

NAVAL POSTGRADUATE SCHOOL Monterey, California



THESIS

**COUPLED STABILITY ANALYSIS OF CLOSE
PROXIMITY SHIP TOWING**

by

Mersin Gokce

March 2002

Thesis Advisor:

Fotis A. Papoulias

Approved for public release; distribution is unlimited.

Report Documentation Page

Report Date 29 Mar 2002	Report Type N/A	Dates Covered (from... to) -
Title and Subtitle Coupled Stability Analysis of Close Proximity Ship Towing	Contract Number	
	Grant Number	
	Program Element Number	
Author(s) Gokce, Mersin	Project Number	
	Task Number	
	Work Unit Number	
Performing Organization Name(s) and Address(es) Naval Postgraduate School Monterey, California	Performing Organization Report Number	
Sponsoring/Monitoring Agency Name(s) and Address(es)	Sponsor/Monitor's Acronym(s)	
	Sponsor/Monitor's Report Number(s)	
Distribution/Availability Statement Approved for public release, distribution unlimited		
Supplementary Notes The original document contains color images.		
Abstract		
Subject Terms		
Report Classification unclassified	Classification of this page unclassified	
Classification of Abstract unclassified	Limitation of Abstract UU	
Number of Pages 99		

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 2002	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: Coupled Stability Analysis Of Close Proximity Ship Towing			5. FUNDING NUMBERS	
6. AUTHOR(S) Gokce, Mersin				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The scope of this thesis is to study the stability of two ships in close proximity towing. Unlike previous studies in the past, the lateral dynamics of both ships are included in the formulation. The equations of motion of the system consist of the sway and yaw motions of the two ships and a control law for the leading ship. An eigenvalue stability analysis of the coupled system confirms the results that are obtained through numerical simulations. It is shown that it is possible for the system to be unstable even through the classical criteria for the towing stability are satisfied. A series of parametric studies is conducted in order to analyze the sensitivity of the system for different towline lengths, tension, and control time constant.				
14. SUBJECT TERMS DIRECTIONAL STABILITY, TOWING			15. NUMBER OF PAGES 99	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited.

COUPLED STABILITY ANALYSIS OF CLOSE PROXIMITY SHIP TOWING

Mersin Gokce
Lieutenant Junior Grade, Turkish Navy
B.A., Turkish Naval Academy, 1996

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
March 2002**

Author: Mersin Gokce

Approved by: Fotis A. Papoulias, Thesis Advisor

Terry McNelley, Chairman
Department of Mechanical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The scope of this thesis is to study the stability of two ships in close proximity towing. Unlike previous studies in the past, the lateral dynamics of both ships are included in the formulation. The equations of motion of the system consist of the sway and yaw motions of the two ships and a control law for the leading ship. An eigenvalue stability analysis of the coupled system confirms the results that are obtained through numerical simulations. It is shown that it is possible for the system to be unstable even though the classical criteria for towing stability are satisfied. A series of parametric studies is conducted in order to analyze the sensitivity of the system for different towline lengths, tension, and control time constant.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
II.	PROBLEM FORMULATION.....	5
	A. EQUATIONS OF MOTION.....	5
	B. LINEARIZATION.....	7
	C. A ZERO EIGENVALUE.....	9
	D. RUDDER CONTROL.....	10
	E. SAMPLE SIMULATION RESULTS.....	11
	1. Stable Condition.....	11
	2. Unstable Condition.....	14
III.	STABILITY ANALYSIS.....	19
	A. LINEARIZATION.....	19
	B. DEGREE OF STABILITY.....	21
	C. REGIONS OF STABILITY.....	22
IV.	CONCLUSIONS AND RECOMMENDATIONS.....	25
	A. CONCLUSIONS.....	25
	B. RECOMMENDATIONS.....	26
	APPENDIX A. DEGREE OF STABILITY VS. TOWLINE LENGTH.....	27
	APPENDIX B. DEGREE OF STABILITY VS. TENSION.....	33
	APPENDIX C. DEGREE OF STABILITY VS. TIME CONSTANT.....	43
	APPENDIX D. MATLAB CODE FOR STABILITY ANALYSIS WITHOUT CONTROL.....	53
	APPENDIX E. MATLAB CODE TO PROVE ZERO EIGENVALUE.....	55
	APPENDIX F. MATLAB CODE TO FIND HYDRODYNAMIC COEFFICIENTS	57
	APPENDIX G. MATLAB CODE TO FIND THE COEFFICIENTS OF DELTA.....	59
	APPENDIX H. MATLAB CODE FOR STABILITY ANALYSIS AND SIMULATION WITH CONTROL.....	61
	APPENDIX I. MATLAB CODE TO FIND THE DEGREE OF STABILITY FOR DIFFERENT TOWLINE LENGTHS.....	67
	APPENDIX J. MATLAB CODE TO FIND THE DEGREE OF STABILITY FOR DIFFERENT TENSIONS.....	71
	APPENDIX K. MATLAB CODE TO FIND THE DEGREE OF STABILITY FOR DIFFERENT TIME CONSTANTS.....	75
	APPENDIX L. MATLAB CODE TO FIND THE REGION OF STABILITY.....	79
	APPENDIX M. MATLAB CODE TO EVALUATE THE STABILITY CRITERIA BASED ON ONLY TRAILING SHIP.....	83

LIST OF REFERENCES	85
INITIAL DISTRIBUTION LIST	87

LIST OF FIGURES

Figure 1. Profile view of SLICE vessel [From: [2]]	2
Figure 2. The profile of the SSP KAIMALINO [From: [2]]	2
Figure 3. Geometry of the Towing Ships.....	6
Figure 4. Ship Offset vs. Time	12
Figure 5. Leading Ship Rudder Angle vs. Time	12
Figure 6. Towline Angle vs. Time	13
Figure 7. Leading Ship Heading Angle vs. Time.....	13
Figure 8. Trailing Ship Heading Angle vs. Time.....	14
Figure 9. Ship Offsets vs. Time.....	15
Figure 10. Leading Ship Rudder Angle vs. Time	15
Figure 11. Towline Angle vs. Time	16
Figure 12. Leading Ship Heading Angle vs. Time.....	16
Figure 13. Trailing Ship Heading Angle vs. Time.....	17
Figure 14. The Stability Region	23

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The stability of ship towing is the primary concern in both naval architecture and maneuvering control. The main problem consists of two individual tasks, first building a mathematical model of the two towed ships and second analyzing the directional stability of the combined system.

Traditionally, ship towing in the open ocean has been performed under relatively large separation distances between the two vessels; i.e., large towlines. Only in the case of barge towing in rivers and canals, has close proximity towing been applied. Over the last few years, however, interest on ocean close proximity towing has been resurfaced. The Office of Naval Research is interested in close proximity towing for its Advanced Hull forms Program, in particular small waterplane area twin hull (SWATH) ships and their variations (such as the SLICE hull). Studies of the SEA LANCE project at the Naval Postgraduate School (Total Ship Systems Engineering Program) have also shown several benefits associated with close proximity towing. One of the main benefits of close proximity towing in military applications is the ability to separate a combatant ship from a main part of its payload.

Towing is not without its problems, however. Directional stability under tow and seakeeping are two main areas of concern. This thesis will concentrate on the problem of directional stability. Previous studies on directional stability of ship towing were performed by Abkowitz in 1964 who developed the characteristic equation for single body towing, and by Bernitsas and others who developed the criteria for stability of Abkowitz's 4th order characteristic equation. In these studies the stability criteria was based solely on the trailing ship, which means that the dynamics of the leading ship were neglected. In fact, the leading ship was approximated by a point mass moving on a straight line under constant forward speed. Our concern is that although this may be valid for long towlines, it may not be a valid approximation for close proximity towing, and the existing stability criteria may be inadequate which is the reason and the justification of this thesis.

In this study as a leading ship we will use the SLICE vessel, which is 105 ft, and 180 tons, and as a trailing ship the SWATH ship Kaimalino which is 89 ft, and 217 tons.

The Slice is a high speed variant of the Swath technology. It has 4 underwater hulls instead of two. Attached to each hull is a strut that extends up to support the main body. Figure 1 shows the profile view of the Slice.[2]

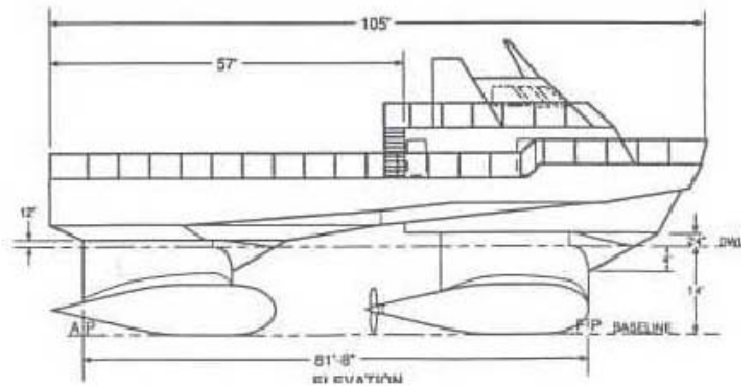


Figure 1. Profile view of SLICE vessel [From: [2]]

The SSP Kaimalino was the world's first high performance open ocean Swath ship. It consists of two parallel torpedo-like hulls. Attached to the hulls are two streamlined struts. The struts extend above the water surface and support the main body. The Kaimalino also has stabilizing fins attached near the aft end of each hull. Figure 2 shows a profile of the SSP Kaimalino. [2]

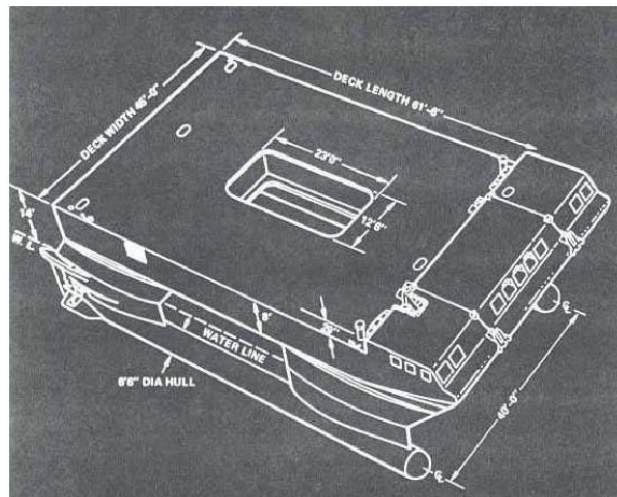


Figure 2. The profile of the SSP KAIMALINO [From: [2]]

The general approach adopted in this thesis is as follows:

1. First we will use the maneuvering equations of motion in the horizontal plane for both leading and trailing ship.
2. Coupling between the two ships will be provided through the towline. Hydrodynamic coupling arising from radiation and diffraction effects will be neglected. This can be easily incorporated into the analysis once the effect of such hydrodynamic coupling on the hydrodynamic coefficients is established.
3. The response of the coupled system will be analyzed by both simulation and an eigenvalue analysis. It is hoped that this analysis will provide regions of directional stability, and can therefore lead to a design methodology for rational towing system parameter selection based on their stability properties.

The hydrodynamic coefficients of both ships used in this study are provided from reference [2]. It should be emphasized that our procedure will be general enough so that it can accommodate different or additional hydrodynamic coefficients.

THIS PAGE INTENTIONALLY LEFT BLANK

II. PROBLEM FORMULATION

A. EQUATIONS OF MOTION

The surge, sway and yaw maneuvering equations of motion are shown below. Throughout this thesis subscript 1 is for the towing (leading) ship and subscript 2 is for the towed (trailing) ship.

$$(m_2 - X_{\dot{u}_2})\dot{u}_2 = -mv_2r_2 - R_2 + T \cos(\psi_2 + \gamma) \quad (1)$$

$$(m_2 - Y_{\dot{v}_2})\dot{v}_2 - Y_{r_2}\dot{r}_2 = -m_2r_2u_2 + Y_{v_2}v_2 + Y_{r_2}r_2 - T \sin(\psi_2 + \gamma) \quad (2)$$

$$(I_{Z_2} - N_{\dot{r}_2})\dot{r}_2 - N_{v_2}\dot{v}_2 = N_{v_2}v_2 + N_{r_2}r_2 - Tx_{p_2} \sin(\psi_2 + \gamma) \quad (3)$$

$$(m_1 - Y_{\dot{v}_1})\dot{v}_1 - Y_{r_1}r_1 = -m_1r_1u_1 + Y_{v_1}v_1 + Y_{r_1}r_1 + T \sin(\psi_1 + \gamma) \quad (4)$$

$$(I_{Z_1} - N_{\dot{r}_1})\dot{r}_1 - N_{v_1}\dot{v}_1 = N_{v_1}v_1 + N_{r_1}r_1 - Tx_{p_1} \sin(\psi_1 + \gamma) \quad (5)$$

and

$$\dot{\psi}_1 = r_1 \quad (6)$$

$$\dot{\psi}_2 = r_2 \quad (7)$$

The detailed explanation of the derivation of the equations of the motion, and the assumptions used can be found in the reference [1].

In these equations u is the surge velocity, v is sway velocity, r is the yaw velocity, R is the resistance of the vessel moving through body of water, and T is the tension in the connection (rope, cable, or some other mechanism) between the two towing ships.

The kinematic relations are as shown below;

$$\dot{y}_1 = u_1 \sin \psi_1 + v_1 \cos \psi_1 \quad (8)$$

$$\dot{y}_2 = u_2 \sin \psi_2 + v_2 \cos \psi_2 \quad (9)$$

If we define

$$\bar{y} = y_2 - y_1 \quad (10)$$

then

$$\dot{\bar{y}} = \dot{y}_2 - \dot{y}_1 \quad (11)$$

These geometric relations are explained in Figure 3.

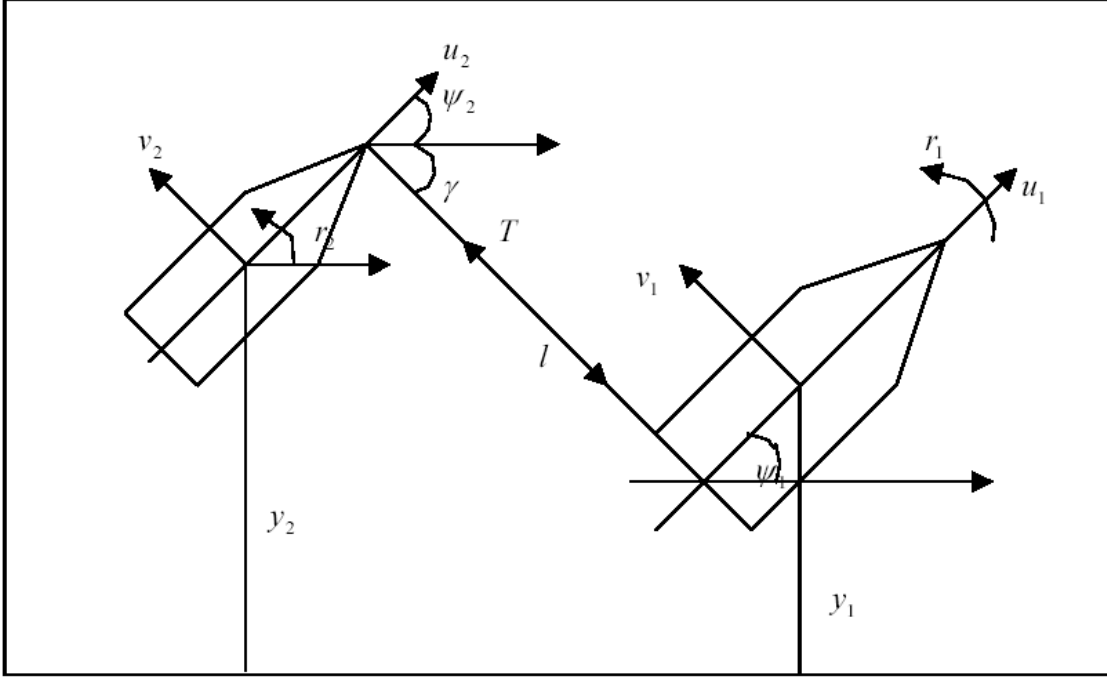


Figure 3. Geometry of the Towing Ships

Substituting equation (8) and equation (9) into equation (10), we get

$$\dot{\bar{y}} = u_2 \sin \psi_2 + v_2 \cos \psi_2 - u_1 \sin \psi_1 - v_1 \cos \psi_1 \quad (12)$$

From the geometry of the figure we can see that;

$$\sin \gamma = \frac{y_2 + x_{p2} \sin \psi_2 - (y_1 - x_{p1} \sin \psi_1)}{l} \quad (13)$$

which can be rewritten as

$$\sin \gamma = \frac{\bar{y}}{l} + \frac{1}{l} (x_{p2} \sin \psi_2 + x_{p1} \sin \psi_1) \quad (14)$$

B. LINEARIZATION

In order to analyze the stability of our system, we should first linearize the system. During linearization we assumed that the velocities of the both vessels are constant and identical which in non-dimensional terms means

$$u_1 = u_2 = 1 \quad (15)$$

Another assumption is that we are dealing with small heading angles, which gives us

$$\sin \psi_{1,2} = \psi_{1,2} \quad (16)$$

and

$$\cos \psi_{1,2} = 1 \quad (17)$$

Since we are dealing with constant surge velocity, equation (1) is dropped, and we are left with the equations (2), (3), (4), and (5).

When we substitute equations (15), (16), (17) into equations (2), (3), (4), (5), (12) and (13) we come up with

$$(m_2 - Y_{\dot{v}_2})\dot{v}_2 - Y_{\dot{r}_2}\dot{r}_2 = -m_2 r_2 + Y_{v_2} v_2 + Y_{r_2} r_2 - T(\psi_2 + \gamma) \quad (18)$$

$$(I_{Z_2} - N_{\dot{r}_2})\dot{r}_2 - N_{\dot{v}_2}\dot{v}_2 = N_{v_2} v_2 + N_{r_2} r_2 - T x_{p_2}(\psi_2 + \gamma) \quad (19)$$

$$(m_1 - Y_{\dot{v}_1})\dot{v}_1 - Y_{\dot{r}_1} r_1 = -m_1 r_1 + Y_{v_1} v_1 + Y_{r_1} r_1 + T(\psi_1 + \gamma) \quad (20)$$

$$(I_{Z_1} - N_{\dot{r}_1})\dot{r}_1 - N_{\dot{v}_1}\dot{v}_1 = N_{v_1} v_1 + N_{r_1} r_1 - T x_{p_1}(\psi_1 + \gamma) \quad (21)$$

$$\dot{\bar{y}} = \psi_2 + v_2 - \psi_1 - v_1 \quad (22)$$

$$\gamma = \frac{\bar{y}}{l} + \frac{1}{l}(x_{p_2}\psi_2 + x_{p_1}\psi_1) \quad (23)$$

The above equations can be rewritten into a standard matrix form as follows:

$$\dot{v}_2 = A_{2vv} v_2 + A_{2vr} r_2 + B_{2v}(\psi_2 + \gamma) \quad (24)$$

$$\dot{r}_2 = A_{2rv}v_2 + A_{2rr}r_2 + B_{2r}(\psi_2 + \gamma) \quad (25)$$

$$\dot{v}_1 = A_{1vv}v_1 + A_{1vr}r_1 + B_{1v}(\psi_1 + \gamma) \quad (26)$$

$$\dot{r}_1 = A_{1rv}v_1 + A_{1rr}r_1 + B_{1r}(\psi_1 + \gamma) \quad (27)$$

After the necessary mathematical steps we can get the coefficients as follows

$$A_{2vv} = [(I_{z2} - N_{\dot{r}2})Y_{v2} + (N_{v2}Y_{\dot{r}2})] \div [(m_2 - Y_{\dot{v}2})(I_{z2} - N_{\dot{r}2}) - N_{\dot{v}2}Y_{\dot{r}2}] \quad (28)$$

$$A_{2vr} = [(Y_{r2} - m_2)(I_{z2} - N_{\dot{r}2}) + (N_{r2}Y_{\dot{r}2})] \div [(m_2 - Y_{\dot{v}2})(I_{z2} - N_{\dot{r}2}) - N_{\dot{v}2}Y_{\dot{r}2}] \quad (29)$$

$$B_{2v} = [-(I_{z2} - N_{\dot{r}2})T - (Y_{\dot{r}2}Tx_{p2})] \div [(m_2 - Y_{\dot{v}2})(I_{z2} - N_{\dot{r}2}) - N_{\dot{v}2}Y_{\dot{r}2}] \quad (30)$$

$$A_{2rv} = [(Y_{v2}N_{\dot{v}2}) + N_{v2}(m_2 - Y_{\dot{v}2})] \div [(m_2 - Y_{\dot{v}2})(I_{z2} - N_{\dot{r}2}) - N_{\dot{v}2}Y_{\dot{r}2}] \quad (31)$$

$$A_{2rr} = [(Y_{r2} - m_2)N_{\dot{v}2} + N_{r2}(m_2 - Y_{\dot{v}2})] \div [(m_2 - Y_{\dot{v}2})(I_{z2} - N_{\dot{r}2}) - N_{\dot{v}2}Y_{\dot{r}2}] \quad (32)$$

$$B_{2r} = [-TN_{\dot{v}2} - Tx_{p2}(m_2 - Y_{\dot{v}2})] \div [(m_2 - Y_{\dot{v}2})(I_{z2} - N_{\dot{r}2}) - N_{\dot{v}2}Y_{\dot{r}2}] \quad (33)$$

$$A_{1vv} = [(I_{z1} - N_{\dot{r}1})Y_{v1} + (N_{v1}Y_{\dot{r}1})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (34)$$

$$A_{1vr} = [(Y_{r1} - m_1)(I_{z1} - N_{\dot{r}1}) + (N_{r1}Y_{\dot{r}1})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (35)$$

$$B_{1v} = [(I_{z1} - N_{\dot{r}1})T - (Y_{\dot{r}1}Tx_{p2})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (36)$$

$$A_{1rv} = [(Y_{v1}N_{\dot{v}1}) + N_{v1}(m_1 - Y_{\dot{v}1})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (37)$$

$$A_{1rr} = [(Y_{r1} - m_1)N_{\dot{v}1} + N_{r1}(m_1 - Y_{\dot{v}1})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (38)$$

$$B_{1r} = [TN_{\dot{v}1} + Tx_{p1}(m_1 - Y_{\dot{v}1})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (39)$$

Now we can write down the complete system in a 7x7 matrix form as

$$\begin{bmatrix} \dot{v}_2 \\ \dot{r}_2 \\ \dot{v}_1 \\ \dot{r}_1 \\ \dot{\bar{y}} \\ \dot{\psi}_2 \\ \dot{\psi}_1 \end{bmatrix} = [A] \begin{bmatrix} v_2 \\ r_2 \\ v_1 \\ r_1 \\ \bar{y} \\ \psi_2 \\ \psi_1 \end{bmatrix} \quad (40)$$

where

$$[A] = \begin{bmatrix} A_{2vv} & A_{2vr} & 0 & 0 & \frac{B_{2v}}{l} & B_{2v} + \frac{B_{2v}x_{p2}}{l} & \frac{B_{2v}x_{p1}}{l} \\ A_{2rv} & A_{2rr} & 0 & 0 & \frac{B_{2r}}{l} & B_{2r} + \frac{B_{2r}x_{p2}}{l} & \frac{B_{2r}x_{p1}}{l} \\ 0 & 0 & A_{1vv} & A_{1vr} & \frac{B_{1v}}{l} & \frac{B_{1v}x_{p2}}{l} & B_{1v} + \frac{B_{1v}x_{p1}}{l} \\ 0 & 0 & A_{1rv} & A_{1rr} & \frac{B_{1r}}{l} & \frac{B_{1r}x_{p2}}{l} & B_{1r} + \frac{B_{1r}x_{p1}}{l} \\ 1 & 0 & -1 & 0 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (41)$$

Following linearization of the system, the next step is to find the eigenvalues of the A matrix. These will determine whether the system is stable or not. The matlab program shown in Appendix D was written and it was used to find the eigenvalues of the system.

