V} = \frac{(m - Y_v)(I_z - N_k) - (mX_G - Y_k)}{(m - Y_v)(I_z - N_k) - (mX_G - Y_k)} \]  
\[ \text{(eqn 2.8)} \]

\[ (m - Y_v)f_1(U,V,R,D) - (mX_G - N_y) f_2(U,V,R,D) \]
\[ \ddot{R} = \frac{(m - Y_v)(I_z - N_R) - (mX_G - Y_R)}{(m - Y_v)(I_z - N_R) - (mX_G - Y_R)} \]  
\[ \text{(eqn 2.9)} \]
Where:

\[
\begin{align*}
X_{uu} &= d^2 X / dU^2 & X_{dd} &= d^2 X / dD^2 & X_{uv} &= d^2 X / dU \times dV \\
X_{uu} &= d^3 X / dU^3 & X_{uR} &= d^3 X / dU^2 \times dR & \text{etc.}
\end{align*}
\]

\[
f_1(U, V, R, D) = 0.5 \times X_{uu} \times DU^2 + (1/6) \times X_{uu} \times DU^3 \quad \text{(eqn 2.4)}
\]
\[
+ 0.5 \times X_{vv} \times V^2 + (0.5 \times X_{RR} \times m \times X_6) \times R^2
+ 0.5 \times X_{VR} \times V^2 \times DU + 0.5 \times X_{RR} \times R^2 \times DU
+ (X_{VR} \times m) \times V \times R + X_{Vo} \times V \times D + X_{VR} \times V \times R \times DU
+ X_{VD} \times V \times D \times DU + X_{RD} \times R \times D \times DU + X \times DU + X_0
+ 0.5 \times X_{DD} \times D^2 + 0.5 \times X_{UU} \times D^2 \times DU + X_{RD} \times R \times D
\]

\[
f_2(U, V, R, D) = Y_0 \times DU + Y_{uu} \times DU^2 + Y_{V} \times V + Y_{0} \times D \quad \text{(eqn 2.5)}
\]
\[
+ Y_0 + 0.5 \times Y_{VR} \times V \times R^2 + 0.5 \times Y_{DD} \times V \times D^2
+ Y_{uu} \times V \times DU + 0.5 \times Y_{uu} \times V \times DU^2 + (Y_{R} - m \times U_1) \times R
+ (1/6) \times Y_{DDD} \times D^3 + Y_{RU} \times R \times DU + 0.5 \times Y_{RUU} \times R \times DU^2
+ (1/6) \times Y_{RRR} \times R^3 + 0.5 \times Y_{DDR} \times D \times V^2 + 0.5 \times Y_{VV} \times R \times V^2
+ 0.5 \times Y_{D} \times D \times DU + 0.5 \times Y_{UU} \times D \times DU^2
+ Y_{V} \times V \times R \times D^2 + (1/6) \times Y_{VVV} \times V^3 + 0.5 \times Y_{RDD} \times R \times D^2
\]

\[
f_3(U, V, R, D) = N_0 \times DU + N_{uu} \times DU^2 + N_{V} \times V + N_{0} \times D \quad \text{(eqn 2.6)}
\]
\[
+ 0.5 \times N_{DD} \times D \times DU + (N_{R} - m \times X_0 \times U_1) \times R
+ N_{uu} \times V \times DU + 0.5 \times N_{uu} \times V \times DU^2 + N_{VR} \times V \times R \times D + N_0
+ 0.5 \times N_{VV} \times D \times V^2 + 0.5 \times N_{RR} \times R \times D^2 + N_{UU} \times D \times DU
+ N_{RU} \times R \times DU + 0.5 \times N_{RUU} \times R \times DU^2 + (1/6) \times N_{DDD} \times D^3
+(1/6) \times N_{RRR} \times R^3 + 0.5 \times N_{VVV} \times R \times V^2 + 0.5 \times N_{RDD} \times R \times D^2
+ (1/6) \times N_{VVV} \times V^3 + 0.5 \times N_{VVV} \times R \times V^2 + 0.5 \times N_{VVV} \times V \times D^2
\]

All of the derivative coefficients of the equations are evaluated on the basis of experimental data obtained from captive model tests, and given in Table I, II and III.

The Y force and N moment induced by the rotation of a single propeller or by unirotating multiple propellers at
II. NONLINEAR EQUATIONS OF MOTION

Nonlinear equations of motion are suitable for predicting tight maneuvers and also suitable for computer programming. The nonlinear equations of motion used in this work have been developed by Abkowitz [Ref. 1, 2], and Strom Tejsen [Ref. 3], based on a Taylor series expansion of forces and moments. Terms higher than third order are not included in the equations because experience has shown that accuracy is not significantly improved by their inclusion.

A result of symmetry about the xz-plane, X is an even function of V, R, D, V and R so on, the crosscoupled terms in the equations involving odd powers of V, R and D are zero, however, crosscouple terms which involve even powers of V, R and D are nonzero. In contrast to X, the expressions for Y and N are odd functions of V, R, D, V and R; that is, only the coefficients of the terms in the expansion with odd powers are nonzero; those with even powers are zero. For some reasons, X is neither an odd nor an even function of U but rather its expansion includes all powers of DU.

Equations X, Y and N are functions of U, V, R, U, V, R and D. Taylor series expansions of X, Y and N including terms up to the third order are as follows:

\[ (m - X_\theta) \dot{U} = f_1(U, V, R, D) \] (eqn 2.1)

\[ (m - Y_\psi) \dot{V} + (mX_\phi - Y_\theta) \dot{R} = f_2(U, V, R, D) \] (eqn 2.2)

\[ (mX_\phi - N_\theta) \dot{V} + (I_\theta - N_H) \dot{R} = f_3(U, V, R, D) \] (eqn 2.3)
presents the assumptions and approaches needed to find the cost function that is used by many researchers.

Ship dynamics change with operating conditions such as ship speed, encounter angle and encounter frequency. Chapter 6 presents optimal controller parameters as a function of different operating conditions.

Chapter 7 addresses an approach to an adaptive controller utilizing information which is easy to measure on ship board such as ship speed, heading error and rudder angle. This adaptive controller must be used to provide minimum added drag due to steering.

Conclusions were drawn from simulation results and are presented in Chapter 8. This chapter also recommends some future studies, which can be done as extensions of this work.
Figure 4.1  Determination of Third and Second Order Nomoto Models.
TABLE IV
The Nomoto Model Parameters for Mariner

<table>
<thead>
<tr>
<th>Third Order</th>
<th>Second Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>K = 0.189</td>
<td>K = 0.0298</td>
</tr>
<tr>
<td>Z = 18.347</td>
<td>P1 = 9.989</td>
</tr>
<tr>
<td>P1 = 7.6739</td>
<td>P2 = 116.929</td>
</tr>
</tbody>
</table>

As seen the answers obtained by function minimization agree closely with the analytic solutions.
V. DERIVATION OF A COST FUNCTION FOR THE MARINER SHIP

In recent years many researchers have studied the problem of optimizing an automatic ship steering controller for minimum fuel consumption. It is well known that additional drag is introduced by steering and that both the rudder motion and the yawing motion contribute to this added drag.

When deriving a cost function we are required to find one cost function that must be convenient for ship board use. The cost function that is commonly used in recent years is

\[ J = \int_0^T (\lambda Y + D^2) \, dt \]  

(eqn 5.1)

Where

- \( Y \) = Yaw error
- \( \lambda \) = Weighting factor

Derivation of this cost function from the surge equation has been well explained in [Ref. 5] for the SL/7 class ship by R.E. Reid.

To derive the cost function for the Mariner, Reid's approach is taken as a reference and his assumptions are used. To show how the cost function for the Mariner was derived, Reid's work is presented here step by step with derivations of the Mariner's cost function.

The Surge Equations:

\[ \left( m - X_U \right) \dot{U} = X^0 + 0.5X_vV^2 + 0.5X_DD^2 + \left( X_{VR} + m \right) V + X_p \]  

(eqn 5.2)

Where:

- \( X^0 = -0.0003 \) for SL/7
- \( X_p = \) Propeller thrust
MARINER:

\[(m - X_G) \dot{U} = 0.5X_{Vv}V^2 + (X_{VR} + m)VR + 0.5X_{DD}D^2 + (0.5X_{RR} + m/X_G)VR^2 + X_{VD}V*D + X_{u}DU + 0.5X_{\omega u}DU^2 + (1/6)X_{\omega u}\omega DU^3 \quad (eqn 5.3)\]

It is seen that there are some terms in the Mariner's surge equation which they are not included in the SL/7's surge equation. Assuming steady state situations since, U=0. The instantaneous surge relevant to steering is

\[DX = 0.5X_{Vv}V^2 + 0.5X_{DD}D^2 + (X_{VR} + m)VR \quad (eqn 5.4)\]

SL/7:

MARINER:

\[DX = 0.5X_{Vv}V^2 + 0.5X_{DD}D^2 + (X_{VR} + m)VR + (0.5X_{RR} + m/X)R^2 + X_{VD}V*D + X_{u}DU + 0.5X_{\omega u}DU^2 + (1/6)X_{\omega u}\omega DU^3 \quad (eqn 5.5)\]

From the instantaneous surge equation relevant to steering of the SL/7 Reid came up with the following cost function:

\[J = (1/2T)\int_{t_0}^{t_0} (LAMBDA''VR + ETA*V^2 + D^2) dt \quad (eqn 5.6)\]

Where \[LAMBDA'' = (m + X_{VR})/0.5X_{DD}\]

\[ETA = (0.5X_{Vv})/0.5X_{DD}\]
and he found that the value of the \((\text{ETA} V^2)\) term is very small so he neglected the \((\text{ETA} V^2)\) term, then the cost function for the SL/7 is

\[
J = 0.5 \int_0^T (\lambda V R + D^2) \, dt \quad \text{(eqn 5.7)}
\]

Using the same approach for the Mariner, the following cost function was derived.

\[
J = \frac{1}{2T} \int_0^T (A_{1*}D + A_{2*}D^2 + A_{3*}D^3 + A_{4*}V^2 - A_{5*}R^2 + D^2 + A_{7*}V R - A_{8*}V D) \, dt
\]

Where

\[
\begin{align*}
A_{1*} &= X_u / 0.5 X_{dd} \quad A_{2*} = 0.5 X_w / 0.5 X_{dd} \\
A_{3*} &= (1/6) X_w / 0.5 X_{dd} \quad A_{4*} = 0.5 X_v / 0.5 X_{dd} \\
A_{5*} &= (0.5 X_{pp} + m X_G) / 0.5 X_{dd} \\
A_{7*} &= X_{vp} / 0.5 X_{dd} \quad A_{8*} = X_{w0} / 0.5 X_{dd}
\end{align*}
\]

\(A_{5*}\) and \(A_{8*}\) are always negative numbers, since then \(A_{5*}\) and \(A_{8*}\) have a minus sign in the equation. For the calm water case and when \(U_1=15\) knots, \(D=2.6\) degrees after 2000 seconds, values of every term in the Mariner's cost function are given below to give an idea about assumptions.

\[
\begin{align*}
A_{1*} &= -0.001939 \quad A_{2*} &= -0.00000111 \\
A_{3*} &= -0.00000000039 \quad A_{4*} &= 0.000003253 \\
A_{5*} &= -0.0002884 \quad A_{7*} &= 0.0001923 \\
D^2 &= 0.002059 \quad A_{8*} &= -0.00002609
\end{align*}
\]

As seen from the above \((A_{2*}D^2)\), \((A_{3*}D^3)\), \((A_{4*}V^2)\) and \((A_{8*}V D)\) terms are very small compared to others, so they
may be neglected. Also to measure the DU on shipboard is very hard although it may someday be measured by new satellite facilities, so we must not include it in the cost function. After these assumptions the cost function for the Mariner is

\[ J = 0.5 \int_{0}^{t} (-A5R^2 + D^2 + A7V'R') dt \]  

(eqns 5.9)

The only difference between Equation (5.9) and Equation (5.7) is the \((A5R^2)\) term that is not included in the SL/7's cost function. To see the effect of the \((A5R^2)\) term on the cost function the Mariner's cost function was evaluated with and without the \((A5R^2)\) term for the calm water case and \(U_1=15\) knots, \(D=2.6\) degree after 2000 seconds, results are:

\[
\begin{align*}
\text{with } (A5R^2) & \quad 0.0025399826296 \\
\text{without } (A5R^2) & \quad 0.002251506462 
\end{align*}
\]