C. A ZERO EIGENVALUE

Several different values for tension and for length of the connection between the two ships were tried using the program shown in Appendix D. In every instant, a zero eigenvalue was found. Therefore, it seemed impossible for the system to reach a stable condition. Simulations also showed the same effect. In order to test whether the existence of a zero eigenvalue was just a coincidence or an inherent property of the system, the matlab program shown in Appendix E is used. This performed a symbolic manipulation utilizing the Matlab symbolic manipulation toolbox. It was found that regardless of the numerical values of the geometric properties and the hydrodynamic coefficients of the

system, the characteristic equation of this system has always a zero constant, which proved the existence of a zero eigenvalue. Physically, this zero eigenvalue can be explained by the lack of any directional stabilizing effects on the leading ship. It can be easily verified, by observing the signs of the towline restoring forces and moments on the two ships that the towing force has a stabilizing effect on the trailing ship but a destabilizing effect on the leading ship. Since ships in the horizontal plane cannot exhibit directional stability, it is necessary in order to continue with the analysis to introduce some kind of directional control on the leading ship. This is analyzed in the following section.

D. RUDDER CONTROL

It was shown in the previous section that some kind of directional control on the leading ship is necessary to proceed with the towing analysis. We do this by means of a rudder control law. The equations of motion (4), (5) are changed because of inclusion of the rudder, but equations (2) and (3) are same.

$$(m_2 - Y_{\dot{v}_2})\dot{v}_2 - Y_{\dot{r}_2}\dot{r}_2 = -m_2 r_2 u_2 + Y_{v_2} v_2 + Y_{r_2} r_2 - T \sin(\psi_2 + \gamma) \quad (2)$$

$$(I_{Z2} - N_{\dot{r}_2})\dot{r}_2 - N_{\dot{v}_2}\dot{v}_2 = N_{v_2} v_2 + N_{r_2} r_2 - T x_{p2} \sin(\psi_2 + \gamma) \quad (3)$$

$$(m_1 - Y_{\dot{v}_1})\dot{v}_1 - Y_{\dot{r}_1} r_1 = -m_1 r_1 u_1 + Y_{v_1} v_1 + Y_{r_1} r_1 + T \sin(\psi_1 + \gamma) + Y_{\delta} \delta \quad (42)$$

$$(I_{Z1} - N_{\dot{r}_1})\dot{r}_1 - N_{\dot{v}_1}\dot{v}_1 = N_{v_1} v_1 + N_{r_1} r_1 - T x_{p1} \sin(\psi_1 + \gamma) + N_{\delta} \delta \quad (43)$$

where

$$\delta = -k_{\psi} \psi_1 - k_v v_1 - k_r r_1 - k_y y_1 \quad (44)$$

The rudder control law shown in equation (44) is a typical full state feedback control law. This was chosen as a representative control law and it by no means limits the applicability of the results that follow. The procedure would be identical even if a different rudder control law were employed. The feedback gains were calculated by using

the standard pole placement scripts in Matlab's control system toolbox. In order to present the results in a compact form, we chose the control time constant as the representative selection criterion for the gains. Different choices do not alter the qualitative features of our results. The control time constant is defined as a non-dimensional response time for the system. It is measured in dimensionless seconds, with one dimensionless second being the time that it takes for a ship to travel one ship length. Typically, a system reaches steady state within three time constants. Thus, a high time constant signifies a rather slow leading ship control law, while a small time constant shows a fast (responsive) control. The closed loop poles are simply set as the negative inverse of the time constant.

E. SAMPLE SIMULATION RESULTS

We want to give one example to each stable and instable condition to clarify some of the concepts mentioned above. These results were obtained using the matlab code shown in Appendix H.

1. Stable Condition

The figures below are obtained for the following parameters;

Tension=0.005

Length=1.2

Time Constant=1

The results show a slow convergence to the nominal equilibrium state; i.e., straight line motion along the x-axis. These results will be confirmed with the stability analysis of the next chapter.

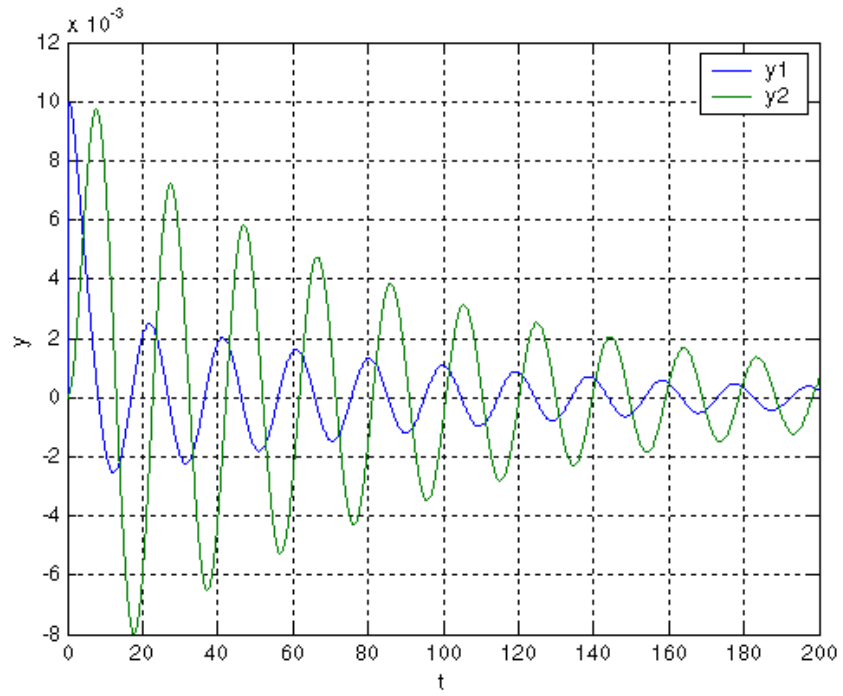


Figure 4. Ship Offset vs. Time

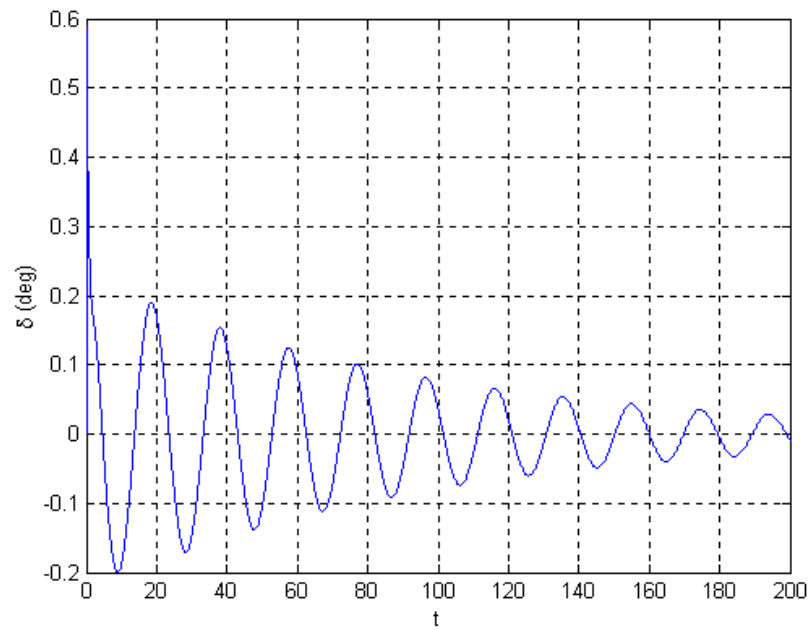


Figure 5. Leading Ship Rudder Angle vs. Time

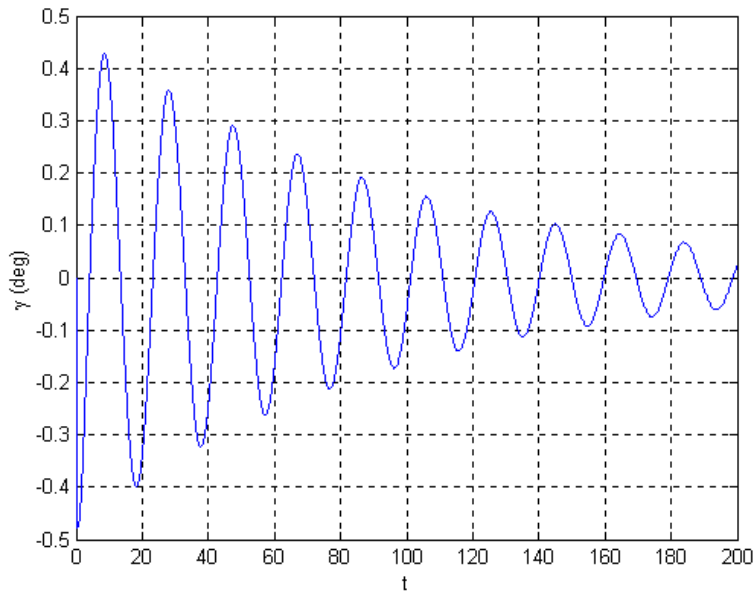


Figure 6. Towline Angle vs. Time

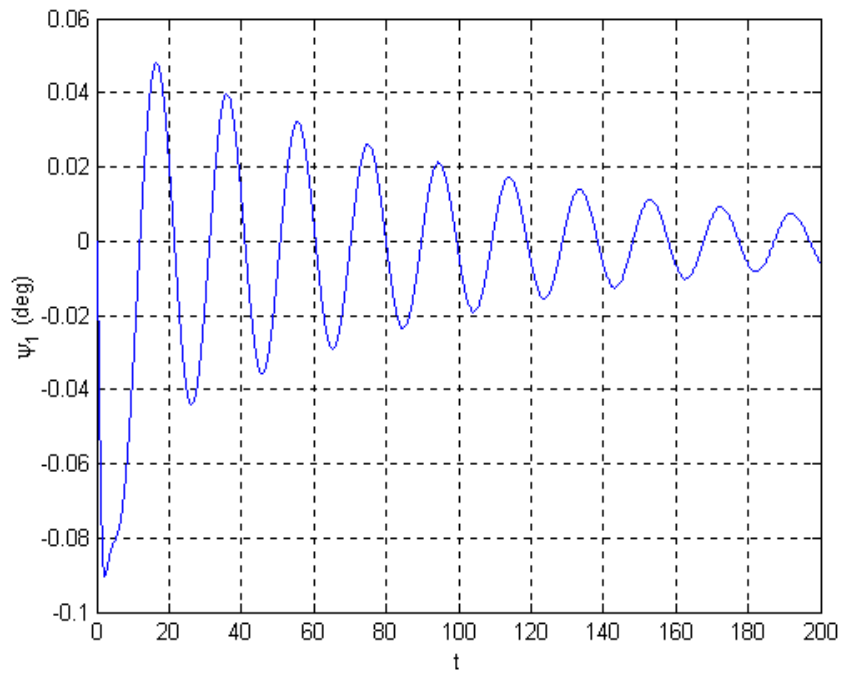


Figure 7. Leading Ship Heading Angle vs. Time

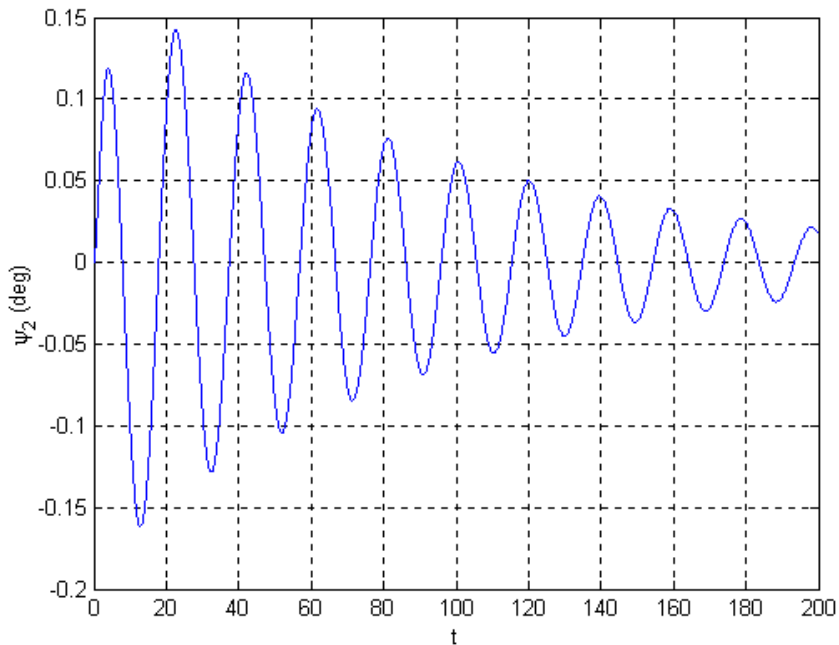


Figure 8. Trailing Ship Heading Angle vs. Time

2. Unstable Condition

The figures shown below were obtained for the following parameters:

Tension=0.005

Length=1.2

Time Constant=1.2

The results clearly show an oscillatory divergence, and the system is unstable despite the stabilizing effect of the leading ship control law. The stability analysis of the following chapter will explain these results.

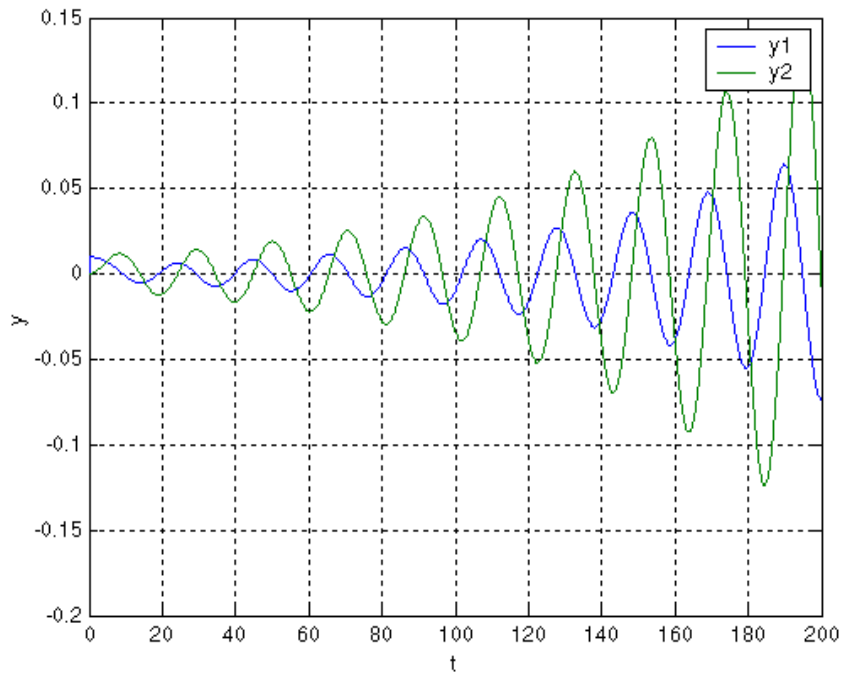


Figure 9. Ship Offsets vs. Time

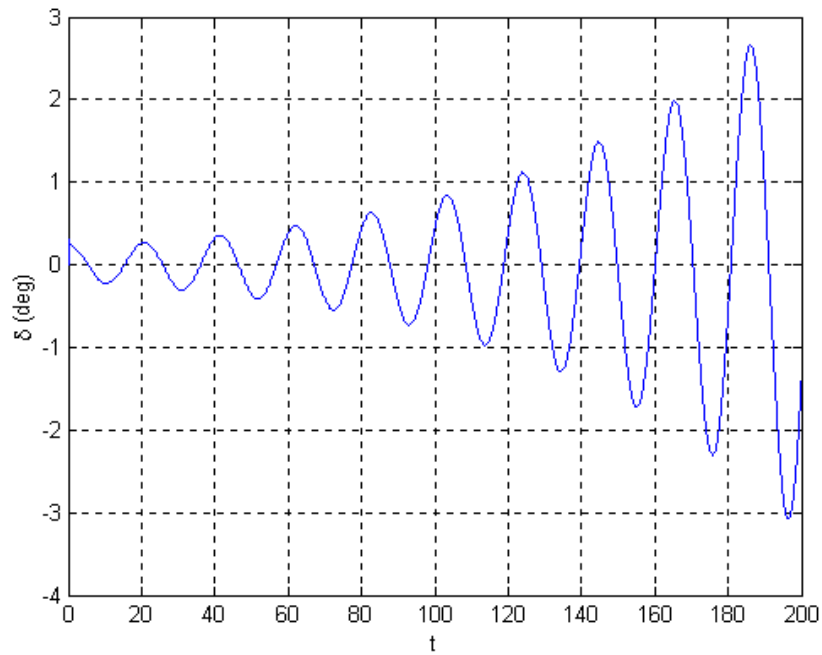


Figure 10. Leading Ship Rudder Angle vs. Time

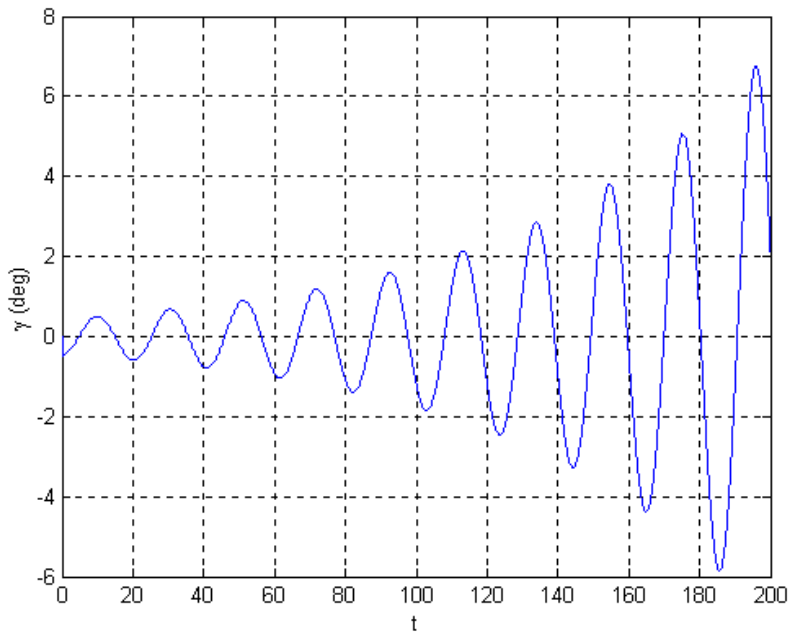


Figure 11. Towline Angle vs. Time

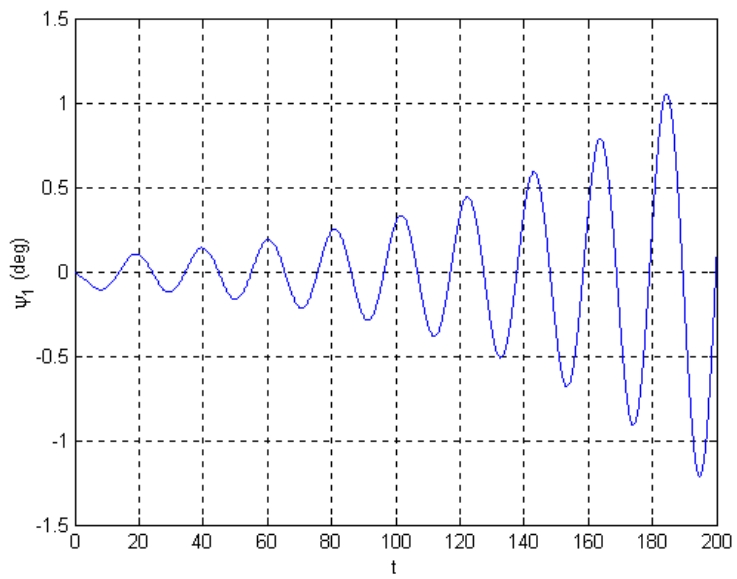


Figure 12. Leading Ship Heading Angle vs. Time

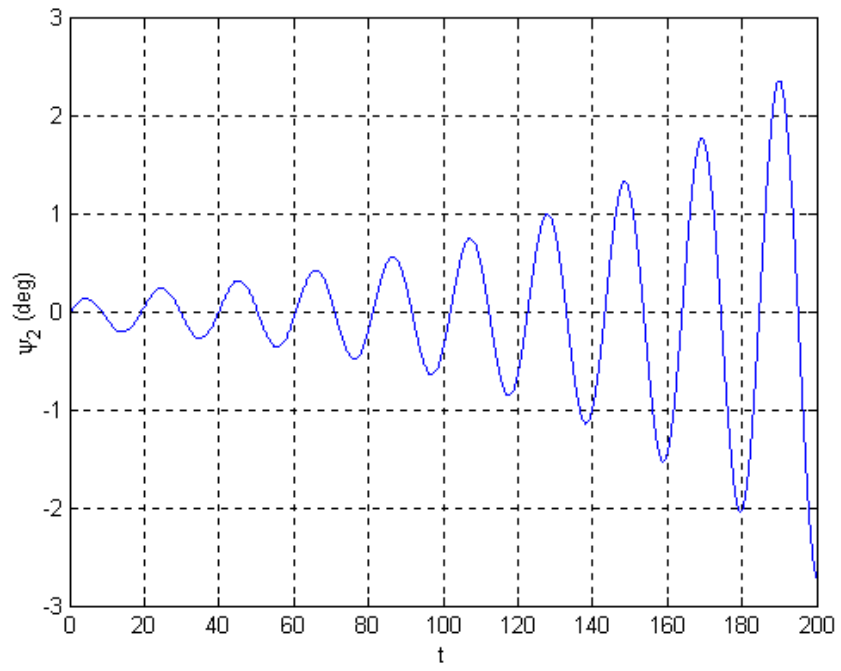


Figure 13. Trailing Ship Heading Angle vs. Time

THIS PAGE INTENTIONALLY LEFT BLANK

III. STABILITY ANALYSIS

A. LINEARIZATION

Since the equations of motion have changed by the inclusion effect of rudder, we have to again linearize the equations to see whether the system is stable or not. We will use the same assumptions that are mentioned before in Chapter 2 Section 2.