There is no big difference between these two J values, and to make the derivations similar to Reid's derivations the \((A5R^2)\) term won't be included in the Mariner's cost function but as it is known that for the Mariner the \((A5R^2)\) term is as big as the \((A7V'R)\) term, it would be better to consider it in the cost function. After all of the above steps the cost function of the Mariner may be written as in Equation (5.7).
V and R are hard to measure on shipboard, but \((V \times R)\) can be defined as

\[ V \times R = OP \times YAW^2 \]

Where

\[ R = YAW = YAW \times w \]

\[ OP = \text{Distance from the ship pivot point to the origin.} = 0.3L \]

\[ w = \text{Natural frequency of the ship's steering system closed loop.} \ (w = 0.05 \text{ rad/sec was initially used.}) \]

Finally the cost function for the Mariner is

\[
J = 0.5 \int_0^t (\text{LAMBDA} \times YAW^2 + D^2) \times dt \quad \text{(eqn 5.10)}
\]

Where

\[ \text{LAMBDA} = \frac{(m + X_{VR}) \times OP \times w^2}{0.5 \times X_{DD}} \]

Since, X depends on ship speed, for different ship speeds values of LAMBDA were calculated and are presented in Table (V). These LAMBDA values were calculated by assuming the natural frequency of the ship's steering system closed loop is equal to 0.05 radian/second. How important the accuracy of the LAMBDA value is with respect to finding the optimal control parameters will be observed in Chapter 6.
### TABLE V

Values of Weighting Factor for Different Speeds of The Mariner Class Ship

<table>
<thead>
<tr>
<th>ship speed (Knots)</th>
<th>LAMBDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>6.57</td>
</tr>
<tr>
<td>15</td>
<td>2.91</td>
</tr>
<tr>
<td>20</td>
<td>1.64</td>
</tr>
</tbody>
</table>
VI. CONTROLLER DESIGN FOR THE MARINER, USING FORTRAN PROGRAM

We coupled a function minimization subroutine to the cascade compensator that is coupled with Mariner's equations of motion and used the subroutine to adjust the controller parameters to minimize the cost function which is derived in Chapter V, and evaluate the minimum cost. The Fortran program to calculate the optimal parameters is given in Appendix C. Compensator 'A' and 'B' are used as controllers, their structures are shown in Figure (6.1).

\[
\frac{K_x(S \times Z_1 + 1)}{(S \times P_1 + 1)} \quad \text{COMPENSATOR A}
\]

\[
\frac{K_x(S \times Z_1 + 1)}{(S \times P_1 + 1)(S \times P_2 + 1)} \quad \text{COMPENSATOR B}
\]

Figure 6.1 Various Structure Controllers.

A. CALM WATER CASE

For calm water, for a given 1 degree yaw command, using the computer method we optimized controllers 'A' and 'B' and the results are shown in Tables VI, VII and VIII.
### TABLE VI
Optimal Controller Parameters I

Simulation Results - Steady State 600 seconds

For ship speed 10 Knots the optimal parameters of various controllers and the cost

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.77</td>
<td>60.07</td>
<td>20.46</td>
<td>-----</td>
<td>0.0494896</td>
</tr>
<tr>
<td>B</td>
<td>0.74</td>
<td>60.07</td>
<td>20.12</td>
<td>0.902</td>
<td>0.0514841</td>
</tr>
</tbody>
</table>

### TABLE VII
Optimal Controller Parameters II

Simulation Results - Steady State 600 seconds

For ship speed 15 Knots the optimal parameters of various controllers and the cost

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.53</td>
<td>51.46</td>
<td>18.08</td>
<td>-----</td>
<td>0.0186974</td>
</tr>
<tr>
<td>B</td>
<td>0.50</td>
<td>51.58</td>
<td>17.81</td>
<td>0.890</td>
<td>0.01957901</td>
</tr>
</tbody>
</table>

### TABLE VIII
Optimal Controller Parameters III

Simulation Results - Steady State 600 seconds

For ship speed 20 Knots the optimal parameters of various controllers and the cost

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.40</td>
<td>44.87</td>
<td>16.06</td>
<td>-----</td>
<td>0.0094217</td>
</tr>
<tr>
<td>B</td>
<td>0.39</td>
<td>41.11</td>
<td>15.84</td>
<td>0.880</td>
<td>0.00991801</td>
</tr>
</tbody>
</table>
These results will be references for the controller design for sea state operation. We observe that increasing the speed gives us smaller controller parameters, also this behavior can be seen from the SL/7’s results that are presented in [Ref. 6].

B. REGULAR SEAS CASE

The ship in regular seas is affected by sea wave disturbance forces and moments. These are functions of sea state and encounter frequency. Also the added mass and the added inertia terms are functions of sea state and encounter frequency, and the encounter frequency depends on the encounter angle and ship speed. All of these variables must be considered when calculating the optimal parameters of the controller.

Using the computer method controllers 'A' and 'B' were optimized for a few different cases and the results are shown in Tables IX, X, XI, XII, XIII, XIV, XV, XVI and XVII.

TABLE IX
Optimal Controller Parameters IV

Simulation Results - Steady State 600 seconds optimal parameters of various controllers and the cost.
For ship speed 15 Knots, encounter angle 030.0 degree, encounter frequency 0.50 rad./sec. and sea state 6.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.358</td>
<td>66.6</td>
<td>24.61</td>
<td>-----</td>
<td>0.001252939</td>
</tr>
<tr>
<td>B</td>
<td>0.35</td>
<td>44.68</td>
<td>06.58</td>
<td>9.720</td>
<td>0.0010086581</td>
</tr>
</tbody>
</table>
### TABLE X

Optimal Controller Parameters V

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.

For ship speed 15 Knots, encounter angle 060.0 degree,
encounter frequency 2.50 rad./sec. and sea state 9.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.60</td>
<td>41.58</td>
<td>11.33</td>
<td>-----</td>
<td>0.000020747</td>
</tr>
<tr>
<td>B</td>
<td>0.70</td>
<td>30.49</td>
<td>03.37</td>
<td>2.940</td>
<td>0.000016038</td>
</tr>
</tbody>
</table>

### TABLE XI

Optimal controller Parameters VI

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.

For ship speed 15 Knots, encounter angle 090.0 degree,
encounter frequency 0.50 rad./sec. and sea state 7.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.54</td>
<td>30.65</td>
<td>09.35</td>
<td>-----</td>
<td>0.054223988</td>
</tr>
<tr>
<td>B</td>
<td>IT DID NOT CONVERGE</td>
<td>IT DID NOT CONVERGE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE XII

Optimal Controller Parameters VII

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.

For ship speed 15 Knots, encounter angle 120.0 degree,
encounter frequency 0.75 rad./sec. and sea state 7.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.52</td>
<td>41.09</td>
<td>08.95</td>
<td>-----</td>
<td>0.018345</td>
</tr>
<tr>
<td>B</td>
<td>IT DID NOT CONVERGE</td>
<td>IT DID NOT CONVERGE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

37
TABLE XIII
Optimal Controller Parameters VIII

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.
For ship speed 15 Knots, encounter angle 150.0 degree,
encounter frequency 1.50 rad./sec. and sea state 8.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.645</td>
<td>42.40</td>
<td>07.70</td>
<td>-----</td>
<td>0.0008188</td>
</tr>
<tr>
<td>B</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
<td>---------</td>
</tr>
</tbody>
</table>

DID NOT CONVERGE

TABLE XIV
Optimal Controller Parameters IX

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.
For ship speed 15 Knots, encounter angle 090.0 degree,
encounter frequency 0.50 rad./sec. and sea state 8.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.53</td>
<td>27.45</td>
<td>08.04</td>
<td>-----</td>
<td>0.070697939</td>
</tr>
<tr>
<td>B</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
<td>---------</td>
</tr>
</tbody>
</table>

DID NOT CONVERGE

TABLE XV
Optimal Controller Parameters X

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.
For ship speed 15 Knots, encounter angle 090.0 degree,
encounter frequency 0.50 rad./sec. and sea state 8.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.56</td>
<td>39.71</td>
<td>11.87</td>
<td>-----</td>
<td>0.019029424</td>
</tr>
<tr>
<td>B</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
<td>---------</td>
</tr>
</tbody>
</table>

DID NOT CONVERGE
TABLE XVI
Optimal Controller Parameters XI

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.
For ship speed 10 Knots, encounter angle 060.0 degree,
encounter frequency 2.50 rad./sec. and sea state 9.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.80</td>
<td>45.00</td>
<td>10.66</td>
<td>----</td>
<td>0.000045194</td>
</tr>
<tr>
<td>B</td>
<td>0.89</td>
<td>36.27</td>
<td>03.22</td>
<td>03.22</td>
<td>0.000036077</td>
</tr>
</tbody>
</table>

TABLE XVII
Optimal Controller Parameters XII

Simulation Results - Steady State 600 seconds
optimal parameters of various controllers and the cost.
For ship speed 20 Knots, encounter angle 060.0 degree,
encounter frequency 2.50 rad./sec. and sea state 9.

<table>
<thead>
<tr>
<th>CONTR</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
<th>P2</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.47</td>
<td>34.67</td>
<td>10.35</td>
<td>----</td>
<td>0.000014376</td>
</tr>
<tr>
<td>B</td>
<td>0.61</td>
<td>23.16</td>
<td>03.23</td>
<td>03.03</td>
<td>0.000011085</td>
</tr>
</tbody>
</table>

As seen from the above results, for all cases the compensators have characteristics of a lead network.

The effects of sea state on the controller parameters can be seen by comparing the Tables XV, XI and XIV. As seen from the tables an increase in sea state causes an increase in the cost value and a decrease in the controller parameters as expected, because heavy sea state brings high disturbance forces and moments and they cause heavy yawing motions. To answer this, the controller time constants
The results showed that the cost curve is flat around the minimum. The same conclusion is made in [Ref. 7 and 8] for the SL/7. Using controller parameters on the flat portion of the cost curve, but not at the optimum did not make a significant differences in simulation results, after 600 seconds of cruising there was little change in the final location of the ship. This property of the cost surface may be useful in reducing the time required to find a practical minimum for the cost function.

The ship speed effects on compensator parameters for regular seas can be seen by comparing Tables X, XVI and XVII. Observe that for both cases increasing the ship speed made the parameter values and the cost value decrease. The same observation can be made by comparing the results that are found for the SL/7 container ship for different speeds in [Ref. 9]. These results strongly indicate that the dynamics of the ship determines the optimum structure for the controller.

To see the stability of the system, the ship's third order Nomoto model was cascaded with the compensator and the open loop system BODE plot was drawn, in every case the system is stable. The phase margin varies between 40 degrees and 70 degrees and the zero cross over of the magnitude curve is around 0.04 radian/second. It was observed that small changes in the compensator parameters do not affect stability. With in large limits of parameter variation the system is always stable. For regular seas, ship speed 15 knots, encounter angle 30.0 degrees, encounter frequency 0.6 radian/second and sea state 6, using compensator 'A' and 'B', the structure of the system is presented in Figure (6.12) and the system open loop BODE plot and NICHOLS plot are shown in Figures (6.13) and (6.14).
The K values of the curves, from top to bottom is
K = 0.3
K = 0.6
K = 0.9
The optimal parameter values and respect to cost value
for this case:
K=0.60, Z1=41.58 , P1=11.33 and J=0.000020747

Figure 6.11  The Cost Curves vs. Z1 and P1 when Parameters
Changing Around the Optimal Values
for the Sea State Case.
The K values of the curves, from top to bottom is
K = 0.2
K = 0.5
K = 0.9

The optimal parameter values to cost value
K=0.53, Z1=51.46 , P1=18.08 and J=0.0186974

Figure 6.10  The Cost Curves vs. Z1 and P1 when Parameters are Changing Around the Optimal Values for the Calm Water Case.
TABLE XX
Test for LAMBDA Value II

Simulation Results - Steady State 600 seconds
for different weighting factor values, regular seas, ship speed 15 knots, encounter angle 90.0 degree, sea state 8, and encounter frequency 0.50 radian/second
optimal parameters of A type controller.