When we linearize equations (2), (3), (8), (9), (13), (42), (43) we get

$$(m_2 - Y_{\dot{v}_2})\dot{v}_2 - Y_{\dot{r}_2}\dot{r}_2 = -m_2 r_2 + Y_{v_2} v_2 + Y_{r_2} r_2 - T(\psi_2 + \gamma) \quad (45)$$

$$(I_{Z2} - N_{\dot{r}_2})\dot{r}_2 - N_{\dot{v}_2}\dot{v}_2 = N_{v_2} v_2 + N_{r_2} r_2 - T x_{p2}(\psi_2 + \gamma) \quad (46)$$

$$(m_1 - Y_{\dot{v}_1})\dot{v}_1 - Y_{\dot{r}_1} r_1 = -m_1 r_1 + Y_{v_1} v_1 + Y_{r_1} r_1 + T(\psi_1 + \gamma) + Y_{\delta} \delta \quad (47)$$

$$(I_{Z1} - N_{\dot{r}_1})\dot{r}_1 - N_{\dot{v}_1} \dot{v}_1 = N_{v_1} v_1 + N_{r_1} r_1 - T x_{p1}(\psi_1 + \gamma) + N_{\delta} \delta \quad (48)$$

$$\dot{y}_1 = \psi_1 + v_1 \quad (49)$$

$$\dot{y}_2 = \psi_2 + v_2 \quad (50)$$

$$\sin \gamma = \frac{y_2 + x_{p2} \psi_2 - (y_1 - x_{p1} \psi_1)}{l} \quad (51)$$

We can convert the equations of motion into a matrix form in order to make it easier study the stability of the system. Since we changed only the equations of the first ship, equations (24), (25) are the same, but equations (26) and (27) are different

$$\dot{v}_2 = A_{2vv} v_2 + A_{2vr} r_2 + B_{2v}(\psi_2 + \gamma) \quad (24)$$

$$\dot{r}_2 = A_{2rv} v_2 + A_{2rr} r_2 + B_{2r}(\psi_2 + \gamma) \quad (25)$$

$$\dot{v}_1 = A_{1vv} v_1 + A_{1vr} r_1 + B_{1v}(\psi_1 + \gamma) + C_{1v} \delta \quad (52)$$

$$\dot{r}_1 = A_{1rv} v_1 + A_{1rr} r_1 + B_{1r}(\psi_1 + \gamma) + C_{1r} \delta \quad (53)$$

The coefficients are the same as in Chapter 2 Section 2, with the following exceptions:

$$C_{1v} = [(I_{z1} - N_{\dot{r}1})Y_{\delta} + N_{\delta}Y_{\dot{r}1}] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (54)$$

$$C_{1r} = [Y_{\delta}N_{\dot{v}1} + N_{\delta}(m_1 - Y_{\dot{v}1})] \div [(m_1 - Y_{\dot{v}1})(I_{z1} - N_{\dot{r}1}) - N_{\dot{v}1}Y_{\dot{r}1}] \quad (55)$$

As mentioned in Chapter 1, all hydrodynamic coefficients are taken from reference [2] with the exception of Y_{δ} and N_{δ} . The matlab program shown in Appendix F was used to calculate these coefficients. Since no data were available on the SLICE rudder design, we made an assumption of a ship turning radius of three ship lengths under fifteen degrees of rudder, and calculated the necessary values of the rudder hydrodynamic coefficients to achieve that turning radius.

As mentioned before, the rudder feedback coefficients k_{ψ} , k_v , k_r and k_y are calculated by standard pole placement techniques. The matlab program shown in Appendix G performs this calculation.

Now we can set our matrix to analysis the stability of the system. The new system matrix is 8x8 and is written as shown below

$$\begin{bmatrix} \dot{v}_2 \\ \dot{r}_2 \\ \dot{v}_1 \\ \dot{r}_1 \\ \dot{y}_2 \\ \dot{y}_1 \\ \dot{\psi}_2 \\ \dot{\psi}_1 \end{bmatrix} = [A] \begin{bmatrix} v_2 \\ r_2 \\ v_1 \\ r_1 \\ y_2 \\ y_1 \\ \psi_2 \\ \psi_1 \end{bmatrix} \quad (56)$$

where

$$[A]=\begin{bmatrix} A_{2vv} & A_{2vr} & 0 & 0 & \frac{B_{2v}}{l} & -\frac{B_{2v}}{l} & B_{2v} + \frac{B_{2v}x_{p2}}{l} & \frac{B_{2v}x_{p1}}{l} \\ A_{2rv} & A_{2rr} & 0 & 0 & \frac{B_{2r}}{l} & -\frac{B_{2r}}{l} & B_{2r} + \frac{B_{2r}x_{p2}}{l} & \frac{B_{2r}x_{p1}}{l} \\ 0 & 0 & A_{1vv} - C_{1v}k_v & A_{1vr} - C_{1v}k_r & \frac{B_{1v}}{l} & -\frac{B_{1v}}{l} - C_{1v}k_y & \frac{B_{1v}x_{p2}}{l} & B_{1v} + \frac{B_{1v}x_{p1}}{l} - C_{1v}k_\psi \\ 0 & 0 & A_{1rv} - C_{1r}k_v & A_{1rr} - C_{1r}k_r & \frac{B_{1r}}{l} & -\frac{B_{1r}}{l} - C_{1r}k_y & \frac{B_{1r}x_{p2}}{l} & B_{1r} + \frac{B_{1r}x_{p1}}{l} - C_{1r}k_\psi \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (57)$$

After forming the matrix, we used the matlab code in Appendix H to find the eigenvalues of the matrix, which will dictate whether the system is stable or not. When we run the code, we experienced that the stability of the system changes according to different values of tension, length and the time constant. A systematic series of runs was conducted in order to ascertain the effects of these parameters.

B. DEGREE OF STABILITY

The critical eigenvalue of the 8x8 system; i.e., the eigenvalue that dictates stability or instability is the one that has the largest real part. We define this as the degree of stability of the system. A positive value indicates instability while a negative value indicates stability. A systematic series of runs was then conducted where the towline length, the towline tension, and the leading ship control time constant were varied. For this analysis we used the matlab code in Appendix I for varying length of the connection between the ships, the code in Appendix J for varying tension on the connection, and the code in Appendix K for varying control time constant.

The results for different tow lengths are shown in Appendix A, for different tensions in Appendix B, and for different time constants in Appendix C. Based on these results we can draw the following conclusions:

1. Relatively large time constants (a slow control law) cannot guarantee stability under towing. Sufficiently fast control laws may, depending on tension and

towline length, stabilize the complete system. Therefore, towing stability must be a consideration during leading ship control law design.

2. Sufficiently large values of the tension (i.e.; trailing ship resistance) may stabilize the complete two-body system. This result is consistent with earlier observations reported in Ref. [3].
3. For very large values of the towline length, the effect of the leading ship control appears to be small. Therefore, the previously reported results in the literature [3] can be considered as a large towline length of the two-body model developed in this work.

C. REGIONS OF STABILITY

The previous results can be summarized in a single graph designating regions of stability and instability. This graph is shown in Figure 14 which was produced by utilizing the Matlab code shown in Appendix L. The graph shows the critical value of the control time constant for stability versus the towline length, and is parameterized by the tension in the towline. Combinations of (T_C, L, T) below the corresponding curve will yield stable response, while combinations that are located above the curve will result in system instability. A comparison between the values of the parameters used in the previous two simulation cases and the results shown in Figure 14, demonstrates the agreement between numerical integrations and the theoretical predictions of the behavior of the system based on our stability analysis.

Finally, we present a comparison between our results and the results presented in Ref. [3], which were based on the 4th order system. In reference [3], the first necessary stability criterion based on the trailing ship is

$$x_p > \frac{N_v}{Y_v} \quad (58)$$

where x_p is the non-dimensional length between the center of the ship and the towline connection point.

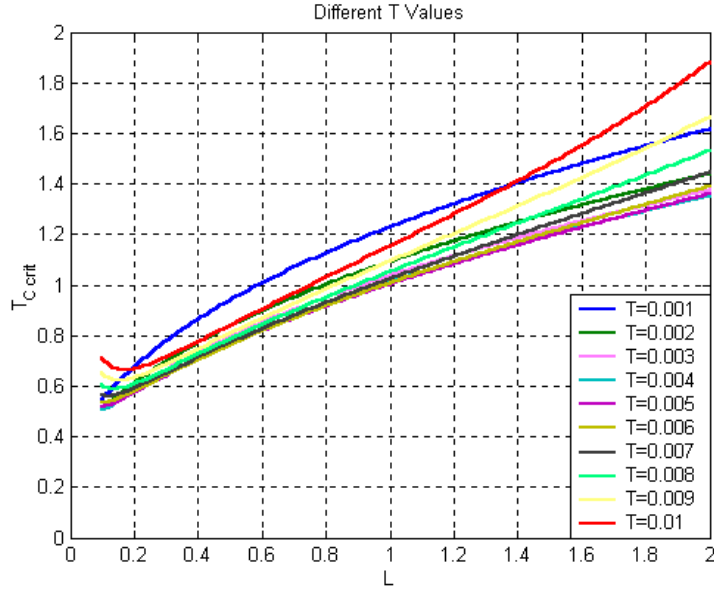


Figure 14. The Stability Region

The second necessary criterion is

$$T > -\frac{\alpha_2}{\alpha_1} \quad (59)$$

where T is the tension of the towed ship. If the above tension is negative then the criterion in equation (59) is inactive and is, therefore, automatically satisfied.

The parameters in the above equations are:

$$\begin{aligned} \alpha_2 = & (B_0 C_1 D_1 - A_0 D_1^2) + \frac{1}{l} (B_0 C_1 D_2 + B_0 C_2 D_1 - 2A_0 D_1 D_2) \\ & + \frac{1}{l^2} (B_0 C_2 D_2 - A_0 D_2^2) \end{aligned} \quad (60)$$

and

$$\alpha_2 = B_0 C_0 D_1 + \frac{1}{l} (B_0 C_0 D_2 - B_0^2 E_1) \quad (61)$$

The coefficients used in equations (60), and (61) are as follows

$$A_0 = (Y_{\dot{v}} - m)(N_{\dot{r}} - I_z) - N_{\dot{v}} Y_{\dot{r}} \quad (62)$$

$$B_0 = (Y_{\dot{v}} - m)N_r + Y_v(N_{\dot{r}} - I_z) + N_{\dot{v}}(m - Y_r) - Y_{\dot{r}}N_v \quad (63)$$

$$C_0 = Y_v N_r + N_v(m - Y_r) \quad (64)$$

$$C_1 = (m - Y_{\dot{v}})x_p + N_{\dot{v}} \quad (65)$$

$$C_2 = -(N_{\dot{r}} - I_z) + (m - Y_{\dot{v}})x_p^2 + x_p(N_{\dot{v}} + Y_{\dot{r}}) \quad (66)$$

$$D_1 = N_v - Y_v x_p \quad (67)$$

$$D_2 = -Y_v x_p^2 + N_v x_p + (Y_r - Y_{\dot{v}})x_p - N_r + N_{\dot{v}} \quad (68)$$

$$E_1 = 0 \quad (69)$$

Using the hydrodynamic coefficients of the Kaimalino estimated in [2], and presented in the Matlab code in several of the Appendices in this thesis, we can see that x_p is always greater than the ratio of N_v to Y_v . Furthermore, the critical tension is always negative as is estimated by the Matlab code shown in Appendix M. As a result, the previous stability criteria would have predicted a system that would always be stable and would have missed the stability and instability regions shown in Figure 14. Therefore, in towing stability studies, the designer should consider not only the effect of the trailing ship, but also the effect of the leading ship.

IV. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

This thesis presented a comprehensive study of the stability properties of the two-body ship towing problem. The results of this study can help in establishing rational guidelines which should be followed in order to ensure stability and safety of close proximity ship towing operations. The main results from this study can be summarized as follows:

1. Relatively large time constants (a slow control law) cannot guarantee stability under towing. Sufficiently fast control laws may, depending on tension and towline length, stabilize the complete system. Therefore, towing stability must be a consideration during leading ship control law design. This is in contrast with previous studies where the dynamics of the leading ship were not considered important in the overall analysis.
2. Sufficiently large values of the tension (i.e.; trailing ship resistance) may stabilize the complete two-body system. This result is consistent with earlier observations reported in the literature.
3. For very large values of the towline length, the effect of the leading ship control becomes less pronounced. Therefore, previously reported results in the literature can be considered as a large towline length of the two-body model developed in this work.

B. RECOMMENDATIONS

Recommendations for continuing studies are the following:

1. Study the effect of additional geometrical parameters such as attachment point location on the regions of stability and instability.
2. Perform a nonlinear analysis in order to analyze the mechanism of the loss of stability. Current results indicate that the predominant mechanism is through the generation of limit cycles (periodic solutions) but this needs to be substantiated further.

APPENDIX A. DEGREE OF STABILITY VS. TOWLINE LENGTH

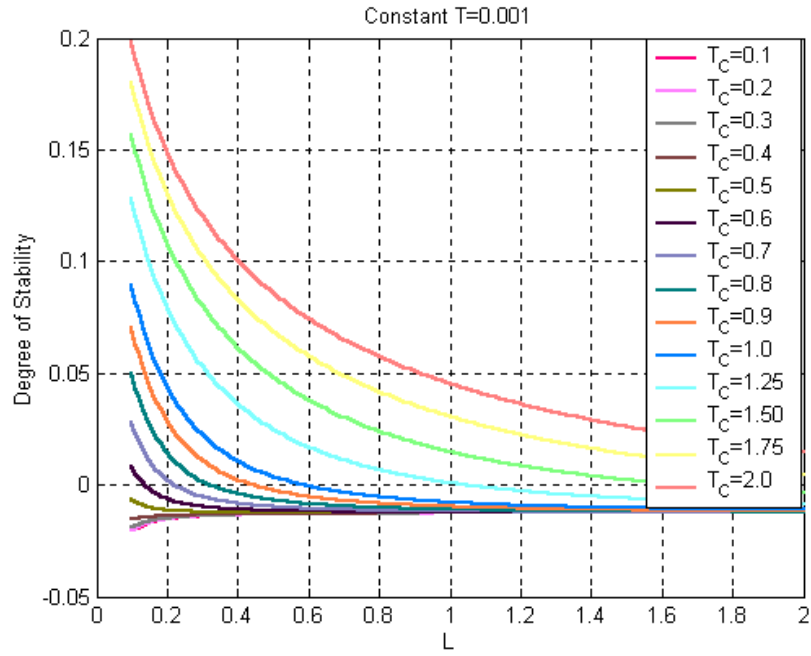


Fig A.1

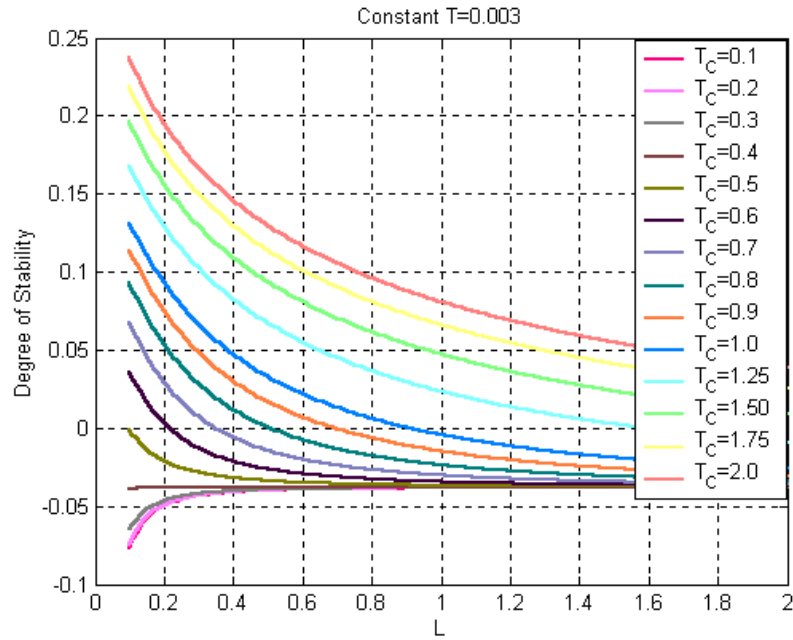


Fig A.2

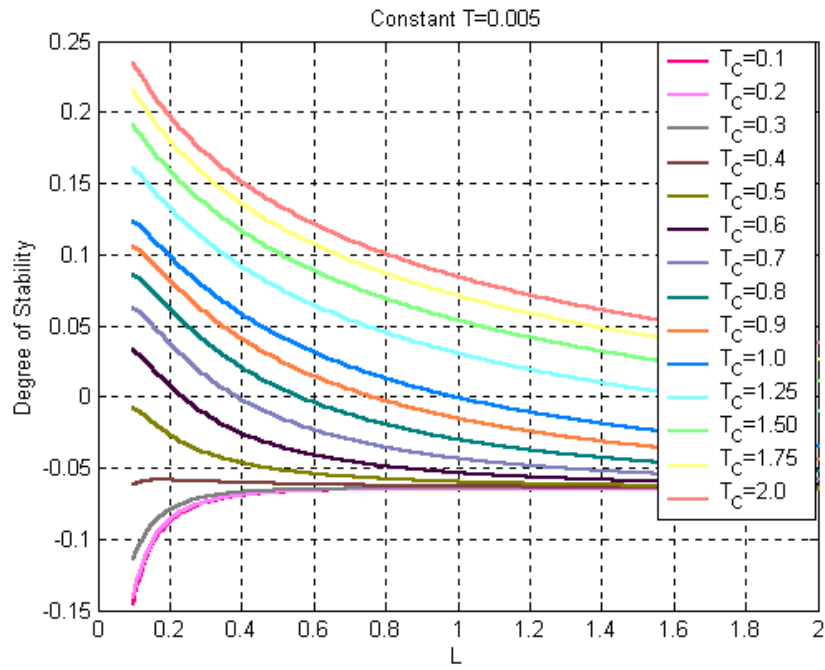


Fig A.3

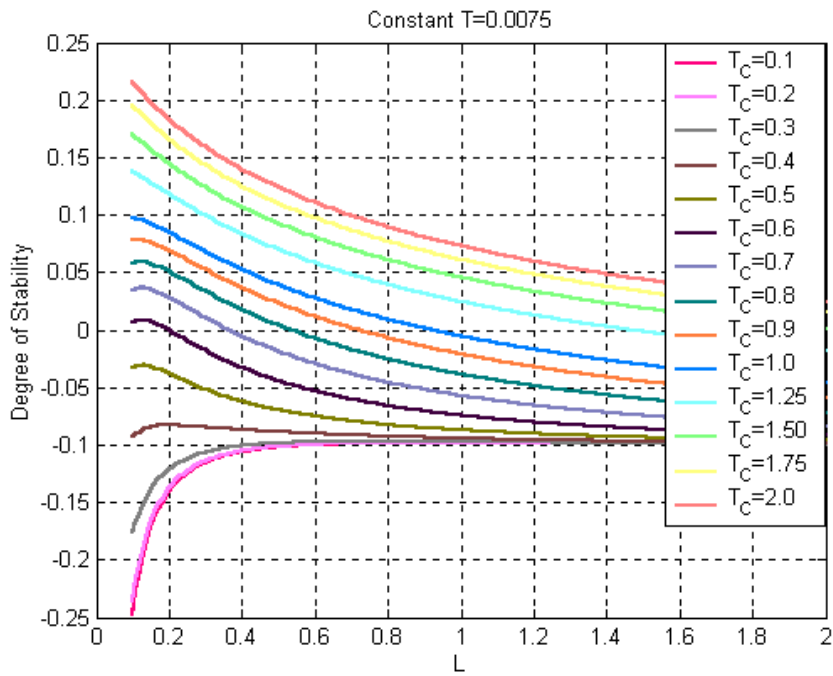


Fig A.4

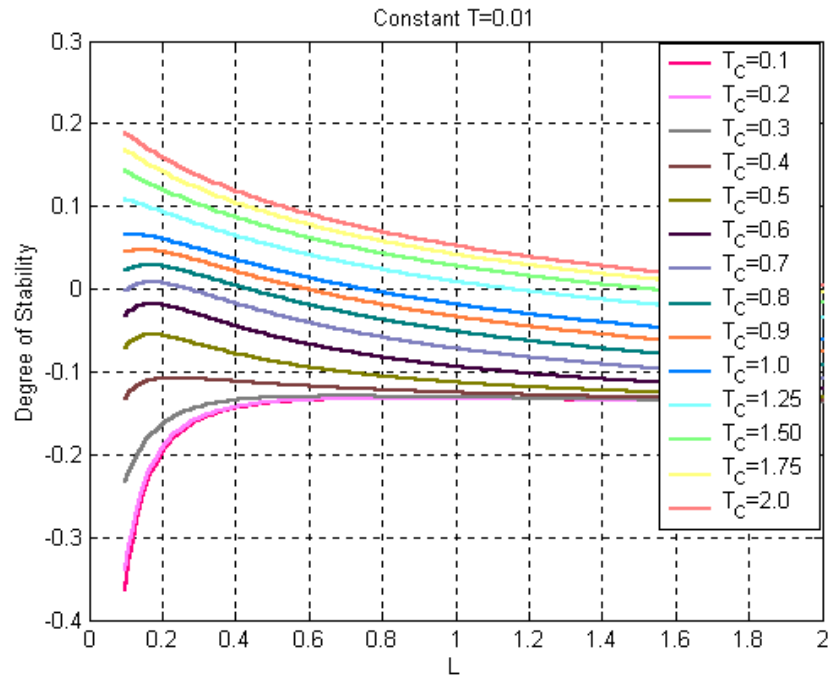


Fig A.5

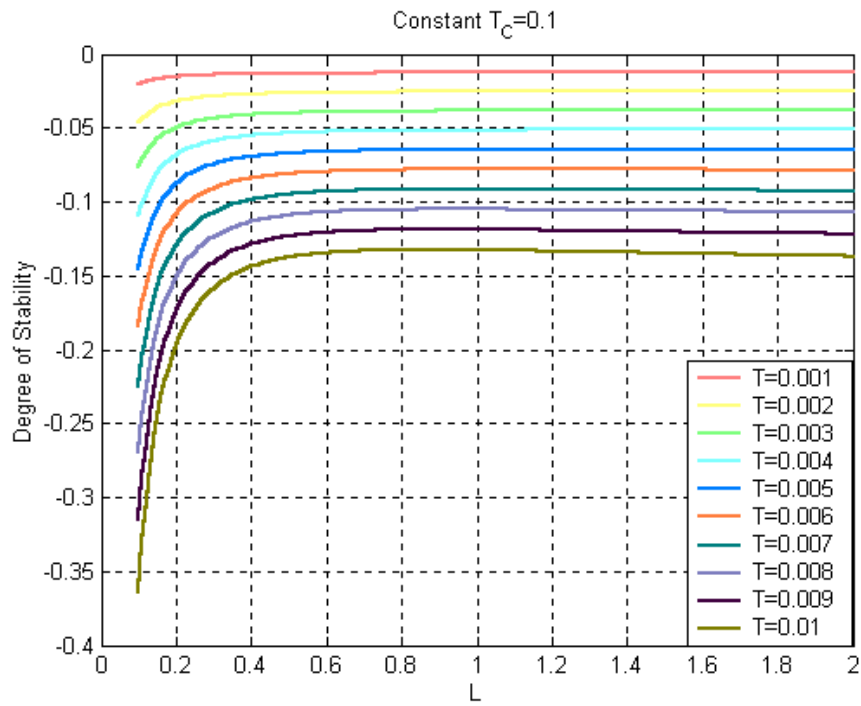


Fig A.6

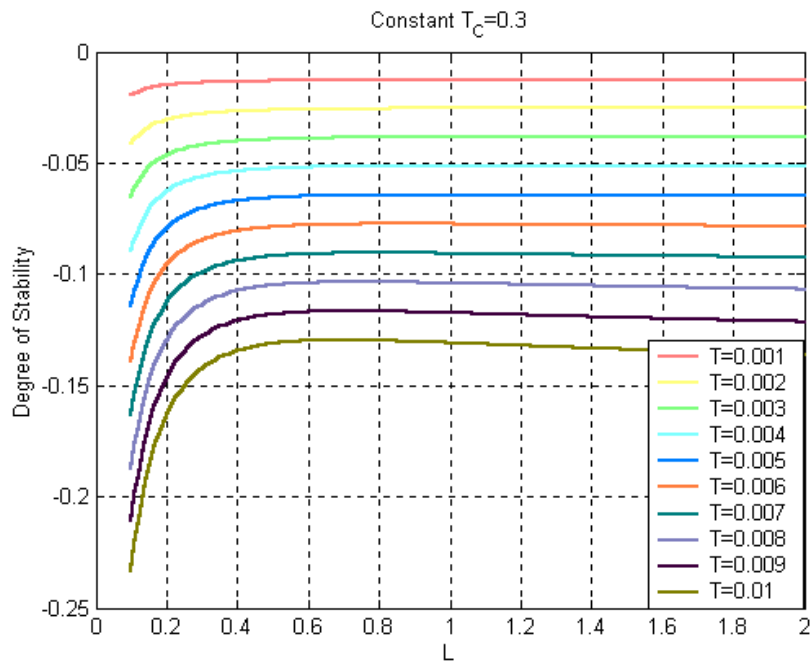


Fig A.7

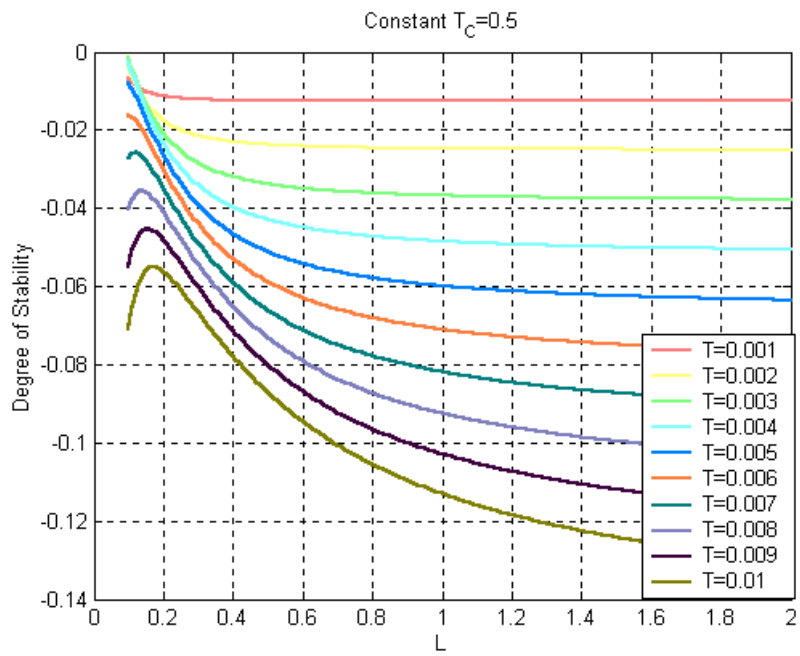


Fig A.8

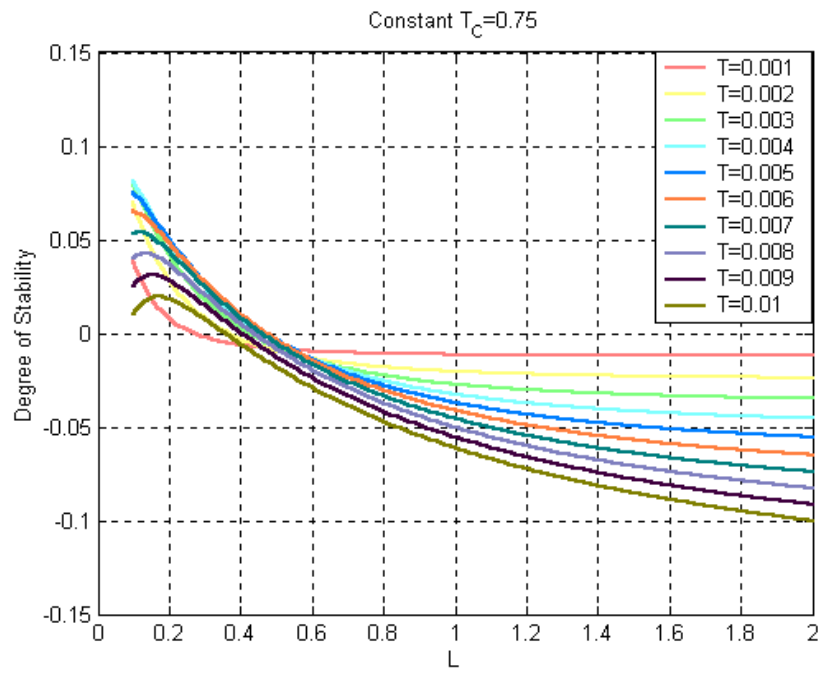


Fig A.9

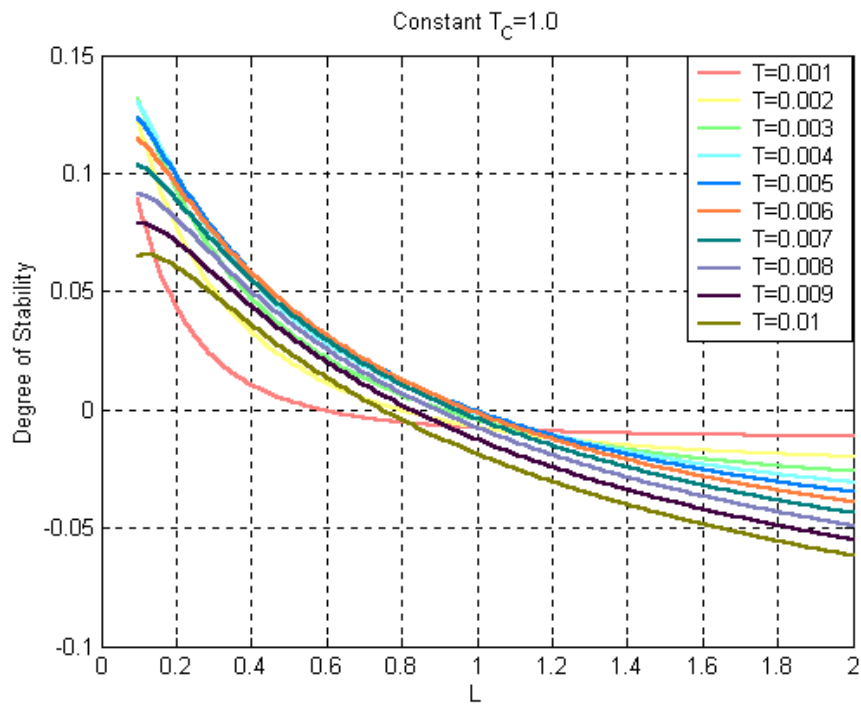


Fig A.10

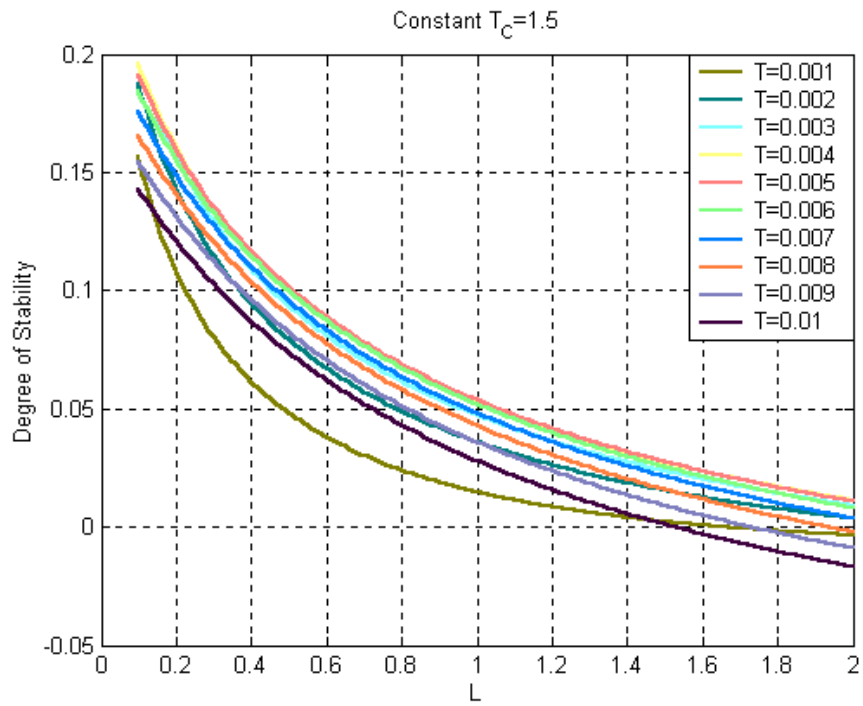


Fig A.11

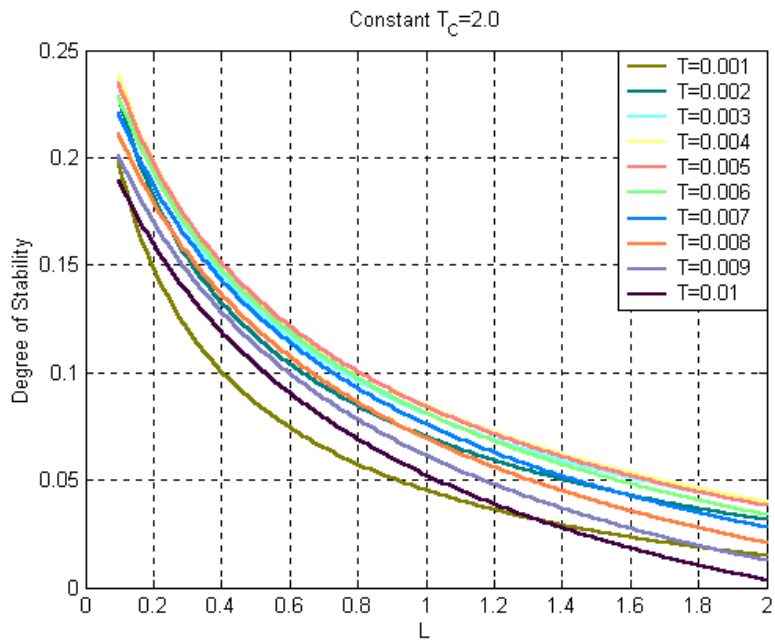


Fig A.12

APPENDIX B. DEGREE OF STABILITY VS. TENSION

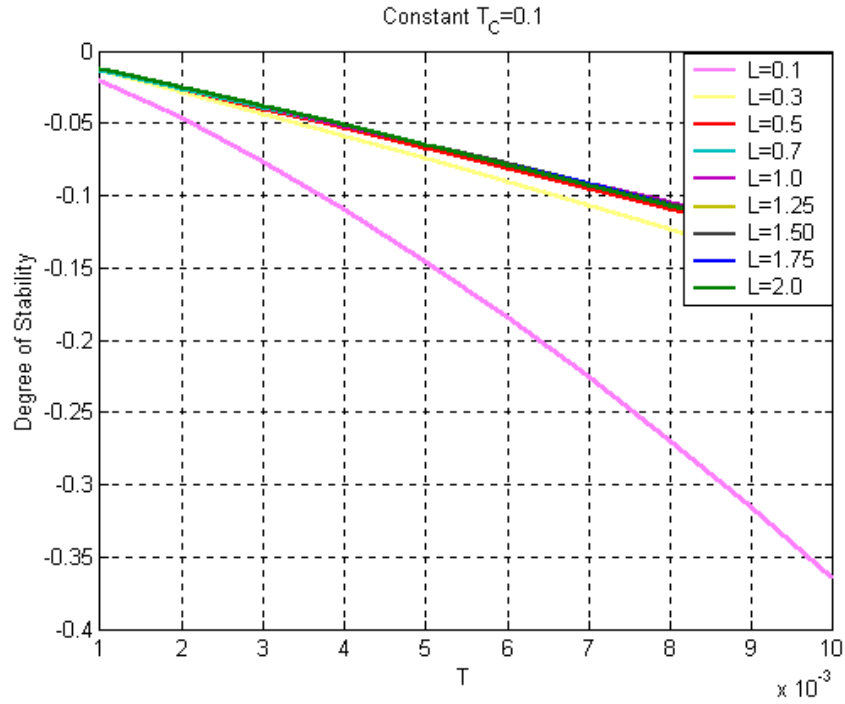


Fig B.1

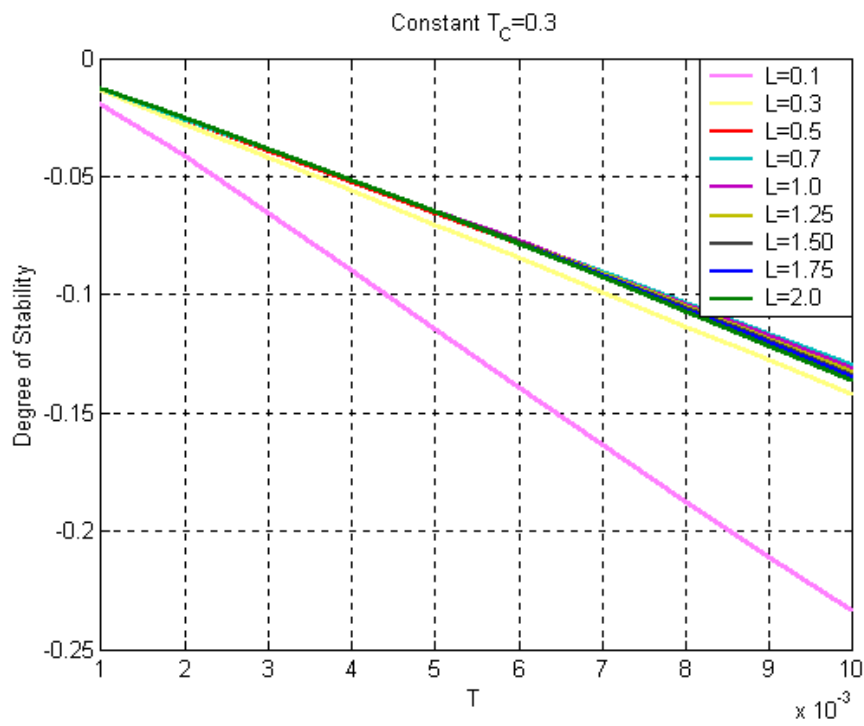


Fig B.2

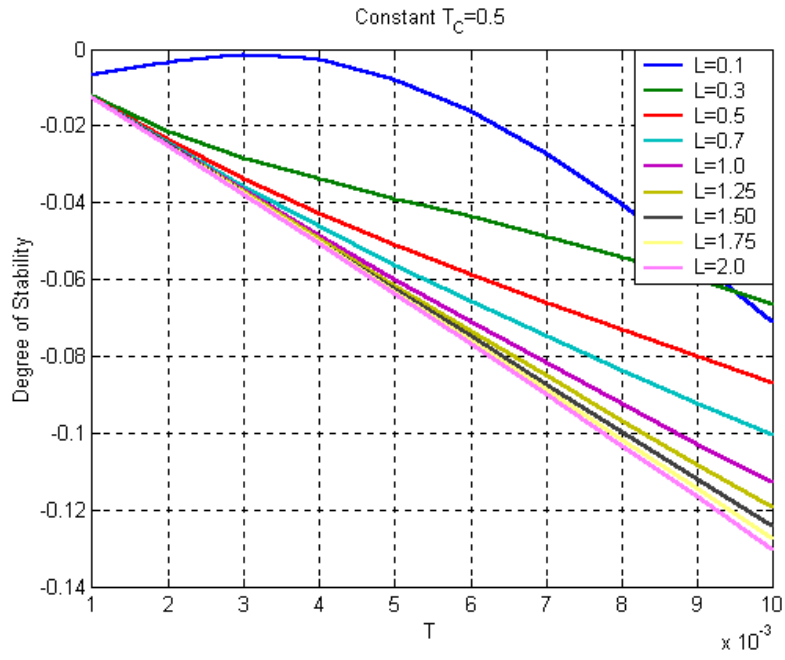


Fig B.3