<table>
<thead>
<tr>
<th>LAMBDA</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.41</td>
<td>0.71</td>
<td>24.91</td>
<td>08.41</td>
</tr>
<tr>
<td>4.91</td>
<td>0.68</td>
<td>25.36</td>
<td>08.37</td>
</tr>
<tr>
<td>4.41</td>
<td>0.64</td>
<td>25.88</td>
<td>08.32</td>
</tr>
<tr>
<td>3.91</td>
<td>0.60</td>
<td>26.34</td>
<td>08.22</td>
</tr>
<tr>
<td>3.41</td>
<td>0.56</td>
<td>26.85</td>
<td>08.11</td>
</tr>
<tr>
<td>2.91</td>
<td>0.53</td>
<td>27.45</td>
<td>08.04</td>
</tr>
<tr>
<td>2.41</td>
<td>0.48</td>
<td>28.28</td>
<td>08.04</td>
</tr>
<tr>
<td>1.91</td>
<td>0.44</td>
<td>29.16</td>
<td>08.16</td>
</tr>
<tr>
<td>1.41</td>
<td>0.40</td>
<td>31.21</td>
<td>08.35</td>
</tr>
<tr>
<td>0.91</td>
<td>0.34</td>
<td>33.95</td>
<td>08.71</td>
</tr>
<tr>
<td>0.41</td>
<td>0.26</td>
<td>39.00</td>
<td>09.27</td>
</tr>
</tbody>
</table>

Using above compensator values did not make a significant change on the ship location after 600 seconds simulation, so it can be said that the accuracy of the weighting factor is not very important.

A few simulation runs were performed by changing the optimal compensator parameters a small amount and the cost curve was plotted. Figure (6.10) shows the cost curves for three different K values versus Z1 and P1 when the ship is in calm water, ship speed 15 knots and 1 degree course change, and Figure (6.11) shows the cost curves for ship speed 15 knots, encounter angle 60.0 degree, encounter frequency 2.5 radian/second, and sea state 9.
To see the effects of the weighting factor (LAMBDA), on the compensator parameters, different LAMBDA values were used with compensator 'A' and the compensator parameters were computed for ship speed 15 knots, calm water and regular seas with encounter angle 90.0 degree, encounter frequency 0.5 radian/second and sea state 8. Results are shown in Table (XIX) and Table (XX).

TABLE XIX
Test for LAMBDA Value I

Simulation Results - Steady State 600 seconds
for different weighting factor values, calm water, ship speed 15 knots, optimal parameters of A type controller.

<table>
<thead>
<tr>
<th>LAMBDA</th>
<th>K</th>
<th>Z1</th>
<th>P1</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.00</td>
<td>1.07</td>
<td>29.59</td>
<td>11.30</td>
</tr>
<tr>
<td>9.00</td>
<td>1.00</td>
<td>31.5</td>
<td>11.90</td>
</tr>
<tr>
<td>8.00</td>
<td>0.94</td>
<td>32.91</td>
<td>12.45</td>
</tr>
<tr>
<td>7.00</td>
<td>0.88</td>
<td>34.84</td>
<td>13.14</td>
</tr>
<tr>
<td>6.00</td>
<td>0.80</td>
<td>37.61</td>
<td>14.02</td>
</tr>
<tr>
<td>5.41</td>
<td>0.76</td>
<td>39.38</td>
<td>14.62</td>
</tr>
<tr>
<td>4.91</td>
<td>0.72</td>
<td>41.26</td>
<td>15.19</td>
</tr>
<tr>
<td>4.41</td>
<td>0.68</td>
<td>43.14</td>
<td>15.78</td>
</tr>
<tr>
<td>3.91</td>
<td>0.63</td>
<td>45.26</td>
<td>16.36</td>
</tr>
<tr>
<td>3.41</td>
<td>0.58</td>
<td>48.06</td>
<td>17.10</td>
</tr>
<tr>
<td>2.91</td>
<td>0.53</td>
<td>51.07</td>
<td>17.93</td>
</tr>
<tr>
<td>2.41</td>
<td>0.48</td>
<td>55.21</td>
<td>18.90</td>
</tr>
<tr>
<td>1.91</td>
<td>0.42</td>
<td>59.79</td>
<td>20.15</td>
</tr>
<tr>
<td>1.41</td>
<td>0.35</td>
<td>64.79</td>
<td>21.31</td>
</tr>
<tr>
<td>0.91</td>
<td>0.27</td>
<td>76.89</td>
<td>24.17</td>
</tr>
<tr>
<td>0.41</td>
<td>0.18</td>
<td>89.61</td>
<td>28.22</td>
</tr>
</tbody>
</table>
Figure 6.9  Simulation Results—Steady State 600 sec.

Curve a -------- Using compensator 'A';
Curve b -------- Using compensator 'B';
Curve c -------- The ship is in calm water
                (D=0 and YAWC=0)
Equation (6.1) was added into the fortran program and the ship's motion were observed for 600 seconds in calm water and in regular seas for 30.0 degree encounter angle, sea state 6, encounter frequency 0.60 radian/second and ship speed 15 knots. Both compensator 'A' and 'B' results are shown in Figure (6.9), and after 600 seconds ship's coordinates are given in Table XVIII.

TABLE XVIII
Location of the Ship

Simulation Results - Steady State 600 seconds

<table>
<thead>
<tr>
<th>CASE</th>
<th>Xog (ft)</th>
<th>YOG (ft)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calm water</td>
<td></td>
<td></td>
</tr>
<tr>
<td>No rudder</td>
<td>15200.994873</td>
<td>0.00</td>
</tr>
<tr>
<td>Regular seas</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comp. 'A'</td>
<td>15199.023714</td>
<td>3.2596</td>
</tr>
<tr>
<td>Regular seas</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comp. 'B'</td>
<td>15199.0763219</td>
<td>2.87116</td>
</tr>
</tbody>
</table>

For calm water and regular seas compensator 'B' gave better results than compensator 'A' did, but for some cases compensator 'B' did not converge and the difference between results is not significant. The comparisons were made as to which type of compensator brings the ship nearest to final location at the end of the 600 seconds.
The transformation from ship to space coordinates is defined as

\[
\begin{align*}
X_{og} &= U \cos(YAW) - V \sin(YAW) \\
Y_{og} &= U \sin(YAW) - V \cos(YAW)
\end{align*}
\]  
(eqn 6.1)

Where \(X_{og}, Y_{og}\) = Coordinates of the center of mass of the ship relative to coordinate system \(\text{fix}_d\) with respect to the surface of the earth

Figure 6.8 Orientation of Space Axes and Moving Axes.
Curve 'A' and 'B' are projection of main curve on JZ1 and JP1 surfaces.

Figure 6.7 The Cost vs. Z1 and P1.

As seen from the figures the optimal parameter values have a linear behavior when changing the sea state. This property would be useful to create look up tables for parameter values. The use of look up tables will be discussed in Chapter 7.

To see the motion of the ship, the ship's equations are solved in ship coordinates \((x, y)\) and then transformed to space coordinates \((x, y)\), Figure (6.8) shows the orientation of space axes and moving axes.
Figure 6.6 The Cost vs. Pl.
Figure 6.5 The Cost vs. Z1.
To have a better understanding about the effects of sea state on optimal controller parameters the cost value versus parameter values were plotted for ship speed 15 knots, encounter angle 90.0 degree, encounter frequency 0.50 radian/second and sea state 6 between 8 and they are shown in Figure (6.4), (6.5), (6.6) and (6.7).

Figure 6.4 The Cost vs. K.
It can be seen from Figure (6.2) and (6.3), for sea state 8 rudder and heading error values are bigger than the sea state 6 rudder and heading error values, but it was noticed that the periods are almost the same for both sea states.
decrease and new parameters introduce more rudder motion, so an increase in rudder motions and heading error causes increase in the cost. For ship speed 15 knots, encounter angle 90.0 degree, encounter frequency 0.5 radian/second, sea state 6 and sea state 8, rudder angles and heading errors were plotted in 200 seconds and they are shown in Figure (6.2) and (6.3).

![Figure 6.2 Rudder Angles in Degrees for Sea State 6 and 8.](image)
Where:

\[
\begin{align*}
K & = 0.34 \\
Z1 & = 75.56 \\
P1 & = 42.88 \\
P2 & = -
\end{align*}
\]

Figure 6.12 Open Loop Steering model.
Figure 6.13  Open Loop System BODE and NICHOLS Plots. (Using Comp. 'A').
Figure 6.14  Open Loop System BODE and NICHOLS Plots. (Using Comp. 'B').
Changing the environmental conditions changes the optimal controller parameters. A few simulations were run changing the environmental conditions only but the optimal controller parameters were kept unchanged. The behavior of rudder and heading error were observed. Three cases were defined depending on the environmental conditions.

Case I: Encounter angle 90 degree, encounter frequency 0.5 radian/second, sea state 6 and ship speed 15 knots. Optimal controller parameters for Case I are

\[ K = 0.56 \quad Z_1 = 39.71 \quad P_1 = 11.87 \]

Case II: Encounter angle 90 degree, encounter frequency 0.5 radian/second, sea state 8 and ship speed 15 knots. Optimal controller parameters for Case II are

\[ K = 0.53 \quad Z_1 = 27.45 \quad P_1 = 08.04 \]

Case III: Encounter angle 30 degree, encounter frequency 0.5 radian/second, sea state 6 and ship speed 15 knots. Optimal controller parameters for Case III are

\[ K = 0.358 \quad Z_1 = 66.60 \quad P_1 = 24.61 \]

Figure (6.15) and figure (6.16) show rudder and heading error when the ship is in Case I condition using Case I and Case II parameters.

Figure (6.17) and figure (6.18) show rudder and heading error when the ship is in Case II condition using Case II and Case I parameters.

Figure (6.19) and figure (6.20) show rudder and heading error when the ship is in Case III condition using Case III and Case I parameters.

Figure (6.21) and figure (6.22) show rudder and heading error when the ship is in Case I condition using Case I and Case III parameters.
Figure 6.15 Rudder Motion in Case I with Case I and Case II Parameters.
Figure 6.16  Heading Error in Case I with Case I
and Case II Parameters.

Figure 6.17  Rudder Motion in Case II with Case II
and Case I Parameters.
Figure 6.18  Heading Error in Case II with Case II and Case I Parameters.

Figure 6.19  Rudder Motion in Case III with Case III and Case I Parameters.
Figure 6.20  Heading Error in Case III with Case III and Case I Parameters.

Figure 6.21  Rudder Motion in Case I with Case I and Case III Parameters.
As seen from Figures (6.15), (6.16), (6.17) and (6.18) using Case I optimal parameters in Case II or Case II optimal parameters in Case I did not make a big difference in rudder motion and heading error, except in the transient response part.

Figures (6.19) and (6.20) show that to use Case I optimal parameters when ship is in Case III is not proper because those parameters increase rudder and heading error.

As seen from Figure (6.21), using Case III parameters in Case I provided better rudder motion after the transient response part. Cost values were calculated and it was observed that when the ship is in Case I using Case I optimum parameters it gave a better cost value than Case III parameters did, end of the 600 seconds simulation. But it
was also observed that if the final time of simulation is extended, the difference between the cost values decreases and if we continue to increase the final time, using Case III parameters in Case I gives a smaller cost value than Case I parameters do.

These results show that the transient response part is important in finding the optimum control parameters.
VII. AN APPROACH TO AN ADAPTIVE AUTOPILOT

As seen from the previous chapter, the optimal controller parameters change for changes in ship conditions such as ship speed, loading etc., and also changes in environmental conditions such as sea state, encounter angle, encounter frequency and depth of water. To maintain optimal steering performance automatically in the presence of changing conditions, it is necessary to design an adaptive system which is capable of self-adjustment of the controller parameters to provide minimum added drag due to steering.

The steering control system, if desired as an adaptive autopilot, would consist of four Subsystems as shown in Figure (7.1).