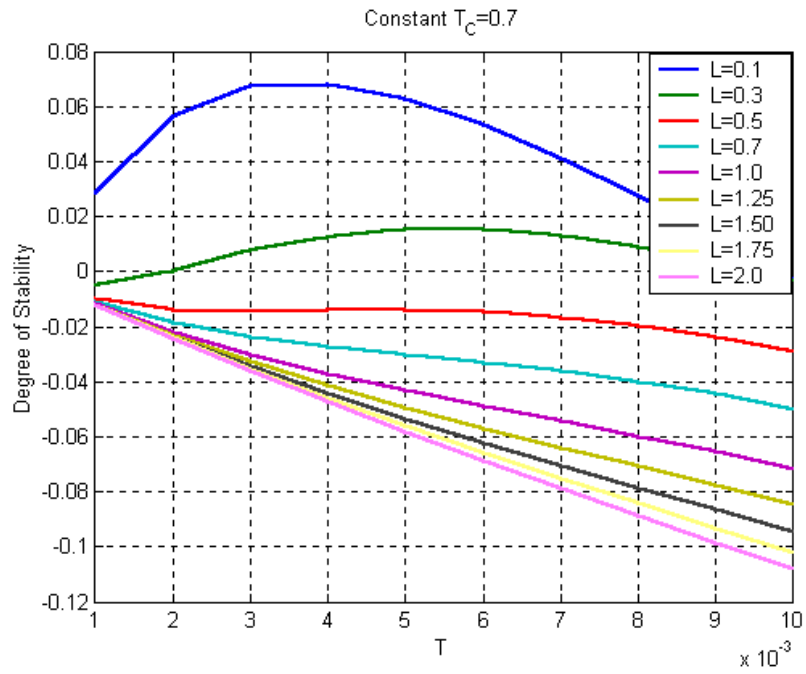


Fig B.4

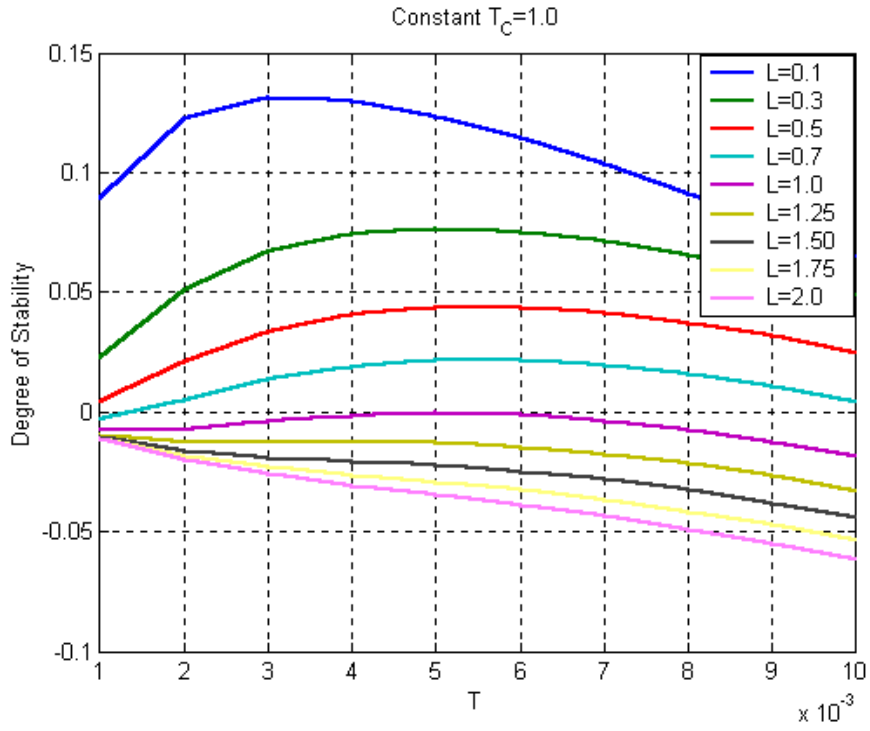


Fig B.5

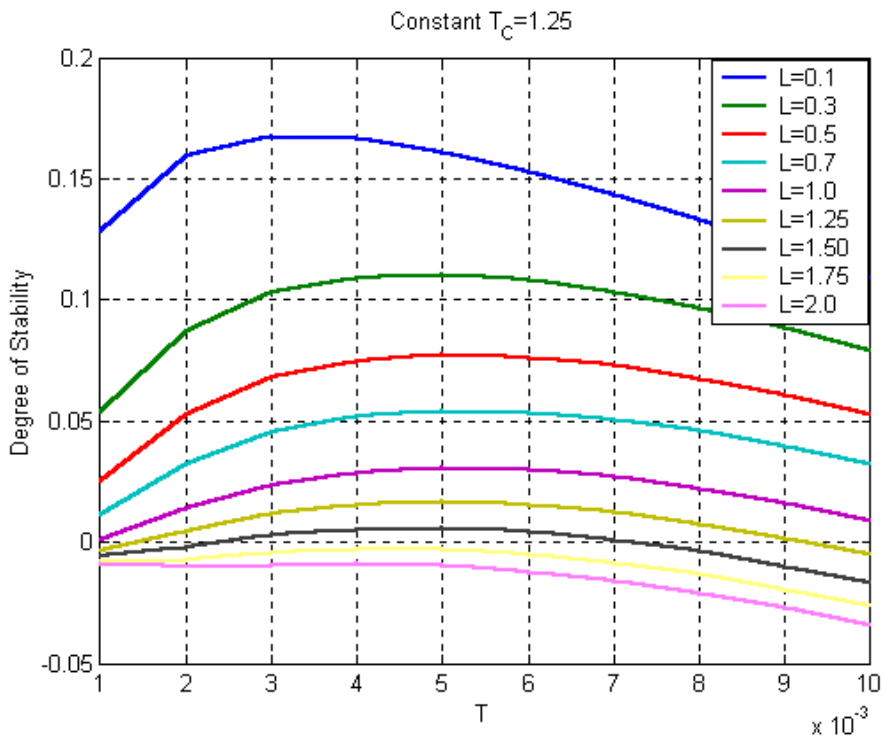


Fig B.6

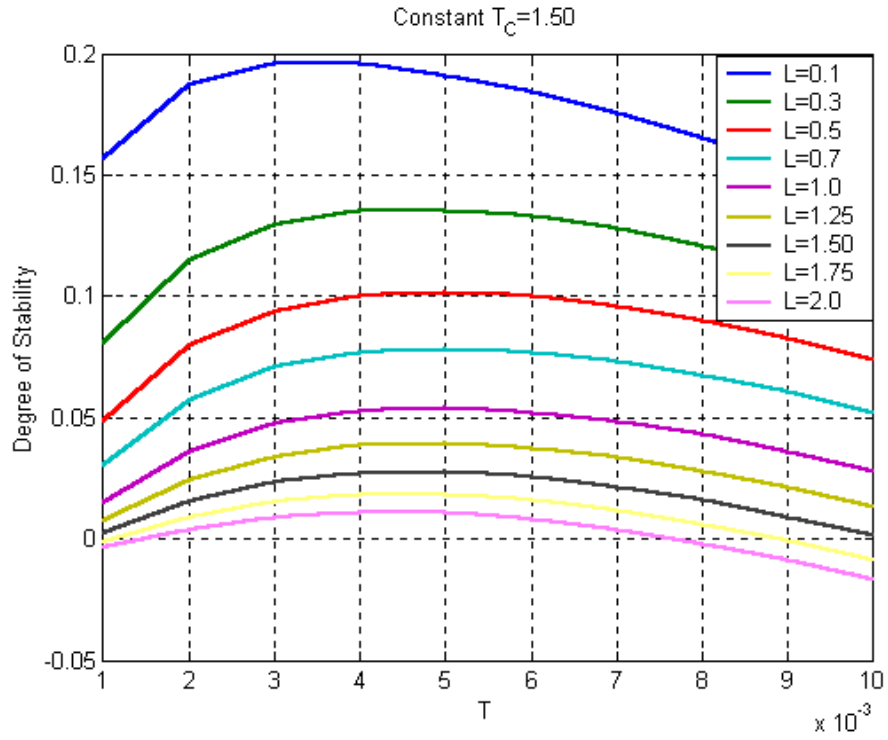


Fig B.7

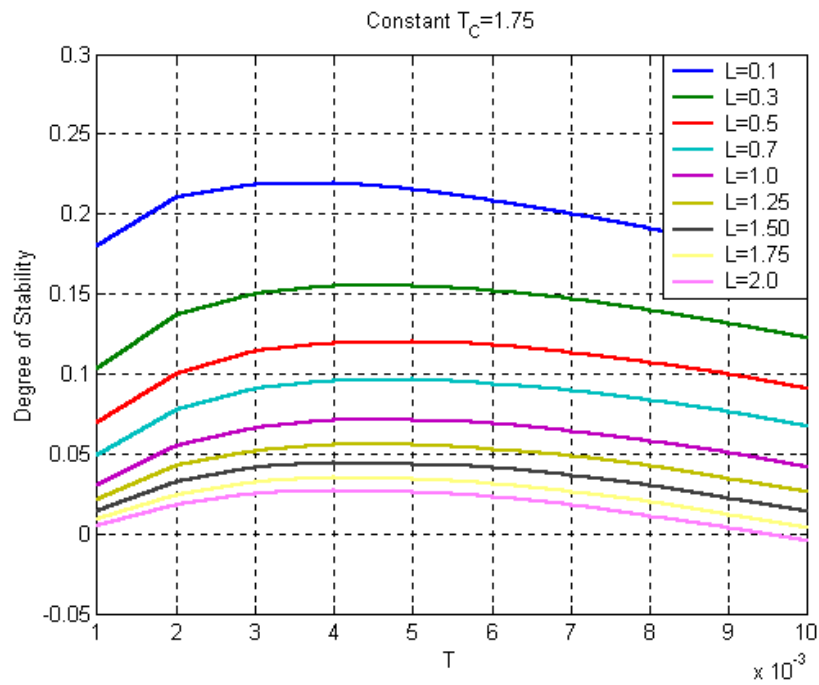


Fig B.8

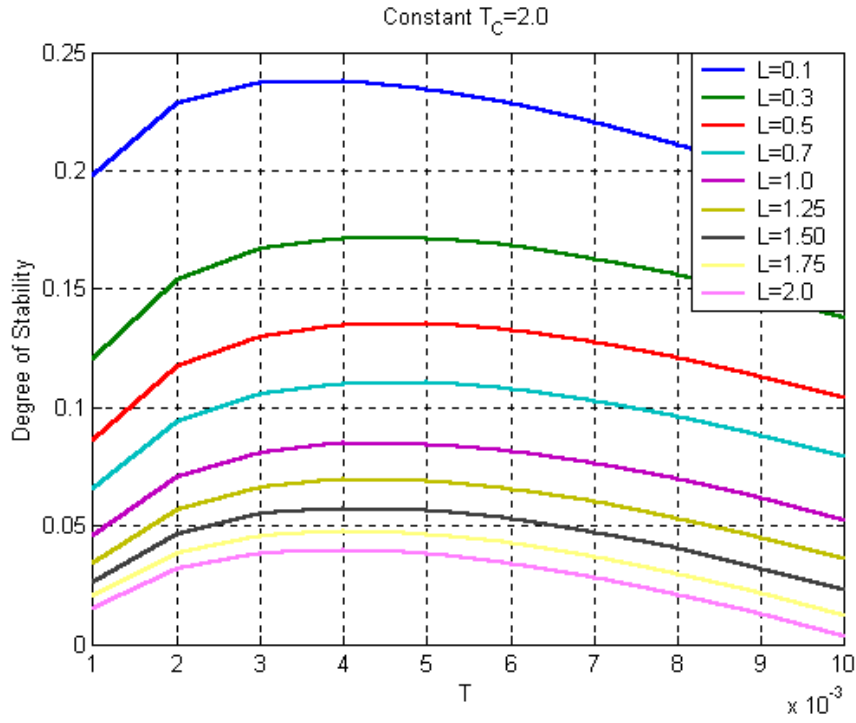


Fig B.9

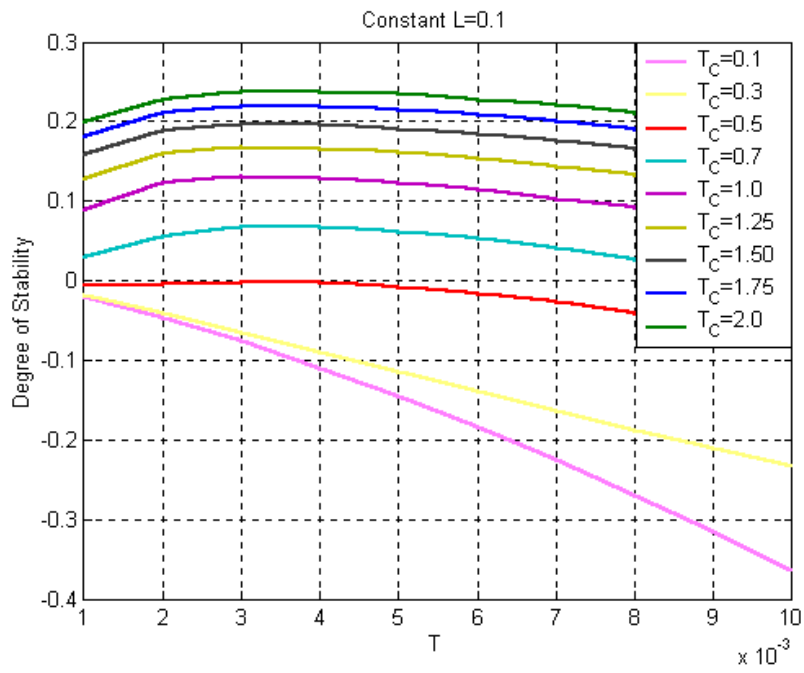


Fig B.10

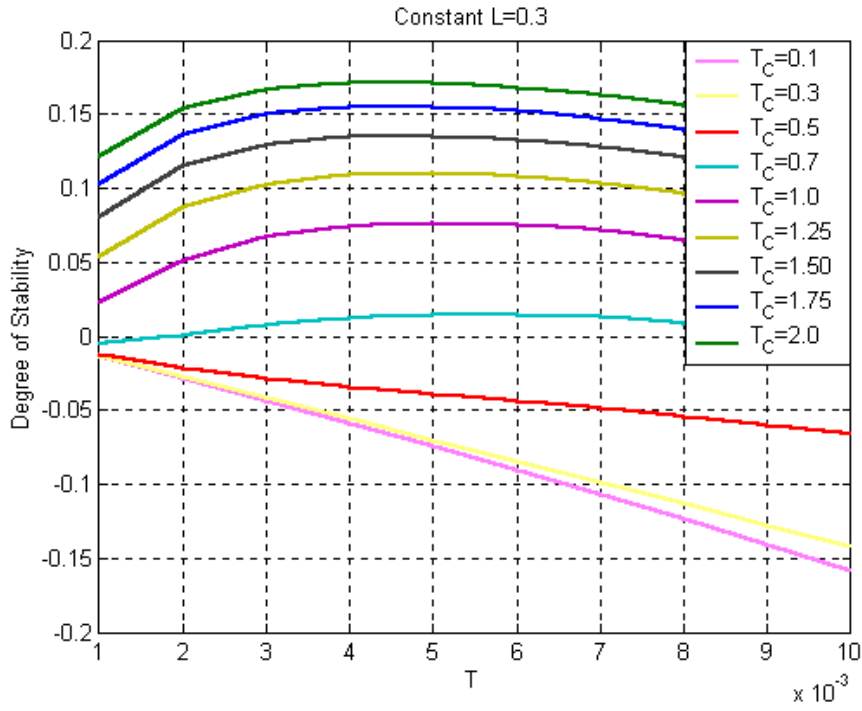


Fig B.11

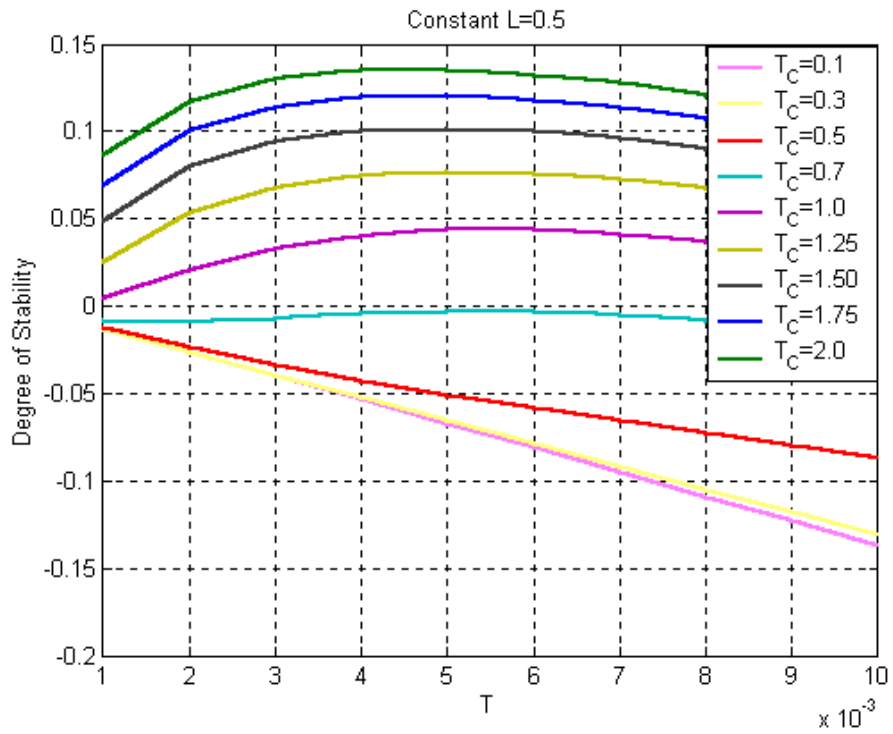


Fig B.12

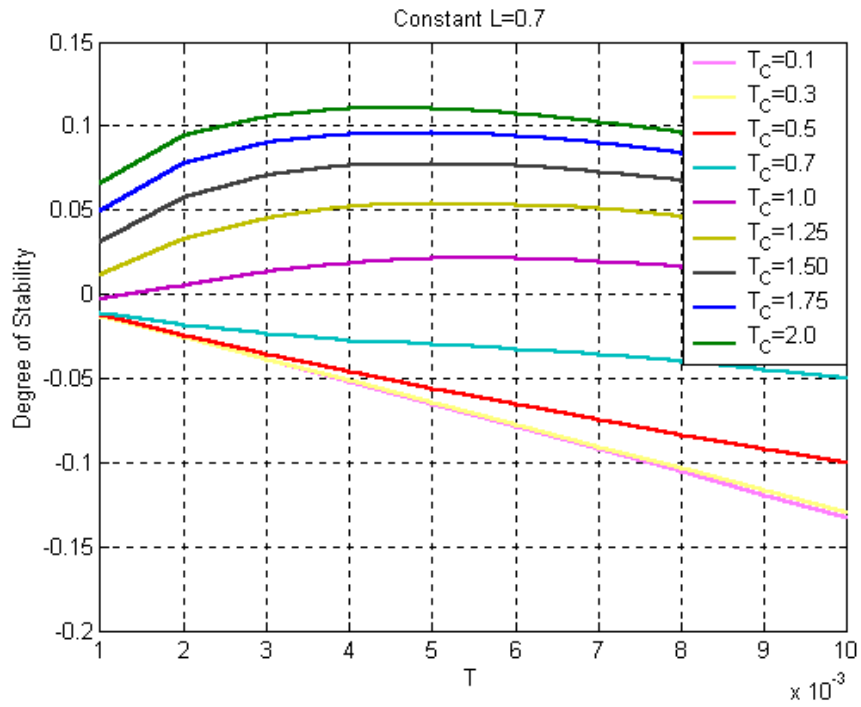


Fig B.13

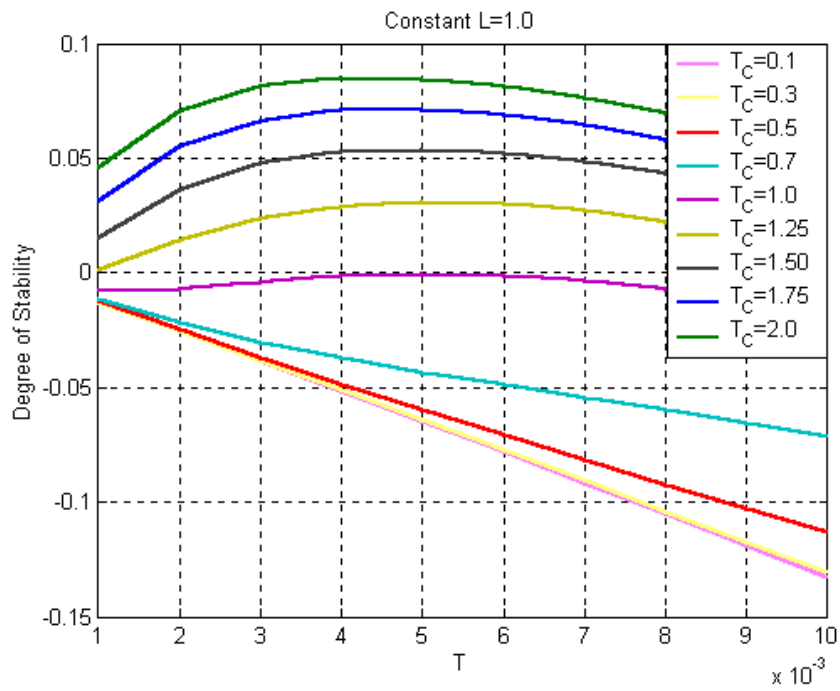


Fig B.14

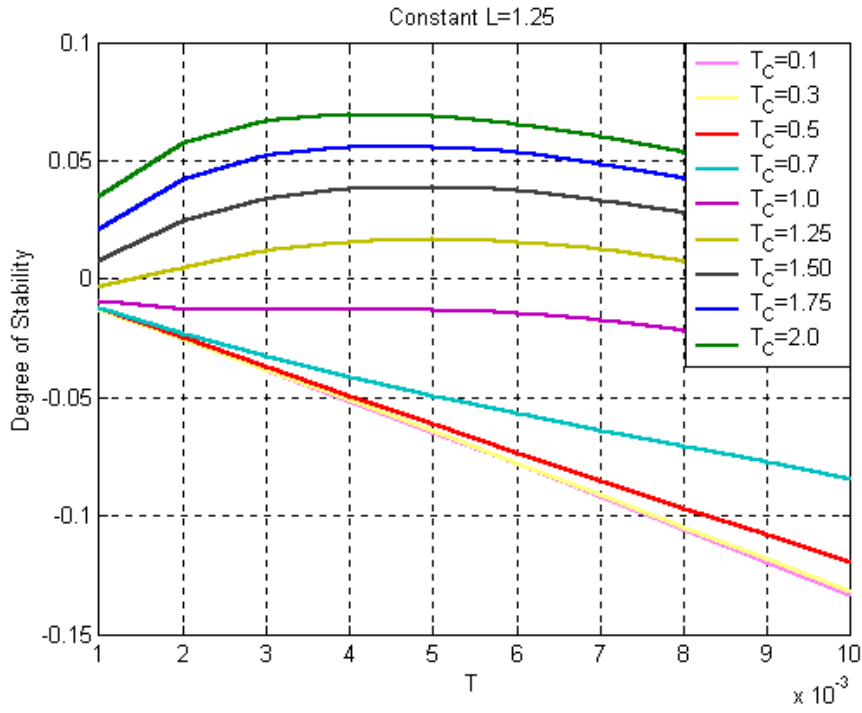


Fig B.15

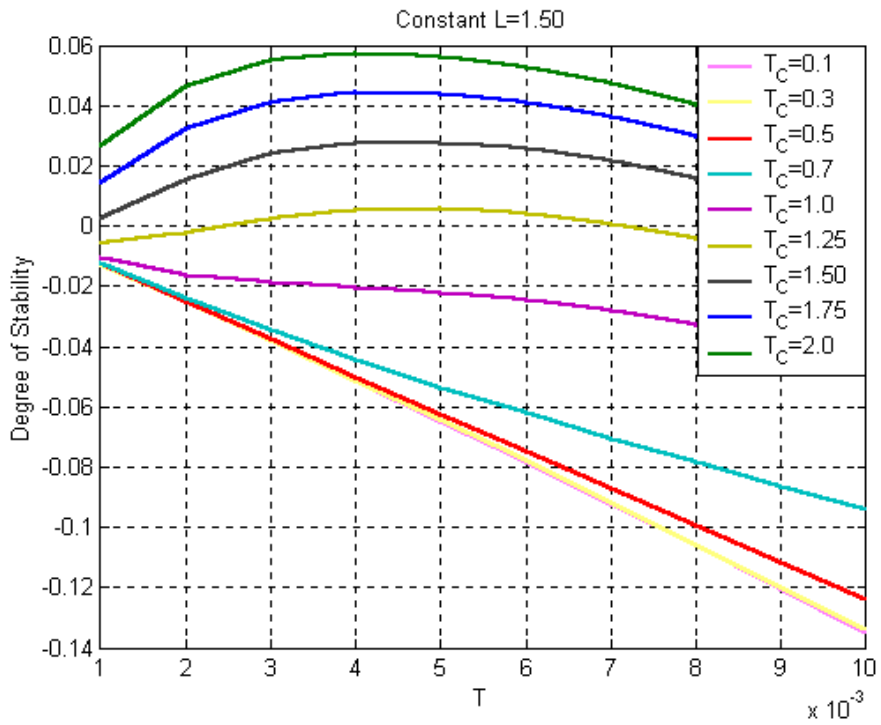


Fig B.16

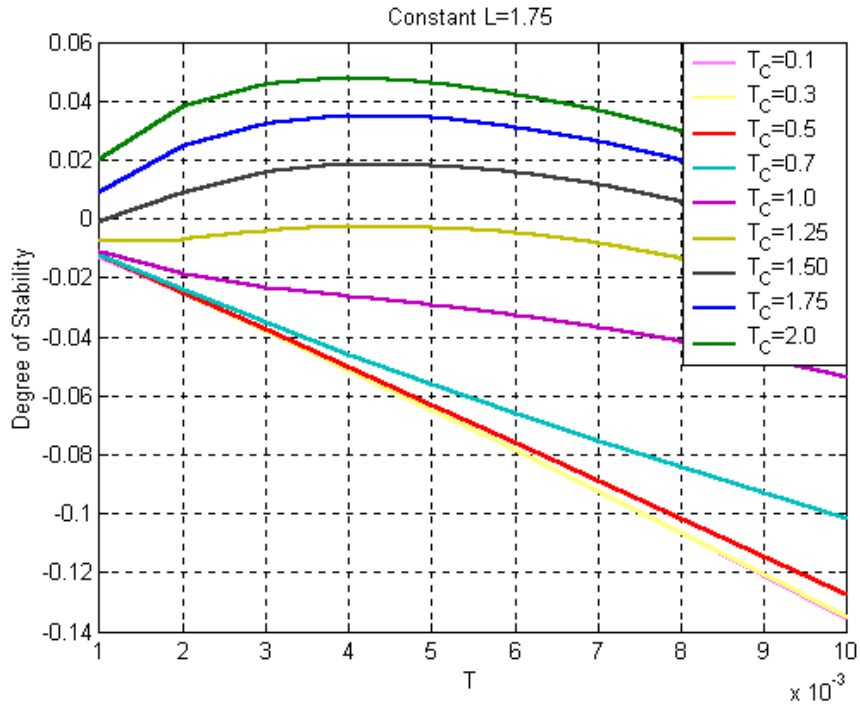


Fig B.17

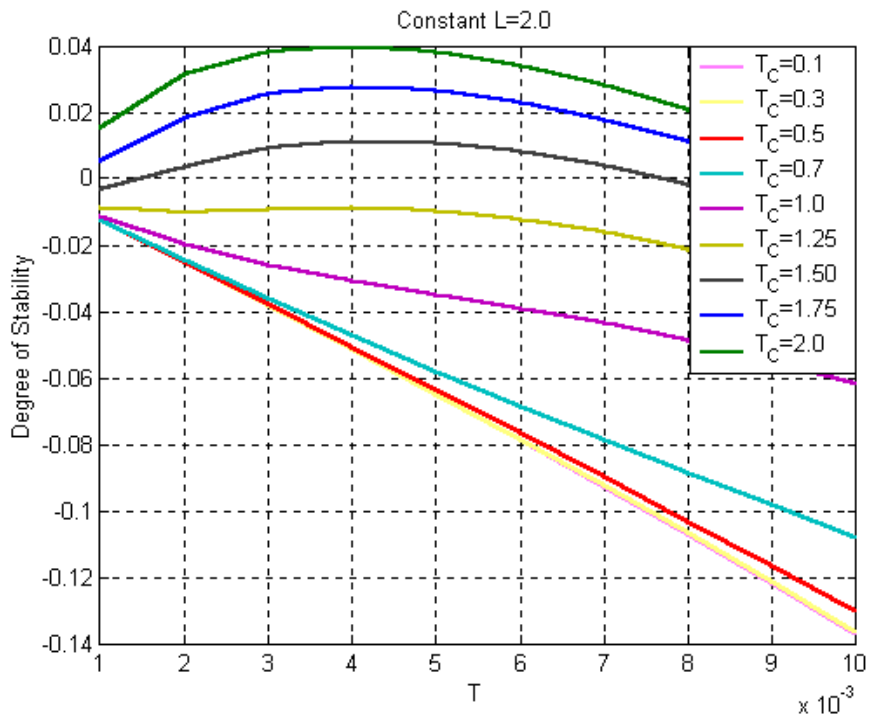


Fig B.18

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. DEGREE OF STABILITY VS. TIME CONSTANT

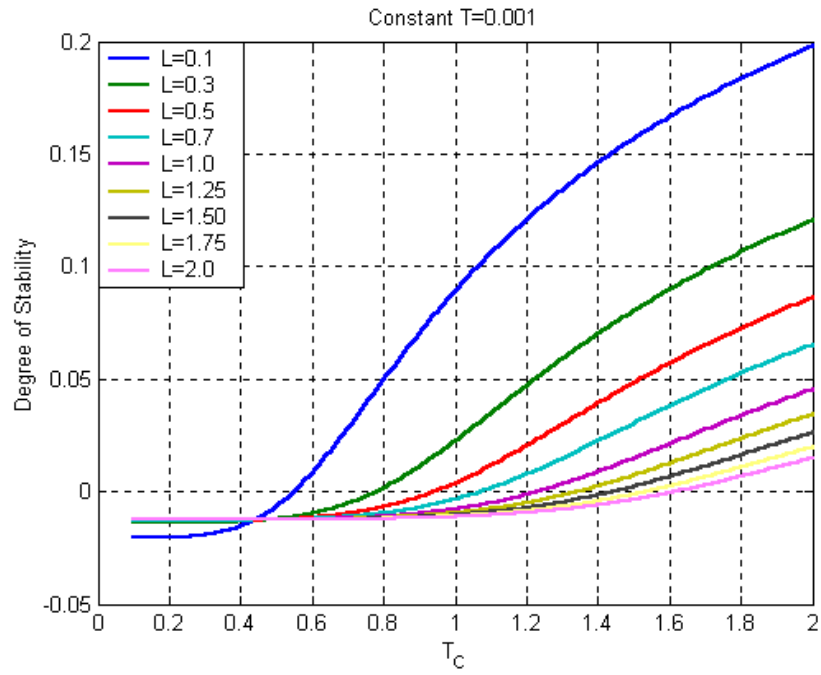


Fig C.1

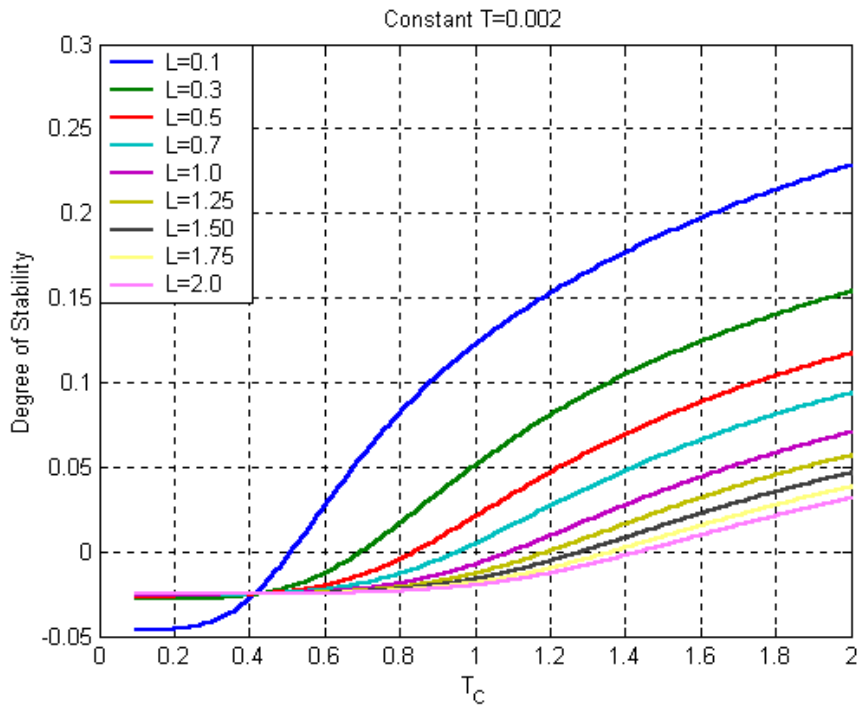


Fig C.2

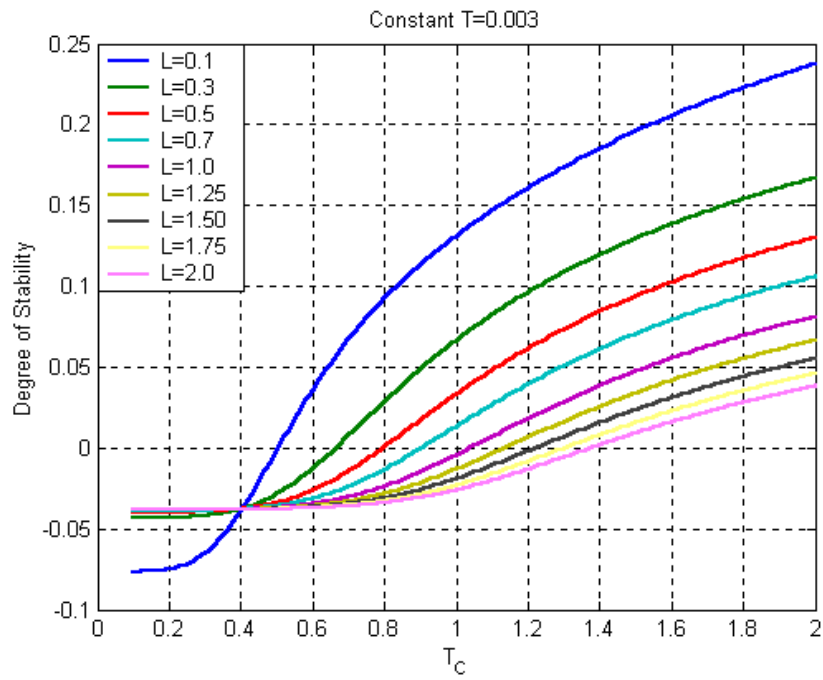


Fig C.3

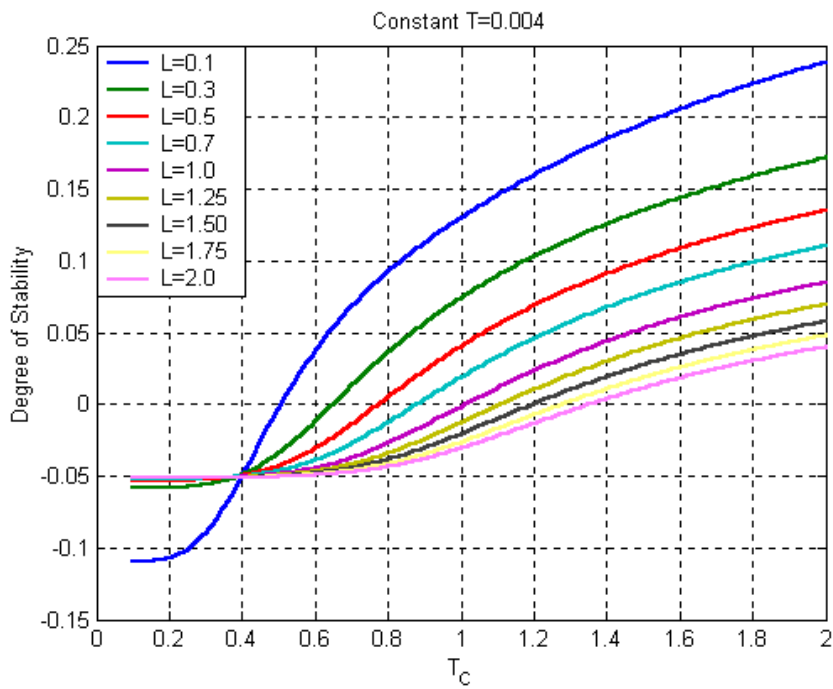


Fig C.4

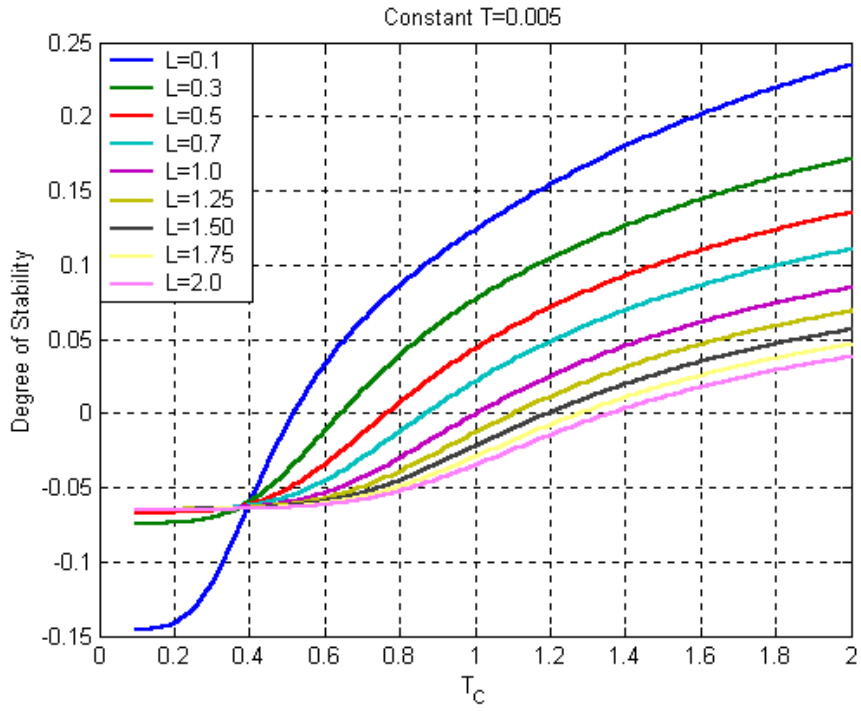


Fig C.5

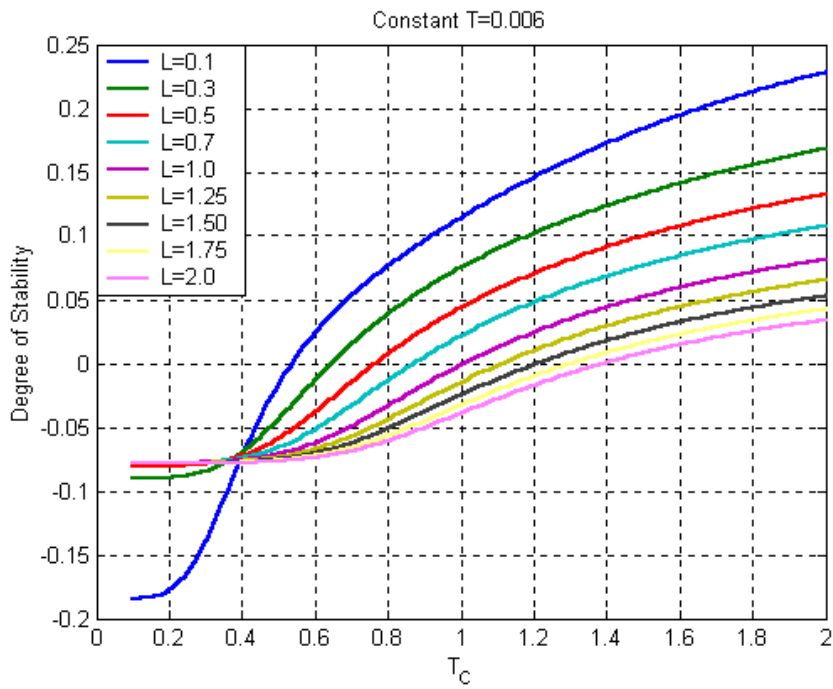


Fig C.6

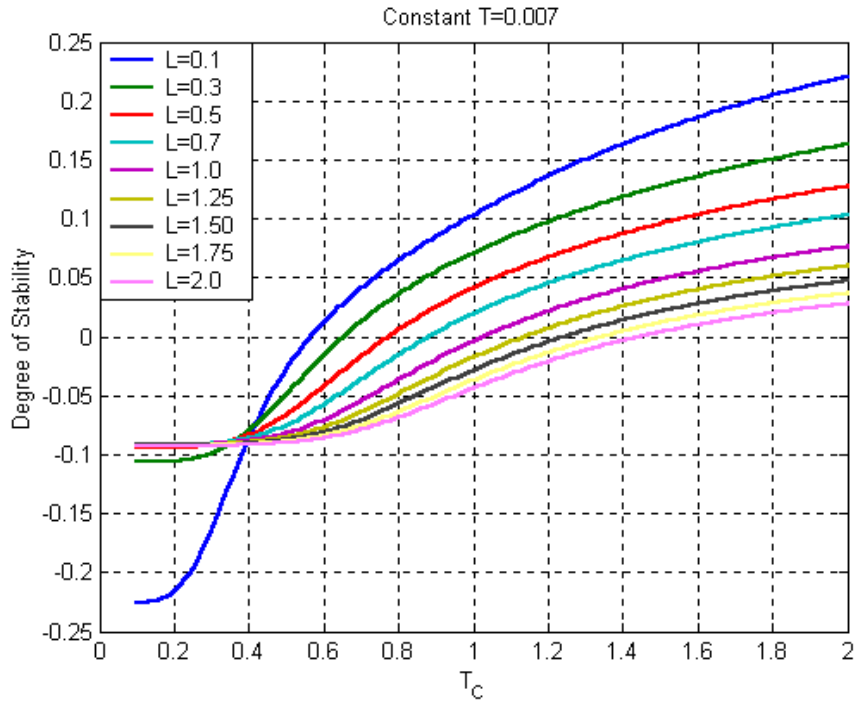


Fig C.7

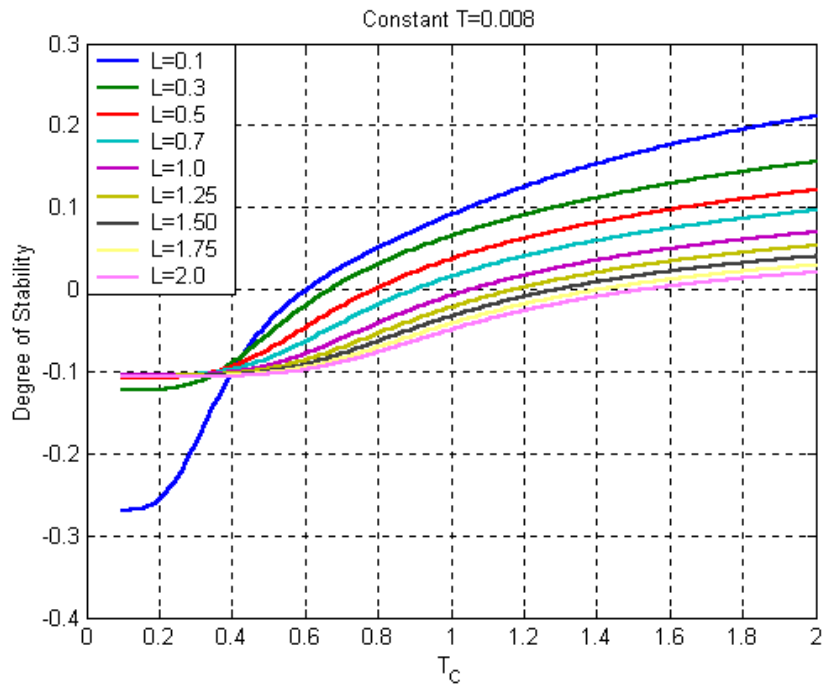


Fig C.8

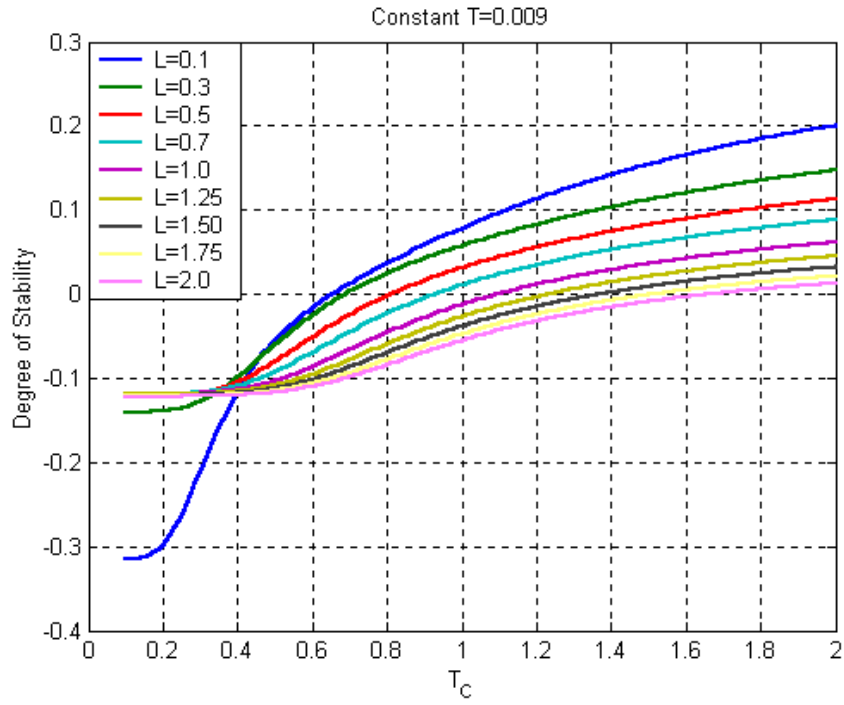


Fig C.9

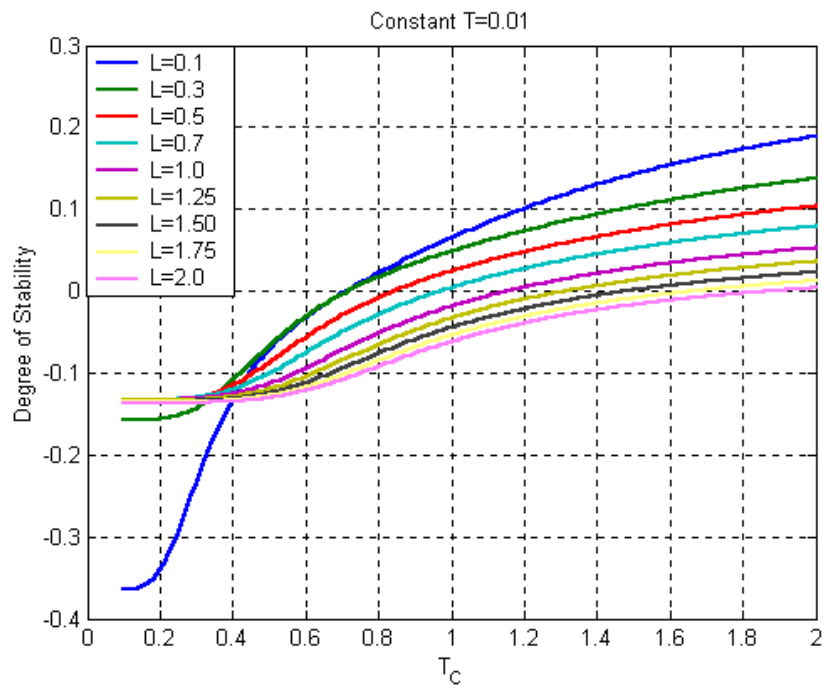


Fig C.10

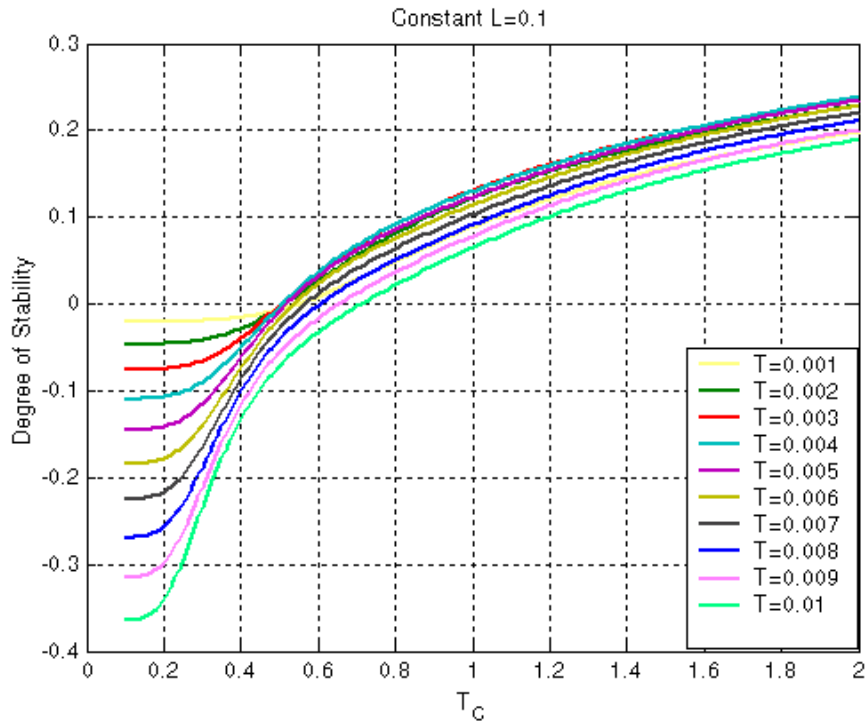


Fig C.11

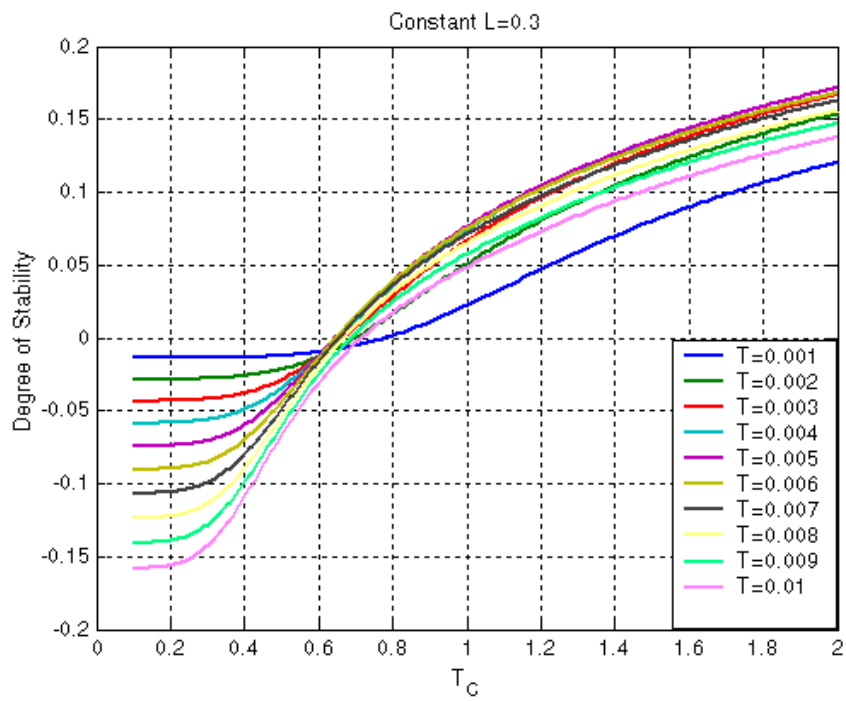


Fig C.12

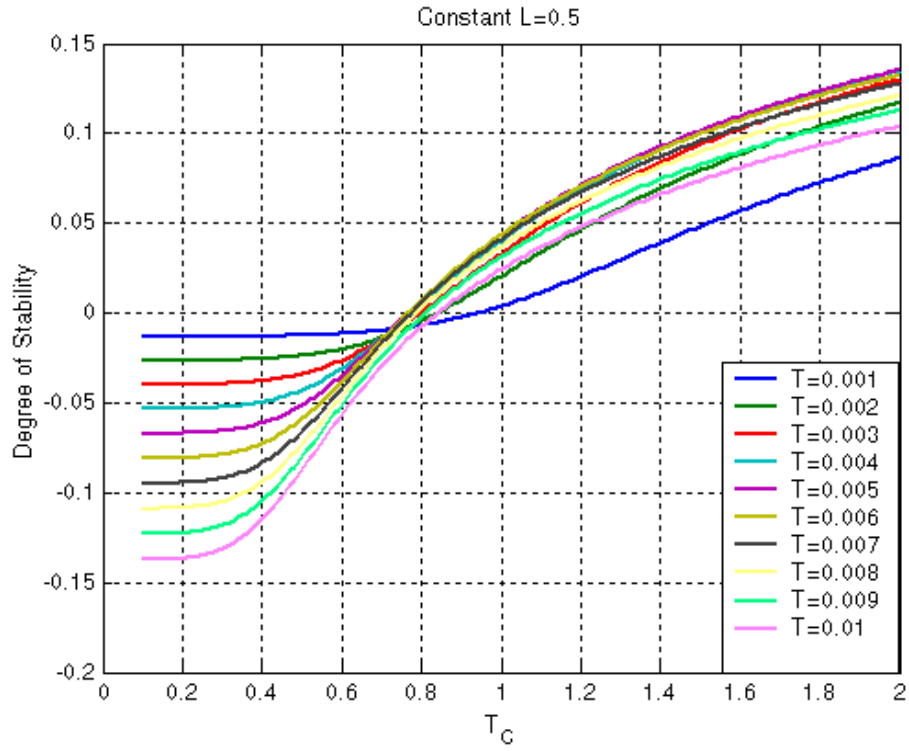


Fig C.13

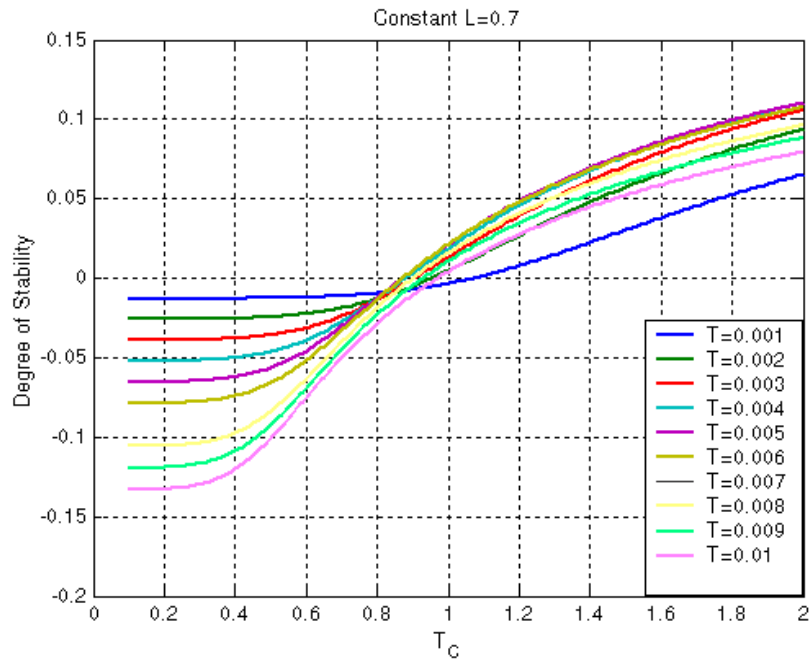


Fig C.14

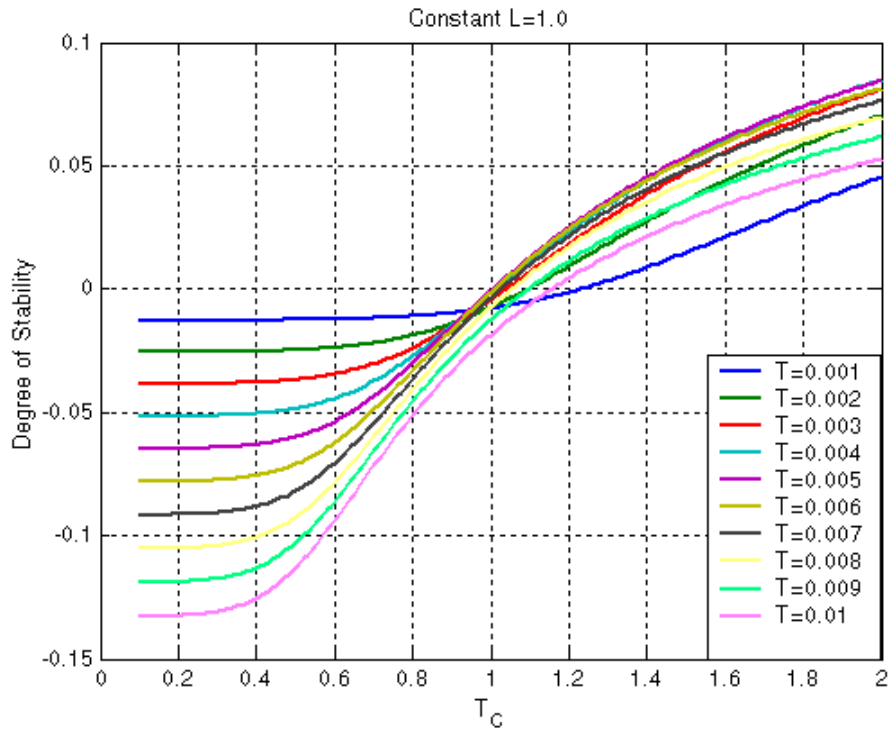


Fig C.15

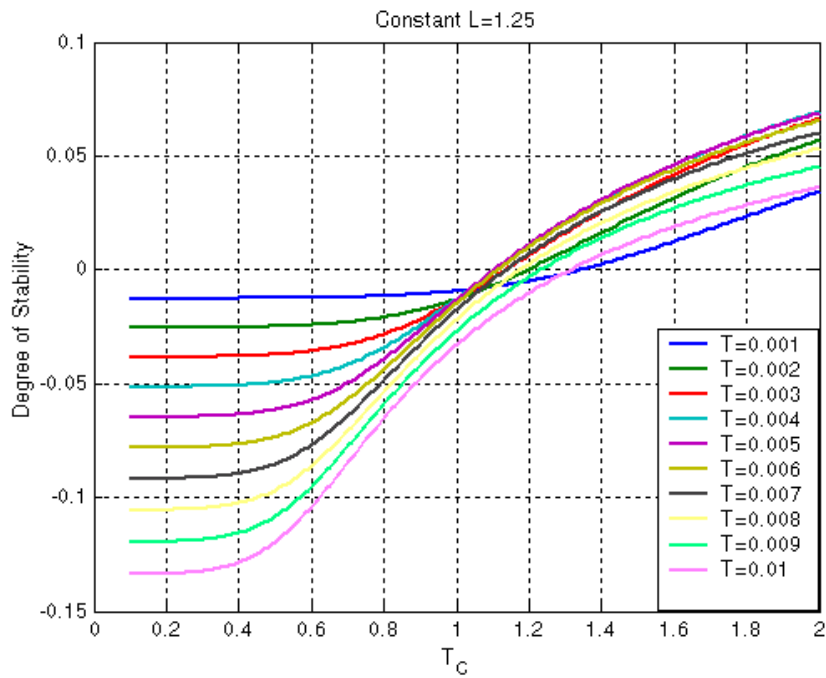


Fig C.16

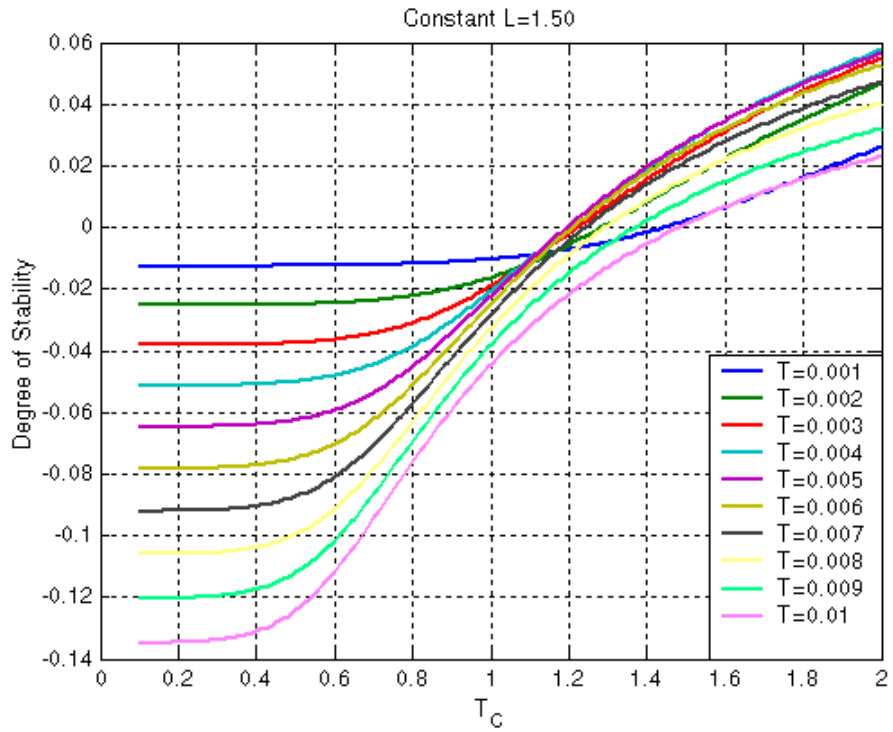


Fig C.17

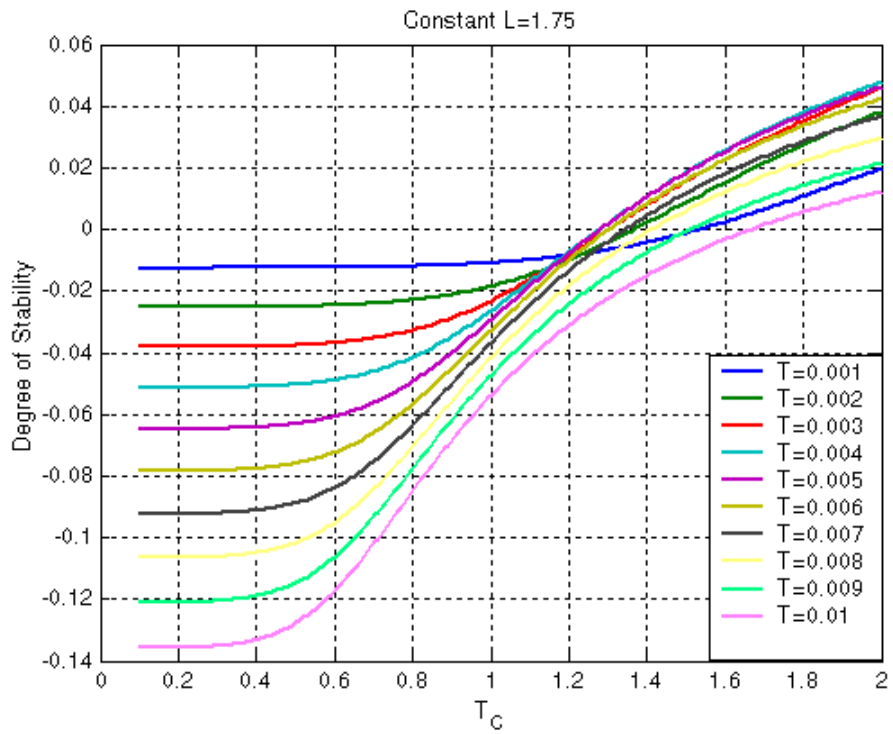


Fig C.18

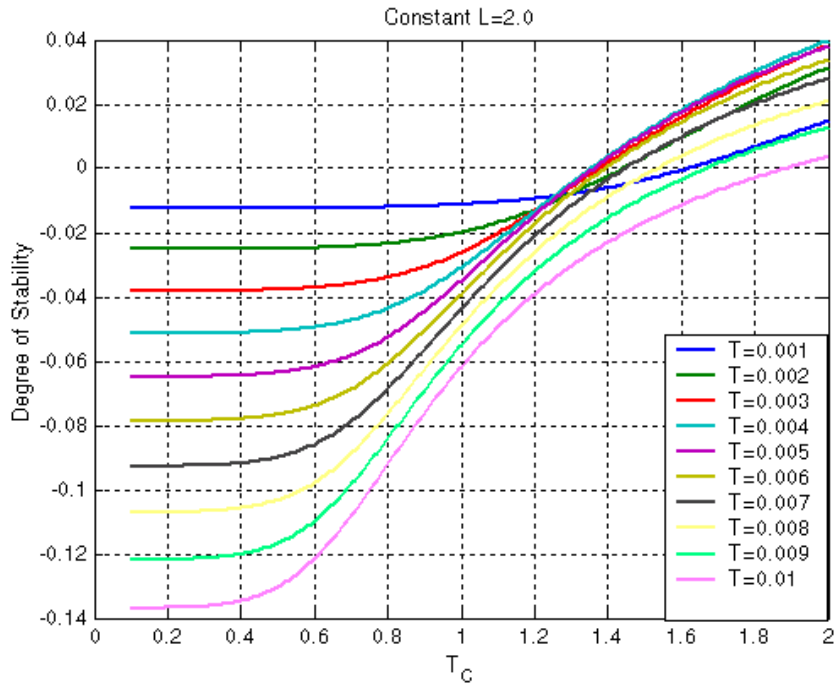


Fig C.19

APPENDIX D. MATLAB CODE FOR STABILITY ANALYSIS WITHOUT CONTROL

```

%LTJG Mersin GOKCE
% Thesis Work
% Model for coupled stability analysis
% Initialization
m1=0.018078; m2=0.018;
u1=1;
u2=1;
T=1;
L=1;
xp1=0.5; xp2=0.5;
Iz1=0.0007; Iz2=0.00069412;
Yv1=-0.07893;
Yv2=-0.1183;
Yr1=-0.004044;
Yr2=-0.0042;
Nv1=-0.016428;
Nv2=-0.0187;
Nr1=-0.010332;
Nr2=-0.0176;
Yvdot1=-0.051328;
Yvdot2=-0.0184;
Yrdot1=0.005617;
Yrdot2=-0.0011;
Nvdot1=-0.001945;
Nvdot2=-0.0008489;
Nrdot1=-0.00564;
Nrdot2=-0.0090;
A3=1/L;
B3=xp1/L;
C3=xp2/L;
%
A2vv=[((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
A2vr=[((Yr2-(m2*u2))*(Iz2-Nrdot2))+(Nr2*Yrdot2)]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
B2v=[-((Iz2-Nrdot2)*T)-(Yrdot2*T*xp2)]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
A2rv=[(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
A2rr=[((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
B2r=[-(T*Nvdot2)-(T*xp2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
Alvv=[((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));
Alvr=[(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
Blv=[((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));
Alrv=[(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));
Alrr=[((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));

```

```

Blr=[(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
A=zeros(7);
A(1,1)=A2vv;
A(1,2)=A2vr;
A(1,5)=B2v/L;
A(1,6)=B2v+(B2v*xp2)/L;
A(1,7)=(B2v*xp1)/L;
A(2,1)=A2rv;
A(2,2)=A2rr;
A(2,5)=B2r/L;
A(2,6)=B2r+(B2r*xp2)/L;