- Subsystem #1 would be a computer which will perform to find the optimal control parameters. It should get the information about ship steering characteristics from system state sensors such as a gyrocompass, rudder angle potentiometer and speed log, and it should feed the control parameter values to the controller.
- Subsystem #2 would be the controller, which should be adjustable. It gets the parameter values from subsystem #1 and sends the rudder command to subsystem #3, which steers the ship.
- Subsystem #3 is the plant which includes the system state sensors and ship steering devices such as servo motors, hydraulic pumps, rudder, steering gear etc.
- Subsystem #4 is a manual control option for safety rules. If manual steering is needed, it cuts the connection between controller and steering devices and gives the control to the helmsman. In case of computer
failure it cuts the connection between computer and controller and sends to the controller parameter values which are chosen by the watch officer. It contains look up tables which specify control parameters as functions of steering and environmental conditions.

Figure 7.1 Adaptive Control Scheme.

Success of the this adaptive autopilot system depends on the computer program as well as the accuracy of the system sensors. The computer program which was used in this research may be used on board, but the present program minimization subroutine needs a lot of computation time and it also needs starting guesses for parameters which are different for every condition. If computation time is reduced to a reasonable time and starting values are made
available in proper limits, the modified function minimization program would be considered as an adequate program for on board purposes. The work on reducing the computation time is presented in [Ref. 7].

In the future, ships could have better measurement of navigation than can be provided by conventional equipment on board. For example, the U.S Navy is involved in a program to build a system which is called NAVSTAR/GLOBAL POSITION SYSTEM (GPS). The system will provide extremely accurate three-dimensional position and velocity information to users anywhere in the world. And also another system is called NAVY REMOTE OCEAN SENSING SYSTEM (NROSS) will be able to determine wind velocities over the world's oceans with an accuracy sufficient to determine ocean surface waves. Using such valid information the watch officer can use look up tables and insert them into the computer, so system operation will be very close to the minimum cost value and the function minimization program will accomplish the fine tuning rapidly. Detailed information about GPS and NROSS can be found in [Ref. 10, 11, 12, 13].
VIII. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE STUDY

A. CONCLUSIONS

The conclusions resulting from this research of the Mariner class ship fuel consumption as it relate to steering might be listed as follows.

• A steering control system would minimize propulsion losses due to steering and maintain desired heading with reasonable heading error in every ship and environmental condition.

• The study shows that the best model to describe the dynamics of the ship is the Taylor's series expansion, which allows both linear and nonlinear terms in the ship's equation of motions. Also the third order ship Nomoto model is reasonable to use instead of the ship's equation of motions. It involves both the sway and yaw equations.

• It is believed that the cost function which is presented in Chapter 6 and is commonly used by many researchers is an adequate function for on board use. It has variables such as heading error and rudder angle which can be easily measured on board, and a weighting factor (LAMBDA) which is also easy to calculate but depends on ship conditions such as ship speed. Results in chapter 7 show that accuracy of the LAMBDA value does not make significant changes in the controller.

• In this study two different types of controller were tried which have been called controller 'A' and 'B'. The structure of these controllers is shown in Chapter 7. Controller 'A' was determined to be a best structure, because in some cases the adaptive calculations for controller 'B' did not converge.
YAW2 = YAW2 + Q2 * DELT

C COST FUNCTION
TDIFF = TDIFF + (YAW - YAW2) ** 2

PROGRAM TO CALCULATE OPTIMAL GAINS FOR CONTROLLER

// TANSAN JOB (1789.0356), 'RESEARCH', CLASS = J
// *MAIN ORG = NPCVM1.1789P
// EXEC FETCHCLGP, IMSL = DP, REGION = 1024K
// FORT.SYSIN DD #
// IN ORDER TO PERFORM SIMULATION ONLY WHEN GAINS HAVE BEEN
// OBTAINED CHANGE XS(*) TO X(*) AND DELETE XU(*) AND
// XL(*).
DIMENSION XS(3), XU(3), XL(3)
XS(1) = 0.6
XS(2) = 41.58
XS(3) = 11.33
C XS(I) IS THE STARTING GUESS
C XL(I) IS THE LOWER LIMIT FOR THE I'TH VARIABLE
C XU(I) IS THE UPPER LIMIT FOR THE I'TH VARIABLE
XL(1) = 3
XL(2) = 30.
XL(3) = 6
XU(1) = 0.9
XU(2) = 50.
XU(3) = 16.
C A DESCRIPTION OF THE FOLLOWING PARAMETERS
C IS DISCUSSED IN BOXPLX
NTA = 1000
NPR = 0
NAV = 3
NP = 3
C THE FOLLOWING STATEMENT MUST BE CHANGED TO
C CALL PLANT(XX)
C IF ONLY SIMULATION IS WANTED
CALL BOXPLX(NV, NAV, NPR, NTA, R, XS, IP, XU, XL, YMIN, IER)
WRITE (6, 25)
25 FORMAT (1X, 'OPTIMAL GAINS', /)
DO 30 I = 1, 3
30 WRITE (6, 40) I, XS(I),
40 FORMAT (I5, X, '(', I2, ')=', F14.7)
STOP
END
SUBROUTINE PLANT(XX)
SUBROUTINE PLANT(XX) SIMULATES THE SHIP
COMMON TDIFF
REAL*8 L, L2, L3, L4, L5, L6
REAL*8 X, XDOT, Y, YDOT, U, UD, V, VDOT, YAW, R, RDOT
REAL*8 TIME, ETIME, X1, X2, X3, X4, X5, X6, X7, X8
REAL*8 Y0, Y1, Y2, Y3, Y4, Y5, Y6, Y7, Y8
REAL*8 NO, N1, N2, N3, N4, N5, N6, N7, N8
REAL*8 C1, C2, C3, C4, C5, F1, F2, F3
REAL*8 Y0, DELT, S, DU, UI, K, Z1, Z2, P1, P2
REAL*8 DXAYW, YAYW, YAVG, ISR, ISER, TDIFF, LAMDA
REAL*8 S1, S2, DS1, DS2, D
REAL*8 MASS, LZ, XG, YV, YDOT, NVDOT, YR, YRDOT
REAL*8 N, NR, NRDOT, EX, FY, MZ, RXR, RYR, RXI, RYI, MZR, MZI
REAL*8 RX, RY, RZ, FX, FY, T, TL, WA, WE
DIMENSION XX(3)
C INITIAL CONDITIONS FOR INTEGRATION
C SIMULATION END TIME IN SECONDS
81
TABLE XXII
Sea State vs Wave Height

<table>
<thead>
<tr>
<th>Sea state</th>
<th>Wave height (ft)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>5.0</td>
</tr>
<tr>
<td>6</td>
<td>10.0</td>
</tr>
<tr>
<td>7</td>
<td>17.5</td>
</tr>
<tr>
<td>8</td>
<td>20.0</td>
</tr>
<tr>
<td>9</td>
<td>25.0</td>
</tr>
</tbody>
</table>

The program is set up to calculate the optimal gains for controller A. It can be modified to obtain optimal gains for the rest of the controllers. After obtaining the optimal gains the program must be modified to do a simulation. The program has sufficient comments for appropriate changes.

This program can be modified to obtain the Nomoto models. It is referenced in Chapter 4. The following need to be changed.

C GAIN COEFFICIENTS TO BE OPTIMIZED
K=XX(1)
Z1=XX(2)
P1=XX(3)
P2=XX(4)

C ERROR SIGNAL TO DRIVE RUDDER (YAW ACTUAL - YAW COMMAND)
c FOR EQUATIONS OF MOTION.
D=YAW - YAWC

C ERROR SIGNAL TO DRIVE RUDDER (YAW COMMAND - YAW ACTUAL)
C FOR NOMOTO 3RD ORDER MODEL.
D2=YAWC-YAW2
DQ1=(D2 - Q1)/P1
DQ2=((K*Z*DQ1+Q1)/P2

C INTEGRATION
Q1=Q1+DQ1*DELT
Q2=Q2+DQ2*DELT

80
• The real and imaginary value of force or moment is read from the sea state program output depending on ship speed, encounter frequency and encounter angle.
• These values are converted to magnitude and phase values.
• Using the formula below, forces and moment are created and added into the ship's equations of motion.

\[
\text{Force or Moment} = WA \times \text{MAGNITUDE} \times \cos(\text{WE} \times \text{TIME} + \text{PHASE})
\]

Where:

\[
\begin{align*}
\text{WE} & = \text{Encounter frequency (radian/second)} \\
\text{WA} & = \text{Significant wave height (ft)}
\end{align*}
\]

The correspondence between sea state and significant wave height is indicated in Table XXI [Ref. 29]. During this research the values that are presented in Table XXII were used as significant wave height (WA).

TABLE XXI
Sea State vs Range for Wave Height

<table>
<thead>
<tr>
<th>Sea State (Beaufort Scale)</th>
<th>Range for wave height (Feet)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>0.32</td>
</tr>
<tr>
<td>2</td>
<td>0.65-0.98</td>
</tr>
<tr>
<td>3</td>
<td>1.96-3.28</td>
</tr>
<tr>
<td>4</td>
<td>3.92-6.97</td>
</tr>
<tr>
<td>5</td>
<td>6.96-9.84</td>
</tr>
<tr>
<td>6</td>
<td>9.84-13.1</td>
</tr>
<tr>
<td>7</td>
<td>13.1-18.2</td>
</tr>
<tr>
<td>8</td>
<td>18.2-23.0</td>
</tr>
<tr>
<td>9</td>
<td>23.0-32.9</td>
</tr>
</tbody>
</table>
APPENDIX C
PROGRAM TO CALCULATE OPTIMAL GAINS

Algorithm used here is showed in Figure (C.1).

Figure C.1 Algorithm to Calculate Optimal Gains.

The disturbance forces (FX,FY) and moment (MZ) for regular seas were applied in program as a cosine wave. The procedure to do this is:
<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>13</td>
<td>32.174</td>
<td>1.98918</td>
<td>-23.9</td>
<td>0.0</td>
<td>8.1</td>
</tr>
<tr>
<td>1.5</td>
<td>0.2</td>
<td>0.4</td>
<td>0.5</td>
<td>0.6</td>
<td>0.75</td>
<td>1.0</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>0.0</td>
<td>1.2</td>
<td>0.0</td>
<td>15.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>30.0</td>
<td>60.0</td>
<td>90.0</td>
<td>120.0</td>
<td>150.0</td>
<td>180.0</td>
</tr>
<tr>
<td>10.0</td>
<td>15.0</td>
<td>40.0</td>
<td>56.0</td>
<td>67.0</td>
<td>89.0</td>
<td>110.0</td>
</tr>
<tr>
<td>563.6</td>
<td>76.0</td>
<td>32.0</td>
<td>520.0</td>
<td>22.62</td>
<td>126.68</td>
<td>126.68</td>
</tr>
<tr>
<td>-260.00</td>
<td>-234.00</td>
<td>-208.00</td>
<td>-156.00</td>
<td>-104.00</td>
<td>-52.00</td>
<td>-104.00</td>
</tr>
<tr>
<td>104.00</td>
<td>156.00</td>
<td>208.00</td>
<td>234.00</td>
<td>260.00</td>
<td>30090.0</td>
<td>572.0</td>
</tr>
<tr>
<td>21093.0</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>17.84</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>18.08</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>25.71</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>6</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>4.43</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>15.03</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>19.28</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>26.187</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>31.77</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>35.61</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>13.54</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>27.125</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>31.23</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>35.677</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>37.437</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
<tr>
<td>37.979</td>
<td>0.0</td>
<td>24.0</td>
<td>32.0</td>
<td>4.0</td>
<td>8.0</td>
<td>16.0</td>
</tr>
</tbody>
</table>
APPENDIX B
DATA FOR SEA STATE PROGRAM

The sea state program which is explained in [Ref. 4], needs information about the ship to calculate the ship's added mass, added inertia and seaway disturbance forces and moments. Information about the Mariner was gathered from [Ref. 27, 28] and presented here in the form which the sea state program needs. The current line is drawn on next page to show the location of the values in the format.
TDIFF = ISE + ISR
GO TO 200
400 CONTINUE
STOP
ENTRY
END
END
C COMMON COEFFICIENTS
C C1 = (0.177)*R0*L3
C C2 = (0.327)*R0*L3
C C3 = (-0.0018)*R0*L4
C C4 = (0.0175)*R0*L5
C C5 = (-0.009478)*R0*L4
C C6 = (0.3277)*R0*L4
C C7 = (-0.0018)*R0*L4
C C8 = (0.0175)*R0*L5

C REGULAR WAVES
FX = WA*RX*DCOS(WE*TIME+TX)
FY = WA*RY*DCOS(WE*TIME+TY)
MZ = WA*RZ*DCOS(WE*TIME+TZ)

C EQUATIONS OF MOTION
F1 = X1*DV + X2*DU*DU + X3*DU*DU*DU + X4*V*V
1 = Y0 + Y1*V + Y2*V*V*V + Y3*V*D*D + Y4*R
F2 = Y0 + Y1*V + Y2*V*V*V + Y3*V*D*D + Y4*R
1 = N0 + N1*V + N2*V*V*V + N3*V*D*D + N4*R
1 = N5*V*V*V + N6*D + N7*D*D*D + N8*D*V*V + MZ

C WHEN TO PRINTOUT
IF (TIME.EQ.0.0) GO TO 50
IF (ICOUNT.EQ.4) GO TO 50
GO TO 300

C CONVERT RADIAN TO DEGREES
50 YAWDEG = YAW - 57.296
RDEG = R*57.296
RDDEG = RDOT*57.296
YAWC = YAWC - 157.296
WRITE(100,100) TIME, U, V, R
100 FORMAT(6,1X,F9.2,1X,F9.6,1X,F9.6,1X,F9.6,1X,F9.6,1X,F9.6)
ICOUNT = 1
C TEST IF WANT TO STOP
300 IF (TIME.GT.ETIME) GO TO 400
C INTEGRATION STEP SIZE DELT
DELT = 1
C INTEGRATION
U = U + UDOT*DELT
V = V + VDOT*DELT
R = RDOT*DELT
YAW = YAW + RDOT*DELT
S1 = S1 + DS1*DELT
S2 = S2 + DS2*DELT

C**************************************************
TIME = TIME + DELT
ICOUNT = ICOUNT + 1
ISE = ISE + LAMDA*YAWE**2
ISR = ISR + D**2
FX=0.133
FY=0.133
MZ=0.
RXR=-.37126D5
RXI= .68406D5
RYR=-.39983D6
RYI= .2447D6
MZR=.296D8
MZI=-.1746D8
RX=(RXR**2+RXI**2)**.5
RY=(RYR**2+RYI**2)**.5
RZ=(MZR**2+MZI**2)**.5
TX=DATAN2(RXI,RXR)
TY=DATAN2(RYI,RYI)
TZ=DATAN2(MZI,MZR)

C SIGNIFICANT WAVE HEIGHT: (SEA STATE 2)
WA=25.

C ENCOUNTER FREQUENCY: (WHEN ENCOUNTER ANGLE IS 00)
WE=.75

C ADDED MASS AND ADDED INERTIA TERMS:
MASS=1.6685D+07
I2=2.3567D+11
XG=-23.9
YR=-.95066D8
YRDOT= .12211D8
NR=-.13152D11
NRDOT= .72177D10
YVDOT=-.72459D6
NVDOT= .53846D8

C HYDRODYNAMIC COEFFICIENTS ARE INSERTED HERE AS PARAMETERS
RO=1.9876*.5
YAVE=0.0
DS1=0.0
DS2=0.0
S1=0.0
S2=0.0

200 CONTINUE
C INPUT YAW COMMAND
YAWC=0.0
C IF (TIME.GE.0.0) YAWC=(1.0/57.296)
C ERROR SIGNAL TO DRIVE RUDDER (YAW ACTUAL - YAW ORDERED)
YAW=YAW-YAWC
S=(U**2)+(V**2)**.5
DU=U/1
DS1=(YAVE-S1)/P1
D=(S1+DS1*Z1)*K

AXIAL FORCE HYDRODYNAMIC COEFFICIENTS (SURGE)
X1=(-.0.0253)*(RO*L2**S)
X2=(.000948)*(RO*L2)
X3=(-.00217)*(RO*L2/S)
X4=(-.189)*(RO*L2)
X5=(.00379)*(RO*L4)
X6=(.002)*(RO*L2**S)
X7=(.188)*(RO*L3)
X8=(.0196)*(RO*L2**S)

C LATERAL FORCE HYDRODYNAMIC COEFFICIENTS (SWAY)
Y0= (-.00008)*(RO*L2**S**S)
Y0=0.0
Y1= (-.244)*(RO*L2**S)
Y2= (-1.702)*(RO*L2/S)
Y3= (-.00008)*(RO*L4**S**S)
Y4= (-.105)*(RO*L3**S)
Y5=(YR-MASS)
Y6=(.23)*(RO*L3/S)
Y7=(-.0586)*(RO*L2**S**S)
Y8=(-.00475)*(RO*L2**S**S)
Y9=(.25)*(RO*L2)

C MOMENT ABOUT Z-AXIS HYDRODYNAMIC COEFFICIENTS (YAW)
APPENDIX A
THESIS FORTRAN

$JOB
REAL*8 L, L2, L3, L4, L5, L6
REAL*8 X, XDOT, Y, YDOT, U, UDOT, V, VDOT, YAW, R, RDOT
REAL*8 TIME, ETIME, X1, X2, X3, X4, X5, X6, X7, X8
REAL*8 Y0, Y1, Y2, Y3, Y4, Y5, Y6, Y7, Y8
REAL*8 NO, N1, N2, N3, N4, N5, N6, N7, N8
REAL*8 C1, C2, C3, C4, C5, F1, F2, F3
REAL*8 R, DELT & DU, U1, K, P1, P2, P3
REAL*8 DYAWE, YAWE, YAWE, ISE, ISR, ISR, TDIFF, LAMDA
REAL*8 S1, S2, S3, S4, S5, S6, S7, S8
C INITIAL CONDITIONS FOR INTEGRATION
C SIMULATION END TIME IN SECONDS
ETIME=210.
TIME=0.0
ICOUNT=1
C INITIALIZE THE COST FUNCTION
ISE=0.0
ISR=0.0
TDIFF=0.0
LAMDA=2.91
C GAIN COEFFICIENTS TO BE OPTIMIZED
K=0.5
Z1=43.32
P1=9.26
C X, XDOT, Y, YDOT ARE FIX COORDINATES ON EARTH
X=0.0
Y=0.0
XDOT=0.0
YDOT=0.0
C U, UDOT, V, VDOT ARE FIX COORDINATES ON SHIP
F1=0.0
F2=0.0
F3=0.0
V=0.0
UDOT=0.0
VDOT=0.0
YAW=0.0
YAME=0.0
R=0.0
RDOT=0.0
C ORDERED SPEED IN FEET/SEC
C 15.3 x 1.689 FT/SEC = 15 KNOTS
U1=15.3 x 1.689
C AT STEADY STATE ACTUAL SPEED (U) = COMMAND SPEED (UC)
U=U1
C D = Rudder Angle
D=0.0 / 57.296
L=520
L2=L x 2
L3=L x L
L4=L x 4
L5=L x 5
C FORCES IN X, Y DIRECTION COMPUTED IN FORCES
C MOMENTS IN Z
The transient response has a big affect on finding the optimum parameter values. For future studies it may be better to increase final time so that the effect of the transient response would not be significant.

Recent studies on roll stabilization shows that using rudder stabilization is successful in reducing roll, [Ref. 14, 15]. Adding the roll equation into the ship model and determining a proper cost function for minimum roll motion a controller could be designed for roll stabilization.
An adaptive controller which minimizes propulsion losses due to steering is needed when ship characteristics and environmental conditions change.

For performance in fuel saving, an adaptive controller may be better than the existing Universal Gyropilot (UGP).

Since, equations of motion of surface ships differ only in the numerical values of the coefficients, the simulation programs used for the Mariner class ship would be useable for other type of ships, knowing their hull characteristics. The studies made for the SL/7 containership are examples. [Ref. 6, 9, 7, 8].

B. RECOMMENDATIONS FOR FUTURE STUDY

This research does not cover all ship and environmental conditions, so to get a better understanding about optimal controller parameters, it is recommended that the methods be applied to find the controller parameters for an expanded range of operating conditions.

This thesis investigated only course keeping with emphasis on minimizing rudder and yawing activity to reduce fuel consumption. If a track following control or an automatic control for replenishment at sea is desired, a different cost function might be needed. The nature of the cost function for such applications should be studied.

Irregular seas case were not considered in this study. For future work it is necessary to study the effect of irregular seas on the controller parameters and on the cost function. It is believed that comparing the regular sea results with irregular sea results would give better understanding about ship's steering characteristics.
ETIME=600.
TIME=0.0
ICOUNT=1
C INITIALIZE THE COST FUNCTION
ISE=0.0
ISR=0.0
TDIFF=0.0
LAMDA=2.91
C GAIN COEFFICIENTS TO BE OPTIMIZED
K=XX(1)
Z1=XX(2)
P1=XX(3)
C X,XDOT,Y,YDOT ARE FIX COORDINATES ON EARTH
X=0.0
Y=0.0
XDOT=0.0
YDOT=0.0
C U,UDOT,V,VDOT ARE FIX COORDINATES ON SHIP
P1=0.0
P2=0.0
P3=0.0
V=0.0
UDOT=0.0
VDOT=0.0
YAW=0.0
YAVE=0.0
R=0.0
RDOT=0.0
C ORDERED SPEED IN FEET/SEC
15.*1.689 FT/SEC=15 KNOTS
U1=15.*1.689
C AT STEADY STATE ACTUAL SPEED (U) = COMMAND SPEED (UC)
U=U1
C D = RUDDER ANGLE
D=0.0/57.296
L=520.
L2=L*L2
L3=L*L3
L4=L*L4
L5=L*L5
L6=L*L6
C FORCES IN X,Y DIRECTION COMPUTED IN FORCES
FX=0.
FY=0.
MX=0.
MY=0.
RX=33374D3
RY=16064D3
RXR=-17254D5
RYR=1435D4
MZR=30976D7
M2=84572D6
RX=(RXR**2+RXI**2)**.5
RY=(RYR**2+RYI**2)**.5
RZ=(MZR**2+MZI**2)**.5
TX=DATAN2(RXI.RXR)
TY=DATAN2(RYI.RYR)
TZ=DATAN2(MZI.MZR)
C SIGNIFICANT WAVE HEIGHT:(SEA STATE 2)
WA=25.
C ENCOUNTER FREQUENCY:(WHEN ENCOUNTER ANGLE IS 00)
WE=2.5
C ADDED MASS AND ADDED INERTIA TERMS:
MASS=.14685D+07
IZ=.23567D+11
XG=.239
YG=.24965D8
YRDOT=-.75199D7
NR=-.47281D10
NRDOT = -2.233D11
YVDOT = -1.9723D6
NVDOT = -2.3146D8

C HYDRODYNAMIC COEFFICIENTS ARE INSERTED HERE AS PARAMETERS
RO = 1.9876*.5
YAVE = 0.0
DS1 = 0.0
DS2 = 0.0
S1 = 0.0
S2 = 0.0