A(2,7)=(B2r*xp1)/L;
A(3,3)=A1vv;
A(3,4)=A1vr;
A(3,5)=B1v/L;
A(3,6)=(B1v*xp2)/L;
A(3,7)=B1v+(B1v*xp1)/L;
A(4,3)=A1rv;
A(4,4)=A1rr;
A(4,5)=B1r/L;
A(4,6)=(B1r*xp2)/L;
A(4,7)=B1r+(B1r*xp1)/L;
A(5,1)=1;
A(5,3)=-1;
A(5,6)=1;
A(5,7)=-1;
A(6,2)=1;
A(7,4)=1;
eig(A)

```

APPENDIX E. MATLAB CODE TO PROVE ZERO EIGENVALUE

```

syms a1 b1 e1 f1 g1
syms a2 b2 e2 f2 g2
syms c3 d3 e3 f3 g3
syms c4 d4 e4 f4 g4
syms h
A=[a1 b1 0 0 e1 f1 g1;
   a2 b2 0 0 e2 f2 g2;
   0 0 c3 d3 e3 f3 g3;
   0 0 c4 d4 e4 f4 g4;
   1 0 -1 0 0 1 -1;
   0 1 0 0 0 0 0;
   0 0 0 1 0 0 0];
B=eye(7).*h;

C=det(A-B);
h=0;
D=subs(C);
syms m1 m2 u1 u2 T L
syms xp1 xp2 Iz1 Iz2
syms Yv1 Yv2 Yr1 Yr2
syms Nv1 Nv2 Nr1 Nr2
syms Yvdot1 Yvdot2 Yrdot1 Yrdot2
syms Nvdot1 Nvdot2 Nrdot1 Nrdot2
syms A2vv A2vr B2v
syms A2rv A2rr B2r
syms Alv Alvr Blv
syms Alrv Alrr Blr
L=1;
xp1=0.5;
xp2=0.5;
A2vv=[((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
A2vr=[(Yr2-m2*u2)*(Iz2-Nrdot2)+(Nr2*Yrdot2)]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
B2v=-[((Iz2-Nrdot2)*T)+(Yrdot2*T*xp2)]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
A2rv=[(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
A2rr=[((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
B2r=-[(T*Nvdot2)+(T*xp2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2));
Alvv=[((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));
Alvr=[(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
Blv=[((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));
Alrv=[(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));
Alrr=[((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
Blr=[(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1));

```

```
a1=A2vv;  
b1=A2vr;  
e1=B2v/L;  
f1=B2v+((B2v*xp2)/L);  
g1=(B2v*xp1)/L;  
a2=A2rv;  
b2=A2rr;  
e2=B2r/L;  
f2=B2r+((B2r*xp2)/L);  
g2=(B2r*xp1)/L;  
c3=A1vv;  
d3=A1vr;  
e3=B1v/L;  
f3=(B1v*xp2)/L;  
g3=B1v+((B1v*xp1)/L);  
c4=A1rv;  
d4=A1rr;  
e4=B1r/L;  
f4=(B1r*xp2)/L;  
g4=B1r+((B1r*xp1)/L);  
E=subs(D)
```

APPENDIX F. MATLAB CODE TO FIND HYDRODYNAMIC COEFFICIENTS

```
% to find Ydel and Ndel
% Ndel=-0.5Ydel
% R=1/(K*del)
% K=[(Nv1*Ydel)-(Yv1*Ndel)]/[(Yv1*Nr1)-(Nv1*(Yr1-(m1*u1)))]
% R=3 (Assumption,tactical diameter)
% del=15 degree= 0.262 radian (Assumption,rudder)
m1=.018078;
u1=1;
Yv1=-0.07893;
Yr1=-0.004044;
Nv1=-0.016428;
Nr1=-0.010332;
R=3;
del=-0.262
Ydel=[Yv1*Nr1-(Nv1*(Yr1-m1))]/[(Nv1+(0.5*Yv1))*R*del]
Ndel=-0.5*Ydel
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX G. MATLAB CODE TO FIND THE COEFFICIENTS OF DELTA