200 CONTINUE
C INPUT YAW COMMAND
YAWE = 0.0
C IF (TIME .GE. 0.0) YW = (1.0/57.296)
C ERROR SIGNAL TO DRIVE RUDDER (YAW ACTUAL - YAW ORDERED)
YAWE = YAW - YAWC
S = (U*U) + (V*V) ** 0.5
DU = 0 - U1
DS1 = (YAWE - S1)/D1
D = (S1 + DS1 * S1)**K
C AXIAL FORCE HYDRODYNAMIC COEFFICIENTS (SURGE)
X1 = (-0.0253) * (RO*L2*S)
X2 = (0.00948) * (RO*L2)
X3 = (-0.00717) * (RO*L2/2)
X4 = (0.00379) * (RO*L4)
X5 = (-0.02) * (RO*L2*S*S)
X6 = (0.168) * (RO*L3)
X7 = (-0.0196) * (RO*L2*S)
C LATERAL FORCE HYDRODYNAMIC COEFFICIENTS (SWAY)
Y1 = (-0.0008) * (RO*L2*S*S)
Y2 = 0.0
Y3 = (-0.0008) * (RO*L4*S*S)
Y4 = -0.105 * (RO*L3*S)
Y5 = YR-MASS
Y6 = (0.0586) * (RO*L2*S*S)
Y7 = (0.00975) * (RO*L2*S*S)
Y8 = (0.25) * (RO*L2)
C MOMENT ABOUT 2-AXIS HYDRODYNAMIC COEFFICIENTS (YAW)
NO = (0.00059) * (RO*L3*S*S)
N0 = 0.0
N1 = (-0.0555) * (RO*L3*S)
N2 = (0.345) * (RO*L3*S)
N3 = (0.00264) * (RO*L3*S)
N4 = (NR-MASS*XG*S)
N5 = (-1.158) * (RO*L4/S)
N6 = (0.0293) * (RO*L3*S*S)
N7 = (0.00482) * (RO*L3*S*S)
N8 = (0.1032) * (RO*L3)
C COMMON COEFFICIENTS
C1 = (0.177) * (RO*L3)
C2 = (0.32) * (RO*L3)
C3 = (MASS*YVDOT)
C4 = (MASS*XG*YVDOT)
C5 = (0.0175) * (RO*L5)
C6 = (I2-NRDOT)
C7 = (0.00478) * (RO*L4)
C8 = (MASS*XG-NVDOT)
C REGULAR WAVES
FX = WA*X*DCOS(WE*TIME+TX)
FY = WA*Y*DCOS(WE*TIME+TY)
MZ = WA*RZ*DCOS(WE*TIME+TZ)
IF (TIME.EQ.0.0) FX=0.0
IF (DABS(FY).LT.0.00000001) FY=0.0
IF (DABS(M2).LT.0.00000001) M2=0.0

C EQUATIONS OF MOTION
F1 = X1*DU + X2*DU*DU + X3*DU*DU*DU + X4*V*V
    + Y5*R*R + X6*V*D + X7*V*V*R + X8*V*V*D + FX
F2 = Y0 + Y1*V + Y2*V*V*V + Y3*V*D*D + Y4*V*R
    + Y5*R*V*V + Y6*D + Y7*D*D*D + Y8*D*V*V + FY
F3 = NO + N1*V + N2*V*V*V + N3*V*D*D + N4*V*R
    + N5*R*V*V + N6*D + N7*D*D*D + N8*D*V*V + MZ

C UDOT = F1/C1
VDOT = (C4*F2-C3*F3)/(C2*C4-C5*C3)
RDOT = (C2*F3-C5*F2)/(C2*C4-C5*C3)

C WHEN TO PRINT OUT
IF (ICOUNT.EQ.21) GO TO 50
GO TO 300

C CONVERT RADIANS TO DEGREES
50 YAWDEG = YAW*57.296
RDEG = R*57.296
RDDEG = RDOT*57.296
YAWC = YAW*C57.296
C WRITE (6,100) TIME,TDIFF
C
C 100 FORMAT(1X,F10.2,1X,F20.10)
C ICOUNT = ICOUNT + 1
C TEST IF WANT TO STOP
300 IF (TIME.GT.ETIME) GO TO 400

C INTEGRATION STEP SIZE DELT
DELT=1
C INTEGRATION
U=U+UDOT*DELT
V=V+VDOT*DELT
R=R+RDOT*DELT
YAW=YAW+R*DELT
S1=S1+DS1*DELT
S2=S2+DS2*DELT

C**************************************
C TIME=TIME-DELT
ICOUNT=ICOUNT+1
ISE=ISE + LAMDA*YAW**2
ISR=ISR + D**2
GO TO 200

400 TDIFF=ISE+ISR
C WRITE(6,390) TIME,TDIFF
500 FORMAT(1X,F9.2,1X,F20.10)
RETURN
C DELETE ALL THE FOLLOWING SUBROUTINE IF SIMULATION ONLY
C AND NOT OPTIMIZATION IS WANTED

C***************************************************************
C SUBROUTINE BOXPLX (CATEGORY HO)

C PURPOSE
C BOXPLX IS A SUBROUTINE USED TO SOLVE THE PROBLEM OF
C LOCATING A MINIMUM (OR MAXIMUM) OF AN ARBITRARY OBJECTIVE
C FUNCTION SUBJECT TO ARBITRARY EXPLICIT AND/OR
C IMPLICIT CONSTRAINTS BY THE COMPLEX METHOD OF M. J. BOX.
C EXPLICIT CONSTRAINTS ARE DEFINED AS UPPER AND LOWER
C BOUNDS ON THE INDEPENDENT VARIABLES IMPLICIT CONSTRAINTS
C MAY BE ARBITRARY FUNCTION OF THE VARIABLES. TWO FUNCTION
C SUBPROGRAM TO EVALUATE THE OBJECTIVE FUNCTION AND
C IMPLICIT CONSTRAINTS, RESPECTIVELY, MUST BE SUPPLIED
C BY THE USER (SEE EXAMPLE BELOW). BOXPLX ALSO HAS THE
C OPTION TO PERFORM INTEGER PROGRAMMING, WHERE THE VALUES
C OF THE INDEPENDENT VARIABLES ARE RESTRICTED TO INTEGERS.
CALL BOXPLX (NV, NAV, NPR, NTA, R, XS, IP, XU, XL, YMN, IER)

DESCRIPTION OF PARAMETERS

NV  AN INTEGER INPUT DEFINING THE NUMBER OF INDEPENDENT VARIABLES OF THE OBJECTIVE FUNCTION TO BE MINIMIZED.
    NOTE: MAXIMUM NV + NAV IS PRESENTLY 50. MAXIMUM NV IS 25. IF THESE LIMITS MUST BE EXCEEDED, PUNCH A SOURCE
    DECK IN THE USUAL MANNER, AND CHANGE THE DIMENSION STATEMENTS.

NAV AN INTEGER INPUT DEFINING THE NUMBER OF AUXILIARY VARIABLES THE USER WISHES TO DEFINE FOR HIS OWN CONVENIENCE. TYPICALLY HE MAY WANT TO DEFINE THE VALUE OF EACH IMPLICIT CONSTRAINT FUNCTION AS AN AUXILIARY VARIABLE. IF THIS IS DONE, THE OPTIONAL OUTPUT FEATURE OF BOXPLX CAN BE USED TO OBSERVE THE VALUES OF THOSE CONSTRAINTS AS THE SOLUTION PROGRESSES. AUXILIARY VARIABLES, IF USED, SHOULD BE EVALUATED IN FUNCTION KE (DEFINED BELOW). NAV MAY BE ZERO.

NPR  INPUT INTEGER CONTROLLING THE FREQUENCY OF OUTPUT DESIRED FOR DIAGNOSTIC PURPOSES.
    IF NPR .LE. 0, NO OUTPUT WILL BE PRODUCED BY BOXPLX. OTHERWISE, THE CURRENT COMPLEX OF K = 2*NV VERTICES AND THEIR CENTROID WILL BE OUTPUT AFTER EACH NPR PERMISSIBLE TRIALS. THE NUMBER OF TOTAL TRIALS, NUMBER OF FEASIBLE TRIALS, NUMBER OF FUNCTION EVALUATIONS AND NUMBER OF IMPLICIT CONSTRAINT EVALUATIONS ARE INCLUDED IN THE OUTPUT. ADDITIONALLY, WHEN NPR .GT. 0) THE SAME INFORMATION WILL BE OUTPUT:
    1) IF THE INITIAL POINT IS NOT FEASIBLE,
    2) IF THE FIRST COMPLETE COMPLEX IS GENERATED,
    3) IF A FEASIBLE VERTEX CANNOT BE FOUND AT SOME TRIAL,
    4) IF THE OBJECTIVE VALUE OF A VERTEX CANNOT BE MADE NO-LONGER-WORST
    5) IF THE LIMIT ON TRIALS (NTA) IS REACHED AND
    6) WHEN THE OBJECTIVE FUNCTION HAS BEEN UNCHANGED FOR 2*NV TRIALS, INDICATING A LOCAL MINIMUM HAS BEEN FOUND.
    IF THE USER WISHES TO TRACE THE PROGRESS OF A SOLUTION, A CHOICE OF NPR = 25, 50 OR 100 IS RECOMMENDED.

NTA  INTEGER INPUT OF LIMIT ON THE NUMBER OF TRIALS ALLOWED IN THE CALCULATION.
    IF THE USER INPUTS NTA .LE. 0, A DEFAULT VALUE OF 2000 IS USED. WHEN THIS LIMIT IS REACHED CONTROL RETURNS TO THE CALLING PROGRAM WITH THE BEST ATTAINED OBJECTIVE FUNCTION VALUE IN YMN, AND THE BEST ATTAINED SOLUTION POINT IN XS.

R  A REAL NUMBER INPUT TO DEFINE THE FIRST RANDOM NUMBER USED IN DEVELOPING THE INITIAL COMPLEX OF 2*NV VERTICES. (0 .LT. R .LT. 1) IF R IS NOT WITHIN THESE BOUNDS, IT WILL BE REPLACED BY 1/3.

XS  INPUT REAL ARRAY DIMENSIONED AT LEAST NV+NAV.
values of the corresponding auxiliary variables.

IP INTEGER INPUT FOR OPTIONAL INTEGER PROGRAMMING.

If IP = 1, the values of the independent variables will be replaced with integer values (still stored as REAL*4).

XU A REAL ARRAY DIMENSIONED AT LEAST NV INPUTTING THE upper bound on each independent variable, (each explicit constraint). Input values are slightly altered by BOXPLX.

XL A REAL ARRAY DIMENSIONED AT LEAST NV INPUTTING THE lower bound on each independent variable, (each explicit constraint).

NOTE: FOR BOTH XU AND XL CHOOSE REASONABLE VALUES IF NONE ARE GIVEN, NOT VALUES WHICH ARE magnitudes ABOVE OR BELOW THE EXPECTED SOLUTION. Input values are slightly altered by BOXPLX.

YMN THIS OUTPUT IS THE VALUE (REAL*4) OF THE OBJECTIVE function, CORRESPONDING TO THE SOLUTION POINT OUTPUT IN XS

IER INTEGER ERROR RETURN. TO BE INTERROGATED UPON return FROM BOXPLX. IER WILL BE ONE OF THE FOLLOWING:

-1 CANNOT FIND FEASIBLE VERTEX OR FEASIBLE CENTROID AT THE START OR A RESTART (SEE 'METHOD' BELOW).
=0 FUNCTION VALUE UNCHANGED FOR N TRIALS. (WHERE N = 5 * NV + 10) THIS IS THE NORMAL RETURN PARAMETER.
=1 CANNOT DEVELOP FEASIBLE VERTEX.
=2 CANNOT DEVELOP A NO-LONGER-WORST VERTEX.
=3 LIMIT ON TRIALS REACHED. (NTA EXCEEDED)

NOTE: VALID RESULTS MAY BE RETURNED IN ANY OF THE ABOVE CASES.

EXAMPLE OF USAGE

THIS EXAMPLE MINIMIZES THE OBJECTIVE FUNCTION SHOWN IN the external function F(X). THERE ARE TWO INDEPENDENT variables X(1) & X(2) AND TWO IMPLICIT CONSTRAINT functions X(3) & X(4) WHICH ARE EVALUATED AS AUXILIARY variables (see external function KE(X)).

DIMENSION XS(4), XU(2), XL(2)

STARTING GUESS
XS(1) = 1.0
XS(2) = 0.5

UPPER LIMITS
XU(1) = 6.0
XU(2) = 6.0

LOWER LIMITS
XL(1) = 0.0
XL(2) = 0.0

R = 9.1/13.
NTA = 5000
NFR = 50
NAV = 2
NV = 2
IP = 0

CALL BOXPLX (NV, NAV, NPR, NTA, R, XS, IP, XU, XL, YMN, IER)
WRITE (6, 1) (XS(I), I = 1, 4), YMN, IER
1 FORMAT (13.E5, 1X, 'THE POINT IS LOCATED AT (XS(I) = ', 2x, E13.7, 1x, 'AND THE FUNCTION VALUE IS ', E13.4, 1x, 'IER = ', I5)

STOP
END

86
FUNCTION KE(X)
EVALUATE CONSTRAINTS. SET KE=0 IF NO IMPLICIT CONSTRAINT
is violated, or set KE=1 IF ANY IMPLICIT
constraint is violated.
DIMENSION X(4)
X1 = X(1)
X2 = X(2)
KE = 0
X(3) = X1 + 1.732051*X2
IF (X(3) LT. 0. OR. X(3) .GT. 6.) GO TO 1
X(4) = X1/1.732051 -X2
IF (X(4) .GE. 0.) RETURN
1 KE = 1
RETURN
END

FUNCTION FE(X)
DIMENSION X(4)

THIS IS THE OBJECTIVE FUNCTION,
FE= -(X(2)**3 *(9.-(X(1)-3.)**2)/(46.76538))
RETURN
END

METHOD

THE COMPLEX METHOD IS AN EXTENSION AND ADAPTATION OF
the simple method of linear programming.
STARTING WITH ANY ONE feasible point in n-dimension
A "COMPLEX" OF 2*N vertices is constructed by
SELECTING RANDOM POINTS WITHIN THE feasible
REGION. FOR THIS PURPOSE N COORDINATES ARE FIRST
RANDOMLY CHOSEN WITHIN THE SPACE BOUNDED BY EXPLICIT CON-
STRAINTS. THIS DEFINES A TRIAL INITIAL VERTEX.
it is then checked for possible violation
OF IMPLICIT CONSTRAINTS. IF one or more are violated,
THE TRIAL INITIAL VERTEX IS DISPLACED half of its
DISTANCE FROM THE CENTROID OF PREVIOUSLY SELECTED initial
VERTICES. IF NECESSARY THIS DISPLACEMENT PROCESS IS
REPEATED UNTIL THE VERTEX HAS BECOME FEASIBLE. IF THIS
fails to happen after 5*n+10 displacements,
THE SOLUTION IS ABANDONED. after each vertex is added
to THE COMPLEX THE CURRENT centroid is checked for
FEASIBILITY. IF IT IS INFEASIBLE, the last trail
VERTEX IS ABANDONED AND AN EFFORT TO GENERATE an alter-
ATIVE TRIAL VERTEX IS MADE. IF 5*N+10 VERTICES ARE
ABANDONED CONSECUTIVELY, THE SOLUTION IS TERMINATED.

IF AN INITIAL COMPLEX IS ESTABLISHED, THE BASIC
computation loop is initiated.
THESE INSTRUCTIONS FIND THE CURRENT WORST vertex, that
IS, THE VERTEX WITH THE LARGEST CORRESPONDING value for
THE OBJECTIVE FUNCTION, AND REPLACE THAT VERTEX BY
ITS OVER-REFLECTION THROUGH THE CENTROID OF ALL OTHER
vertices. (IF the vertex to be
REPLACED IS CONSIDERED AS A VECTOR IN n-space, IT
IS OVER-REFLECTION IS OPPOSITE IN DIRECTION, IN-
CREASED IN LENGTH BY THE FACTOR 1.3, AND COLLINEAR WITH
THE REPLACED VERTEX AND CENTROID OF ALL OTHER VERTICES.)

WHEN AN OVER-REFLECTION IS NOT FEASIBLE OR REMAINS
worst, it is considered not-permissible
AND IS DISPLACED HALFWAY TOWARD the centroid.
AFTER FOUR SUCH ATTEMPTS ARE MADE UNSUCCESSFULLY
EVERY FIFTH ATTEMPT IS MADE BY REFLECTING THE OFFENDING
vertex through the present best vertex instead of through the centroid. if 5n+10
vertex displacements and over-reflections occur without a successful (permissible) result. The current best vertex is taken as an initial feasible point for a restart run of the complete process. Restarting is also undertaken when 6n-10 consecutive trials have been made with no significant change in the value of the objective function. In all cases restarting is inhibited if the last restart did not produce a significant improvement in the minimum attained.

It is recommended that the user read the reference for further useful information. It should be noted that the algorithm defined there has been altered to find the constrained minimum, rather than the maximum.

Remarks

The integer programming option was added to this program as suggested in Reference (2). A mixed integer/continuous variable version of boxplx would be easy to create by declaring "ip" to be an array of nv control variables where ip (i)=1 would indicate that the i-th variable is to be confined to integer values. Each statement of the form if (ip(i).eq.1) etc. would then need to be altered to if (ip(i).eq.1) etc. where the subscript is appropriately chosen. normally, xu and xl values are altered to be an epsilon 'within' actual values declared by the user. this adjustment is not made when ip(i)=1.

Note: no non-linear programming algorithm can guarantee that the answer found is the global minimum, rather than just a local minimum. however, according to ref.2, the complex method has an advantage in that it tends to find the global minimum more frequently than many other non-linear programming algorithms.

It should be noted that the auxiliary variable feature can also be used to deal with containing equality constraints. any equality constraint implies that a given variable is not truly independent. therefore, in general, one variable involved in an equality constraint can be renumbered from the set of nv independent variables and added to the set of nav auxiliary variables. this usually involves renumbering the independent variables of the given problem

Subroutines and functions required

Subroutine 'bout' and function 'fbv' are integral parts of the boxplx package.

two functions must be supplied by the user. the first, ke(x), is used to evaluate the implicit constraints. set ke=0 at the beginning of the function above, the first constraint, x(3), must be within the range (0. le. x(3) .le. 6.) the second constraint x(4) is not within bounds. control is transferred to statement 1 and ke is set to "1" and control is returned to boxplx.
THE SECOND FUNCTION THE USER MUST PROVIDE EVALUATES THE OBJECTIVE FUNCTION. IT IS CALLED FE(X) AS SHOWN IN THE EXAMPLE ABOVE, AND FE MUST BE SET TO THE VALUE OF THE OBJECTIVE FUNCTION CORRESPONDING TO CURRENT VALUES OF THE NV INDEPENDENT VARIABLES IN ARRAY X.

REFERENCES
BOX, M. J. "A NEW METHOD OF CONSTRAINED OPTIMIZATION AND A COMPARISON WITH OTHER METHODS", COMPUTER JOURNAL, 8 APR. '65, PP. 45-52.

PROGRAMMER
R. R. HILLEARY 1/1966
REVISED FOR SYSTEM 360 4/1967
CORRECTED 1/1969
REVISED/EXTENDED BY L. NOLAN/R. HILLEARY 2/1975
CORRECTED 8/1976

SUBROUTINE BOXPLX (NV, NAV, NPR, NTZ, RZ, XS, IP, BU, BL, YM, IER)

DIMENSION V(50,50), FUN(50), SUM(25), CEN(25), XS(NV)
1, BU(NV), BL(NV)

KV = 5
EP = 1.E-6
NTA = 2000
IF (NTZ.GT.0) NTA = NTZ
R = RZ
IF (R.LE.0 .OR. R.GE.1.) R = 1./3.
NVT = NV+NAV

NT = 0  TOTAL VARS, EXPLICIT PLUS IMPLICIT
NPT = 0  CURRENT TRIAL NO.
NTFS = 0  CURRENT NO. OF PERMISSIBLE TRIALS

CHECK FEASIBILITY OF START POINT

DO 4 I=1,NV
VT = XS(I)
IF (BL(I).LE.VT) GO TO 1
II = -I
VT = BL(I)
GO TO 2
1 IF (BU(I).GE.VT) GO TO 3
II = I
VT = BU(I)
2 IF (NPR.GT.0) WRITE (6,49) II
3 V(I,1) = VT
CEN(I) = VT
IF (IP.EQ.1) GO TO 4
BL(I) = BL(I)*A(MAX1{EP, EP*ABS(BL(I))})
BU(I) = BU(I)*A(MAX1{EP, EP*ABS(BU(I))})
4 SUM(I) = VT

89
NCE = 1
NUMBER OF CONSTRAINT EVALUATIONS
I = 1
IF (KE(V(1,1)).EQ.0) GO TO 5
IF (NPR.LE.0) GO TO 12
WRITE (6,50)
GO TO 12
5 NFE = 1
NUMBER OF VERTICES (K) = 2 TIMES NO. OF VARIABLES.
K = 2*NV
NUMBER OF DISPLACEMENTS ALLOWED.
NLIM = 5*NV+10
NUMBER OF CONSECUTIVE TRIALS WITH UNCHANGED FE TO
terminate.
NCT = NLIM+NV
ALPHA = 1.3
FK = K
FKM = FK-1.
BETA = ALPHA+1.
INSURE SEED OF R.NDOM NUMBER GENERATOR IS ODD.
IQR = R*1.E7
IF (MOD(IQR,2).EQ.0) IQR=IQR+101
SET UP INITIAL VERTICES
FUN(1) = FE(V(1,1))
YMN = FUN(1)
6 FI = 1.
FUNOLD = FUN(1)
DO 15 I=2,K
FI = FI+1.
LINT = 0
7 LINT = LINT+1
END CALCULATION IF FEASIBLE CENTROID CANNOT BE FOUND.
IF (LINT.GE.NLIM) GO TO 11
DO 8 J=1,NV
RANDOM NUMBER GENERATOR (RANDU)
IQR = IQR*65539
IF (IQR.LT.0) IQR = IQR+2147483647+1
ROX = IQR
ROX = ROX*.4656613E-9
V(J,1) = BL(J)+ROX*(BU(J)-BL(J))
8 CONTINUE
IF (IP.EQ.1) V(J,1)=AINT(V(J,1)+.5)
DO 10 L=1,NLIM
NCE = NCE+1
IF (KE(V(1,1)).EQ.0) GO TO 13
DO 9 J=1,NV
VT = .5*(V(J,1)+CEN(J))
9 CONTINUE
IF (IP.EQ.1) VT = AINT(VT+.5)
V(J,1) = VT
10 CONTINUE
11 IF (NPR.LE.0) GO TO 12
WRITE (6,51)
CALL BOUT (NT,NPT,NFE,NCE,NV,NVT,V,I,FUN,CEN,I)
12 IER = -1
GO TO 48
C 13 DO 14 J=1,NV
 14 SUM(J) = SUM(J) + V(J,I)
C TRY TO ASSURE FEASIBLE CENTROID FOR STARTING.
 15 CEN(J) = SUM(J)/I
NCE = NCE+1
 16 IF (KE(CEN).EQ.0) GO TO 60
 17 SUM(J) = SUM(J) - V(J,I)
C GO TO 7
 18 NFE = NFE+1
 19 FUN(I) = FE(V(I,I))
 20 CONTINUE
C END OF LOOP SETTING OF INITIAL COMPLEX.
 21 IF (NPR.LE.0) GO TO 17
 22 CALL BOUT (NT,NPT,NFE,NCE,NV,NVT,V,K,FUN,CEN,0)
C FIND THE WORST VERTEX, THE 'J' TH.
 23 J = 1
 24 DO 16 I=2,K
 25 IF (FUN(J).GE.FUN(I)) GO TO 16
 26 J = I
 27 CONTINUE
C BASIC LOOP. ELIMINATE EACH WORST VERTEX IN TURN.
 28 IT must become NO LONGER WORST, NOT MERELY IMPROVED.
C find next-to-vertex, THE 'JN'TH ONE.
 29 JN = 1
 30 IF (J.EQ.1) JN = 2
 31 DO 18 I=1,K
 32 IF (I.EQ.J) GO TO 18
 33 IF (FUN(JN).GE.FUN(I)) GO TO 18
 34 JN = I
 35 CONTINUE
C LIMIT = NUMBER OF MOVES DURING THIS TRIAL TOWARD THE
C centroid DUE TO FUNCTION VALUE.
 36 LIMIT = 1
C COMPUTE CENTROID AND OVER REFLECT WORST VERTEX.
 37 DO 19 I=1,NV
 38 VT = V(I,J)
 39 SUM(I) = SUM(I) - VT
 40 CEN(I) = SUM(I)/FKM
 41 VT = BETA*CEN(I) - ALPHA*VT
 42 IF (IP.EQ.1) VT = AINT(VT+.5)
C INSURE THE EXPLICIT CONSTRAINTS ARE OBSERVED.
 43 VT(I,J) = AMAX1(AMINI(VT,BU(I)),BL(I))
 44 NT = NT+1
C CHECK FOR IMPLICIT CONSTRAINT VIOLATION.
 45 DO 25 N=1,NLIM
 46 NCE = NCE+1
 47 IF (KE(V(I,J)).EQ.0) GO TO 26
C EVERY 'KV'TH TIME, OVER-REFLECT THE OFFENDING VERTEX
C through the BEST VERTEX.
 48 IF (MOD(N,KV).NE.0) GO TO 22
 49 CALL FBV (K,FUN,M)
C DO 21 I=1,NV
 50 VT = BETA*V(I,M) - ALPHA*V(I,J)
IF (IP.EQ.1) VT = AINT(VT+.5)
21 V(I,J) = AMAX1(AMIN1(VT,BU(I)),BL(I))
GO TO 24

CONSTRAINT VIOLATION: MOVE NEW POINT TOWARD CENTROID.
22 DO 23 I=1,NV
   VT = .5*(CEN(I)+V(I,J))
   IF (IP.EQ.1) VT = AINT(VT+.5)
   V(I,J) = VT
23 CONTINUE

24 NT = NT+1
25 CONTINUE

IER = 1

CANNOT GET FEASIBLE VERTEX BY MOVING TOWARD CENTROID,
OR BY OVER-REFLECTING THRU THE BEST VERTEX.
IF (NPR.LE.0) GO TO 42
WRITE (6,52) NT,J
CALL BOUT (NT,NPT,NFE,NCE,NVT,V,K,FUN,CEN,J)
GO TO 42

FEASIBLE VERTEX FOUND, EVALUATE THE OBJECTIVE FUNCTION.
26 NFE = NFE+1
FUNTRY = FE(V(I,J))

TEST TO SEE IF FUNCTION VALUE HAS NOT CHANGED.
AFQ = ABS(FUNTRY-FUNOLD)
AMX = AMAX1(ABS(EF*FUNOLD),EP)

ACTIVATE THE FOLLOWING TWO STATEMENTS FOR DIAGNOSTIC
purposes only.
WRITE (6,99)
FORMAT (IX,13.6E15.7,2I5)
IF (AFQ.GT.AM) GO TO 27
NTFS = NTFS+1
IF (NTFS.LT.NCT) GO TO 28
IER = 0
IF (NPR.LE.0) GO TO 42
CALL BOUT (NT,NPT,NFE,NCE,NVT,V,K,FUN,CEN,0)
GO TO 42

27 NTFS = 0

IS THE NEW VERTEX NO LONGER WORST?
28 IF (FUNTRY.LT.FUN(JN)) GO TO 34

TRIAL VERTEX IS STILL WORST; ADJUST TOWARD CENTROID.
EVERY KV TH TIME, OVER-REFLECT THE OFFENDING VERTEX
through the BEST VERTEX.
LIMT = LIMT+1
IF (MOD(LIMT,KV).NE.0) GO TO 30
CALL FBV (K,FUN,M)

DO 29 I=1,NV
   VT = BETA*V(I,M)-ALPHA*V(I,J)
   IF (IP.EQ.1) VT = AINT(VT+.5)
29 V(I,J) = AMAX1(AMIN1(VT,BU(I)),BL(I))
GO TO 32

DO 31 I=1,NV
   VT = .5*(CEN(I)+V(I,J))
   IF (IP.EQ.1) VT = AINT(VT+.5)
   V(I,J) = VT
31 CONTINUE

92
31 CONTINUE
C 32 IF (LIMT.LT.NLIM) GO TO 33
C CANNOT MAKE THE 'J' TH VERTEX NO LONGER WORST BY
C displacing toward THE CENTROID OR BY OVER-REFLECTING THRU THE BEST VERTEX.
C IER = 2
IF (NPR .LE. 0) GO TO 42
WRITE (6,52) NT, J
CALL BOUT (NT,NPT,NFE,NCE,NV,NVT,V,K,FUN,CEN,J)
33 NT = NT+1
GO TO 20
C SUCCESS: WE HAVE A REPLACEMENT FOR VERTEX J.
C 34 FUN(J) = FUNTRY
FUN OLD = FUNTRY
NPT = NPT+1
C EVERY 100 'TH PERMISSIBLE TRIAL, RECOMPUTE CENTROID summation to AVOID CREEPING ERROR.
IF (MOD(NPT,100).NE.0) GO TO 37
C DO 36 I=1,NV
SUM(I) = 0.
C DO 35 N=1,K
SUM(I) = SUM(I)+V(I,N)
C CEN(I) = SUM(I)/FK
36 CONTINUE
C LC = 0
GO TO 39
C 37 DO 38 I=1,NV
SUM(I) = SUM(I)+V(I,J)
C LC = J
38 CONTINUE
C IF (NPR .LE. 0) GO TO 40
IF (MOD(NPT,NPR).NE.0) GO TO 40
CALL BOUT (NT,NPT,NFE,NCE,NV,NVT,V,K,FUN,CEN,LC)
C HAS THE MAX. NUMBER OF TRIALS BEEN REACHED WITHOUT
C convergence? IF NOT, GO TO NEW TRIAL.
40 IF (NT.GE.NTA) GO TO 41
C NEXT-TO-WORST VERTEX NOW BECOMES WORST.
J = JN
GO TO 17
41 IER = 3
IF (NPR.GT.0) WRITE (6,54)
C COLLECTOR POINT FOR ALL ENDINGS.
C 1) CANNOT DEVELOP FEASIBLE VERTEX. IER = 1
C 2) CANNOT DEVELOP A NO-LONGER-WORST VERTEX. IER = 2
C 3) FUNCTION VALUE UNCHANGED FOR K TRIALS. IER = 0
C 4) LIMIT ON TRIALS REACHED. IER = 3
C 5) CANNOT FIND FEASIBLE VERTEX AT START. IER = -1
42 CONTINUE
C FIND BEST VERTEX.
CALL FBV (K,FUN,M)
IF (IER.GE.3) GO TO 44
C RESTART IF THIS SOLUTION IS SIGNIFICANTLY BETTER THAN
the previous, OR IF THIS IS THE FIRST TRY.
IF (NPR.LE.0) GO TO 43
WRITE (6,55) (M, YMN, FUN(M))
43 IF (FUN(M).GT.YMN) GO TO 47
IF (ABS(FUN(M)-YMN).LE.AMAX1(EP,EP*YMN)) GO TO 47
GIVE IT ANOTHER TRY UNLESS LIMIT ON TRIALS REACHED.

44 YMN = FUN(M)
FUN(1) = FUN(M)
DO 45 I=1,NV
CM(I) = V(I,M)
SUM(I) = V(I,M)
45 V(I,1) = V(I,M)
DO 46 I=1,NVT
XS(I) = V(I,M)

IF (IER.LT.3) GO TO 6
IF (NPR.LE.0) GO TO 48
CALL BOUT (NT, NPTNFE, NV, NVT, V, K, FUN(V(1,M), -1))
WRITE (6,56) tUN(M)
48 RETURN

49 FORMAT (5OH0INDEX AND DIRECTION OF OUTLYING
variable at starti5)
50 FORMAT (5OH0IMPLICIT CONSTRAINT VIOLATED AT
start. dead end.)
51 FORMAT ('OCANNOT FIND FEASIBLE', I4, 'TH VERTEX OR
centroid at start.')
52 FORMAT (10OHAT TRIAL I4,54H CANNOT FIND FEASIBLE
vertex which is no
LONGER WORST I4, 15X, 'RESTART FROM BEST VERTEX.')
53 FORMAT (4OH0FUNCTION HAS BEEN ALMOST UNCHANGED
for i5,7h trails)
54 FORMAT (27OH0LIMIT ON TRIALS EXCEEDED.)
55 FORMAT ( 'OMIN OBJECTIVE FUNCTION IS ',E15.7)
56 FORMAT ( 'OLD MIN WAS ',E15.7)

SUBROUTINE FBV (K, FUN, M)
DIMENSION FUN(50)
M = 1
DO 1 I=2,K
IF (FUN(M).LE.FUN(I)) GO TO 1
M = I
1 CONTINUE
RETURN
END

SUBROUTINE BOUT (NT, NPTNFE, NCE, NV, NVT, V, K, FUN, V(1,M), -1)
WRITE (6,4) NT, NPTNFE, NCE
DO 1 I=1,K
WRITE (6,5) FUN(I), (V(J,I), J=1,NV)
IF (NVT.LE.NV) GO TO 1
NVP = NV+1
WRITE (6,6) (V(J,I), J=NVP,NVT)
1 CONTINUE
IF (IK.NE.0) GO TO 2
WRITE (6,7) (C(I), I=1,NV)
RETURN
2 IF (IK.GE.0) GO TO 3
WRITE (6,8) (C(I), I=1,NV)
RETURN
3 WRITE (6,9) IK,(C(I),I=1,NV)
    RETURN
C
4 FORMAT ('NO. TOTAL TRIALS = ',I5,4X,
  1'NO. feasible trails = ',I5,4X,
  2'NO. FUNCTION EVALUATIONS = ',I5,4X,
  3'NO. constraint evaluations = ',I5/
  4'0 FUNCTION VALUE 6X INDEPENDENT VARIABLES/
  5dependent OR IMPLICIT CONSTRAINTS')
5 FORMAT (1H,E18.7,2X,7E14.7/(21X,7E14.7))
6 FORMAT (21X,7E14.7)
7 FORMAT (10HCENTROID 11X,7E14.7/(21X,7E14.7))
8 FORMAT (0 BEST VERTEX 7X,7E14.7/(21X,7E14.7))
9 FORMAT ('CENTROID LESS VX',12,2, Ei4.7/(21X,7E14.7))
END
FUNCTION FE(X)
DIMENSION X(3)
COMMON TDIFF
CALL PLANT(X)
FE=TDIFF
RETURN
END
FUNCTION KE(X)
DIMENSION X(3)
KE=0
RETURN
END
//GO.SYSIN DD *
C THIS CARD SHOULD BE USED ONLY REGULAR SEA CASE
//GO.FT12FOO1 DD DISP=SHR,DSN=MSS.S2160.A341
AN AUTOMATIC CONTROL DESIGN FOR THE MARINER CLASS SHIPS 2/2
(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA T TANSAN
DEC 84

UNCLASSIFIED

END
LIST OF REFERENCES


2. Abkowitz, M.A., Stability and Motion Control of Ocean Vehicles, Massachusetts Institute of Technology National Science Foundation, Sea Grant Project GH-1 May 1969.


5. Reid, R.E., Improvement of Ship Steering Control for Merchant Ships, Phase IIa, National Maritime Research Center, Kings Point, New York, October 1978. Reid, R.E. and Moore J.W.,


INITIAL DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>No.</th>
<th>Copies</th>
<th>Distribution List</th>
</tr>
</thead>
</table>
| 1. | 2 | Library, Code 0142  
Naval Postgraduate School  
Monterey, California 93943 |
| 2. | 2 | Defense Technical Information Center  
Cameron Station  
Alexandria, Virginia 22314 |
| 3. | 5 | Professor George J. Thaler, Code 62Tr  
Department of Electrical and  
Computer Engineering,  
Naval Postgraduate School  
Monterey, California 93943 |
| 4. | 1 | Professor Alex Gerba Jr., Code 62Gz  
Department of Electrical and  
Computer Engineering,  
Naval Postgraduate School  
Monterey, California 93943 |
| 5. | 1 | Department Chairman, Code 62Rr  
Department of Electrical and  
Computer Engineering,  
Naval Postgraduate School  
Monterey, California 93943 |
| 6. | 1 | Mr. George Curry  
HENSCHEL  
9 Hoyt Drive, P.O. BOX 30  
Newburyport, Mass 01950 |
| 7. | 1 | Mr. John Carter  
HENSCHEL  
9 Hoyt Drive, P.O. BOX 30  
Newburyport, Mass 01950 |
| 8. | 1 | LT. Garcia Vicente, USN  
P.O. Box 985  
Anthony, New Mexico 88021 |
| 9. | 5 | Tayfun Tansan  
Goztepe Istasyon cad. No: 31/13  
Istanbul, Turkey |
| 10. | 1 | LT. Kyritsis-Spyromilios Pericles, H.N.  
Barantaporou 125, Papagou  
Athens, Greece |
| 11. | 1 | LT. Horianopoulos Emmanuel, H.N.  
Travlantoni 39, Patisia  
Athens, Greece |
| 12. | 1 | Dz. K. Komutanligi  
Okullar ve Kurslar Dairesi  
Bakanliklar, Ankara, Turkey |