```

% STEP 2
% To Find Coefficients of Delta
m1=.018078;
u1=1;
T=1;
L=1;
xpl=0.5;
Iz1=.0007;
Yv1=-0.07893;
Yr1=-0.004044;
Nv1=-0.016428;
Nr1=-0.010332;
Yvdot1=-0.051328;
Yrdot1=0.005617;
Nvdot1=-0.001945;
Nrdot1=-0.00564;
Ydel=0.0103;
Ndel=-0.0051;
%
%
Alvv=[ ((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
Alvr=[ (Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Blv=[ ((Iz1-Nrdot1)*T)-(Yrdot1*T*xpl)]/[ ((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
Clv=[ ((Iz1-Nrdot1)*Ydel)+(Ndel*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
Alrv=[ (Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
Alrr=[ ((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Blr=[ (T*Nvdot1)-(T*xpl*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
Clr=[ (Ydel*Nvdot1)+(Ndel*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
A=zeros(4);
A(1,3)=1;
A(2,2)=Alvv;
A(2,3)=Alvr;
A(3,2)=Alrv;
A(3,3)=Alrr;
A(4,1)=1;
A(4,2)=1;
B=zeros(4,1);
B(2,1)=Clv;
B(3,1)=Clr;
poles=[-0.1 -0.11 -0.12 -0.13];
k=place(A,B,poles)

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX H. MATLAB CODE FOR STABILITY ANALYSIS AND SIMULATION WITH CONTROL

```

% Main program
% Eigenvalue analysis and numerical simulation
%
% Parameters:
%   T       = Nondimensional tension
%   L       = Towline length
%   TC      = Control law time constant
%   xp1     = Attachment point (leading ship - positive forward of
amidships)
%   xp2     = Attachment point (trailing ship - positive aft of
amidships)
%   DeltaT  = Time step increment
%   SimTime = Simulation time
%
T       = 0.0075;
L       = 0.201;
TC      = 0.6;
bpole   = -1/TC;
xp1     = 0.5;
xp2     = 0.5;
DeltaT  = 0.002;
SimTime = 200;
NPrint  = 10;           % Print out every (NPrint) points
%
% Initial conditions for simulation
% y1,y2 must be consistent with L
%
psi1_old=0;v1_old=0;r1_old=0;y1_old=0.01;
psi2_old=0;v2_old=0;r2_old=0;y2_old=0.00;
%
% Constants
%
u1      = 1;
u2      = 1;
m1      = 0.018078;
m2      = 0.018;
Iz1     = 0.0007;
Iz2     = 0.00069412;
Yv1     = -0.07893;
Yv2     = -0.1183;
Yr1     = -0.004044;
Yr2     = -0.0042;
Nv1     = -0.016428;
Nv2     = -0.0187;
Nr1     = -0.010332;
Nr2     = -0.0176;
Yvdot1 = -0.051328;
Yvdot2 = -0.0184;
Yrdot1 = 0.005617;
Yrdot2 = -0.0011;
Nvdot1 = -0.001945;
Nvdot2 = -0.0008489;
Nrdot1 = -0.00564;
Nrdot2 = -0.0090;

```

```

Ydel = 0.0103;
Ndel = -0.0051;
A3 = 1/L;
B3 = xp1/L;
C3 = xp2/L;
D3 = -1/L;
%
% Setup the matrix coefficients
%
A2vv = [((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/[((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2)];
A2vr = [((Yr2-(m2*u2))*(Iz2-Nrdot2))+(Nr2*Yrdot2)]/[((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
B2v = [ -((Iz2-Nrdot2)*T)-(Yrdot2*T*xp2)]/[((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2)];
A2rv = [(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/[((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2)];
A2rr = [((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/[((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
B2r = [-(T*Nvdot2)-(T*xp2*(m2-Yvdot2))]/[((m2-Yvdot2)*(Iz2-Nrdot2))-
(Nvdot2*Yrdot2)];
A1vv = [((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
A1vr = [(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/[((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
B1v = [((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
C1v = [((Iz1-Nrdot1)*Ydel)+(Ndel*Yrdot1)]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
A1rv = [(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
A1rr = [((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/[((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
B1r = [(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
C1r = [(Ydel*Nvdot1)+(Ndel*(m1-Yvdot1))]/[((m1-Yvdot1)*(Iz1-Nrdot1))-
(Nvdot1*Yrdot1)];
%
% Find control gains
%
C = zeros(4);
C(1,3) = 1;
C(2,2) = Alvv;
C(2,3) = Alvr;
C(3,2) = Alrv;
C(3,3) = Alrr;
C(4,1) = 1;
C(4,2) = 1;
D = zeros(4,1);
D(2,1) = C1v;
D(3,1) = C1r;
poles = [bpole bpole-0.05 bpole-0.10 bpole-0.15];
k = place(C,D,poles);
Kpsi = k(1,1);
Kv = k(1,2);
Kr = k(1,3);
Ky = k(1,4);

```

```

%
% A matrix
%
A      = zeros(8);
A(1,1) = A2vv;
A(1,2) = A2vr;
A(1,5) = B2v/L;
A(1,6) = -B2v/L;
A(1,7) = B2v+((B2v*xp2)/L);
A(1,8) = (B2v*xp1)/L;
A(2,1) = A2rv;
A(2,2) = A2rr;
A(2,5) = B2r/L;
A(2,6) = -B2r/L;
A(2,7) = B2r+((B2r*xp2)/L);
A(2,8) = (B2r*xp1)/L;
A(3,3) = Alv-(C1v*Kv);
A(3,4) = Alvr-(C1v*Kr);
A(3,5) = B1v/L;
A(3,6) = -B1v/L-(C1v*Ky);
A(3,7) = (B1v*xp2)/L;
A(3,8) = B1v+((B1v*xp1)/L)-(C1v*Kpsi);
A(4,3) = Alrv-(C1r*Kv);
A(4,4) = Alrr-(C1r*Kr);
A(4,5) = B1r/L;
A(4,6) = -B1r/L-(C1r*Ky);
A(4,7) = (B1r*xp2)/L;
A(4,8) = B1r+((B1r*xp1)/L)-(C1r*Kpsi);
A(5,1) = 1;
A(5,7) = 1;
A(6,3) = 1;
A(6,8) = 1;
A(7,2) = 1;
A(8,4) = 1;
%
% Find eigenvalues
%
B=eig(A)
%
% Start simulation (Euler fixed step)
%
NT=SimTime/DeltaT;
iPrint=1;iStore=1;
for i=1:NT,
    %
    % Rudder angle
    %
    delta = - ( Kpsi*psil_old + Kv*v1_old + Kr*r1_old + Ky*y1_old );
    %
    % Limit rudder angle between -0.4 and +0.4 radians
    %
    if delta > 0.4
        delta=0.4;
    end
    if delta < -0.4
        delta=-0.4;
    end
end

```

```

%
% Calculate angle gamma
%
gamma=asin((y2_old+xp2*sin(psi2_old)-y1_old+xp1*sin(psi1_old))/L);
%
% Calculate xdot=f(x)
%
psildot = r1_old;
vldot   = Alvv*v1_old   + Alvr*r1_old   + Clv*delta   +
Blv*sin(gamma+psi1_old);
rldot   = Alrr*r1_old   + Alrv*v1_old   + Clr*delta   +
Blr*sin(gamma+psi1_old);
yldot   = u1*sin(psi1_old) + v1_old*cos(psi1_old);
psi2dot = r2_old;
v2dot   = A2vv*v2_old + A2vr*r2_old + B2v*sin(gamma+psi2_old);
r2dot   = A2rr*r2_old + A2rv*v2_old + B2r*sin(gamma+psi2_old);
y2dot   = u2*sin(psi2_old) + v2_old*cos(psi2_old);
%
% Calculate new x = old x + Dt * xdot
%
psi1_new = psi1_old + DeltaT * psildot;
v1_new   = v1_old   + DeltaT * vldot;
r1_new   = r1_old   + DeltaT * rldot;
y1_new   = y1_old   + DeltaT * yldot;
psi2_new = psi2_old + DeltaT * psi2dot;
v2_new   = v2_old   + DeltaT * v2dot;
r2_new   = r2_old   + DeltaT * r2dot;
y2_new   = y2_old   + DeltaT * y2dot;
%
% Ensure psi is between 2pi and -2pi
%
if psi1_new > (2*pi)
    psi1_new=psi1_new-2*pi;
end
if psi2_new > (2*pi)
    psi2_new=psi2_new-2*pi;
end
if psi1_new < (-2*pi)
    psi1_new=psi1_new+2*pi;
end
if psi2_new < (-2*pi)
    psi2_new=psi2_new+2*pi;
end
%
% Update x
%
psi1_old=psi1_new;v1_old=v1_new;r1_old=r1_new;y1_old=y1_new;
psi2_old=psi2_new;v2_old=v2_new;r2_old=r2_new;y2_old=y2_new;
%
% Store results every NPrint time steps
%
iPrint=iPrint+1;
if iPrint > NPrint
    iStore      = iStore+1;
    time(iStore) = i*DeltaT;
    del(iStore)  = delta;
    gam(iStore)  = gamma;
end

```

```

        psi1(iStore) = psi1_new;
        psi2(iStore) = psi2_new;
        v1(iStore)   = v1_new;
        v2(iStore)   = v2_new;
        r1(iStore)   = r1_new;
        r2(iStore)   = r2_new;
        y1(iStore)   = y1_new;
        y2(iStore)   = y2_new;
        iPrint       = 1;
    end
end
%
% Results
%
figure(1)
plot(time,y1,time,y2);legend('y1','y2');xlabel('y');ylabel('y');grid
figure(2)
plot(time,gam*57.2958);xlabel('t');ylabel('\gamma (deg)');grid
figure(3)
plot(time,del*57.2958);xlabel('t');ylabel('\delta (deg)');grid
figure(4)
plot(time,psi1*57.2958);xlabel('t');ylabel('\psi_1 (deg)');grid
figure(5)
plot(time,psi2*57.2958);xlabel('t');ylabel('\psi_2 (deg)');grid

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX I. MATLAB CODE TO FIND THE DEGREE OF STABILITY FOR DIFFERENT TOWLINE LENGTHS

```

% Main program for Length
% Eigenvalue analysis and numerical simulation
%
% Parameters:
%   T      = Nondimensional tension
%   L      = Towline length
%   TC     = Control law time constant
%   xp1    = Attachment point (leading ship - positive forward of
amidships)
%   xp2    = Attachment point (trailing ship - positive aft of
amidships)
%
% Constants
%
u1      = 1;
u2      = 1;
m1      = 0.018078;
m2      = 0.018;
Iz1     = 0.0007;
Iz2     = 0.00069412;
Yv1     = -0.07893;
Yv2     = -0.1183;
Yr1     = -0.004044;
Yr2     = -0.0042;
Nv1     = -0.016428;
Nv2     = -0.0187;
Nr1     = -0.010332;
Nr2     = -0.0176;
Yvdot1 = -0.051328;
Yvdot2 = -0.0184;
Yrdot1 = 0.005617;
Yrdot2 = -0.0011;
Nvdot1 = -0.001945;
Nvdot2 = -0.0008489;
Nrdot1 = -0.00564;
Nrdot2 = -0.0090;
Ydel    = 0.0103;
Ndel    = -0.0051;
T       = 0.01;
TC      = 2.0;
bpole   = -1/TC;
xp1     = 0.5;
xp2     = 0.5;
index=0;
for i =0.1:0.01:2.0;
    index=index+1;
    L      = i;
    A3     = 1/L;
    B3     = xp1/L;
    C3     = xp2/L;
    D3     = -1/L;
    % Setup the matrix coefficients
    %

```

```

    A2vv = [((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    A2vr = [((Yr2-(m2*u2))*(Iz2-Nrdot2))+(Nr2*Yrdot2)]/(((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2)];
    B2v = [-(Iz2-Nrdot2)*T)-(Yrdot2*T*xp2)]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    A2rv = [(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    A2rr = [((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/(((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2)];
    B2r = [-(T*Nvdot2)-(T*xp2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    Alv = [((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Alvr = [(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Blv = [((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Clv = [((Iz1-Nrdot1)*Ydel)+(Ndel*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Alrv = [(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Alrr = [((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/(((m1-
Yvdot1)*(Iz1-Nrdot1))-(Nvdot1*Yrdot1)];
    Blr = [(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Clr = [(Ydel*Nvdot1)+(Ndel*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    %
    % Find control gains
    %
    C = zeros(4);
    C(1,3) = 1;
    C(2,2) = Alv;
    C(2,3) = Alvr;
    C(3,2) = Alrv;
    C(3,3) = Alrr;
    C(4,1) = 1;
    C(4,2) = 1;
    D = zeros(4,1);
    D(2,1) = Clv;
    D(3,1) = Clr;
    poles = [bpole bpole-0.05 bpole-0.10 bpole-0.15];
    k = place(C,D,poles);
    Kpsi = k(1,1);
    Kv = k(1,2);
    Kr = k(1,3);
    Ky = k(1,4);
    %
    % A matrix
    %
    A = zeros(8);
    A(1,1) = A2vv;
    A(1,2) = A2vr;
    A(1,5) = B2v/L;
    A(1,6) = -B2v/L;
    A(1,7) = B2v+((B2v*xp2)/L);

```

```

A(1,8) = (B2v*xp1)/L;
A(2,1) = A2rv;
A(2,2) = A2rr;
A(2,5) = B2r/L;
A(2,6) = -B2r/L;
A(2,7) = B2r+(B2r*xp2)/L;
A(2,8) = (B2r*xp1)/L;
A(3,3) = Alvv-(Clv*Kv);
A(3,4) = Alvr-(Clv*Kr);
A(3,5) = B1v/L;
A(3,6) = -B1v/L-(Clv*Ky);
A(3,7) = (B1v*xp2)/L;
A(3,8) = B1v+((B1v*xp1)/L)-(Clv*Kpsi);
A(4,3) = Alrv-(Clr*Kv);
A(4,4) = Alrr-(Clr*Kr);
A(4,5) = B1r/L;
A(4,6) = -B1r/L-(Clr*Ky);
A(4,7) = (B1r*xp2)/L;
A(4,8) = B1r+((B1r*xp1)/L)-(Clr*Kpsi);
A(5,1) = 1;
A(5,7) = 1;
A(6,3) = 1;
A(6,8) = 1;
A(7,2) = 1;
A(8,4) = 1;
%
% Find eigenvalues
%
B=eig(A);
F=real(B);
H(index)=max(F);
Lv(index)=L;
end

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX J. MATLAB CODE TO FIND THE DEGREE OF STABILITY FOR DIFFERENT TENSIONS

```

% Main program for Tension
% Eigenvalue analysis and numerical simulation
%
% Parameters:
%   T      = Nondimensional tension
%   L      = Towline length
%   TC     = Control law time constant
%   xp1    = Attachment point (leading ship - positive forward of
amidships)
%   xp2    = Attachment point (trailing ship - positive aft of
amidships)
%
% Constants
%
u1      = 1;
u2      = 1;
m1      = 0.018078;
m2      = 0.018;
Iz1     = 0.0007;
Iz2     = 0.00069412;
Yv1     = -0.07893;
Yv2     = -0.1183;
Yr1     = -0.004044;
Yr2     = -0.0042;
Nv1     = -0.016428;
Nv2     = -0.0187;
Nr1     = -0.010332;
Nr2     = -0.0176;
Yvdot1 = -0.051328;
Yvdot2 = -0.0184;
Yrdot1 = 0.005617;
Yrdot2 = -0.0011;
Nvdot1 = -0.001945;
Nvdot2 = -0.0008489;
Nrdot1 = -0.00564;
Nrdot2 = -0.0090;
Ydel    = 0.0103;
Ndel    = -0.0051;
L       = 2.0;
TC      = 2.0;
bpole   = -1/TC;
xp1     = 0.5;
xp2     = 0.5;
A3      = 1/L;
B3      = xp1/L;
C3      = xp2/L;
D3      = -1/L;
index=0;
for i =0.001:0.001:0.01;
    index=index+1;
    T      = i;
    % Setup the matrix coefficients
    %

```

```

    A2vv      =      [ ((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    A2vr      =      [ ((Yr2-(m2*u2))*(Iz2-Nrdot2))+(Nr2*Yrdot2)]/[ ((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2)];
    B2v       =      [ -((Iz2-Nrdot2)*T)-(Yrdot2*T*xp2)]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    A2rv      =      [(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    A2rr      =      [ ((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/[ ((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2)];
    B2r       =      [ -(T*Nvdot2)-(T*xp2*(m2-Yvdot2))]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
    Alv       =      [ ((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Alvr      =      [(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Blv       =      [ ((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Clv       =      [ ((Iz1-Nrdot1)*Ydel)+(Ndel*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Alrv      =      [(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Alrr      =      [ ((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/[ ((m1-
Yvdot1)*(Iz1-Nrdot1))-(Nvdot1*Yrdot1)];
    Blr       =      [(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    Clr       =      [(Ydel*Nvdot1)+(Ndel*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
    %
    % Find control gains
    %
    C         = zeros(4);
    C(1,3)   = 1;
    C(2,2)   = Alv;
    C(2,3)   = Alvr;
    C(3,2)   = Alrv;
    C(3,3)   = Alrr;
    C(4,1)   = 1;
    C(4,2)   = 1;
    D         = zeros(4,1);
    D(2,1)   = Clv;
    D(3,1)   = Clr;
    poles    = [bpole bpole-0.05 bpole-0.10 bpole-0.15];
    k         = place(C,D,poles);
    Kpsi     = k(1,1);
    Kv       = k(1,2);
    Kr       = k(1,3);
    Ky       = k(1,4);
    %
    % A matrix
    %
    A         = zeros(8);
    A(1,1)   = A2vv;
    A(1,2)   = A2vr;
    A(1,5)   = B2v/L;
    A(1,6)   = -B2v/L;
    A(1,7)   = B2v+((B2v*xp2)/L);

```

```

A(1,8) = (B2v*xp1)/L;
A(2,1) = A2rv;
A(2,2) = A2rr;
A(2,5) = B2r/L;
A(2,6) = -B2r/L;
A(2,7) = B2r+(B2r*xp2)/L;
A(2,8) = (B2r*xp1)/L;
A(3,3) = Alvv-(Clv*Kv);
A(3,4) = Alvr-(Clv*Kr);
A(3,5) = B1v/L;
A(3,6) = -B1v/L-(Clv*Ky);
A(3,7) = (B1v*xp2)/L;
A(3,8) = B1v+((B1v*xp1)/L)-(Clv*Kpsi);
A(4,3) = Alrv-(Clr*Kv);
A(4,4) = Alrr-(Clr*Kr);
A(4,5) = B1r/L;
A(4,6) = -B1r/L-(Clr*Ky);
A(4,7) = (B1r*xp2)/L;
A(4,8) = B1r+((B1r*xp1)/L)-(Clr*Kpsi);
A(5,1) = 1;
A(5,7) = 1;
A(6,3) = 1;
A(6,8) = 1;
A(7,2) = 1;
A(8,4) = 1;
%
% Find eigenvalues
%
B=eig(A);
F=real(B);
H(index)=max(F);
Tv(index)=T;
end

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX K. MATLAB CODE TO FIND THE DEGREE OF STABILITY FOR DIFFERENT TIME CONSTANTS

```

% Main program Time Constant
% Eigenvalue analysis and numerical simulation
%
% Parameters:
%   T      = Nondimensional tension
%   L      = Towline length
%   TC     = Control law time constant
%   xp1    = Attachment point (leading ship - positive forward of
amidships)
%   xp2    = Attachment point (trailing ship - positive aft of
amidships)
%
% Constants
%
u1      = 1;
u2      = 1;
m1      = 0.018078;
m2      = 0.018;
Iz1     = 0.0007;
Iz2     = 0.00069412;
Yv1     = -0.07893;
Yv2     = -0.1183;
Yr1     = -0.004044;
Yr2     = -0.0042;
Nv1     = -0.016428;
Nv2     = -0.0187;
Nr1     = -0.010332;
Nr2     = -0.0176;
Yvdot1  = -0.051328;
Yvdot2  = -0.0184;
Yrdot1  = 0.005617;
Yrdot2  = -0.0011;
Nvdot1  = -0.001945;
Nvdot2  = -0.0008489;
Nrdot1  = -0.00564;
Nrdot2  = -0.0090;
Ydel    = 0.0103;
Ndel    = -0.0051;
T       = 0.01;
L       = 2.0;
xp1     = 0.5;
xp2     = 0.5;
A3      = 1/L;
B3      = xp1/L;
C3      = xp2/L;
D3      = -1/L;
index=0;
for i =0.1:0.01:2.0;
    index=index+1;
    TC =i;
    bpole = -1/TC;
    % Setup the matrix coefficients
    %

```

```

A2vv      =      [ ((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
A2vr      =      [ ((Yr2-(m2*u2))*(Iz2-Nrdot2))+(Nr2*Yrdot2)]/[ ((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2)];
B2v       =      [ -((Iz2-Nrdot2)*T)-(Yrdot2*T*xp2)]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
A2rv      =      [(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
A2rr      =      [ ((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/[ ((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2)];
B2r       =      [ -(T*Nvdot2)-(T*xp2*(m2-Yvdot2))]/[ ((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2)];
Alvv      =      [ ((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Alvr      =      [(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Blv       =      [ ((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Clv       =      [ ((Iz1-Nrdot1)*Ydel)+(Ndel*Yrdot1)]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Alrv      =      [(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Alrr      =      [ ((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/[ ((m1-
Yvdot1)*(Iz1-Nrdot1))-(Nvdot1*Yrdot1)];
Blr       =      [(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
Clr       =      [(Ydel*Nvdot1)+(Ndel*(m1-Yvdot1))]/[ ((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1)];
%
% Find control gains
%
C         = zeros(4);
C(1,3)   = 1;
C(2,2)   = Alvv;
C(2,3)   = Alvr;
C(3,2)   = Alrv;
C(3,3)   = Alrr;
C(4,1)   = 1;
C(4,2)   = 1;
D         = zeros(4,1);
D(2,1)   = Clv;
D(3,1)   = Clr;
poles    = [bpole bpole-0.05 bpole-0.10 bpole-0.15];
k        = place(C,D,poles);
Kpsi     = k(1,1);
Kv       = k(1,2);
Kr       = k(1,3);
Ky       = k(1,4);
%
% A matrix
%
A         = zeros(8);
A(1,1)   = A2vv;
A(1,2)   = A2vr;
A(1,5)   = B2v/L;
A(1,6)   = -B2v/L;
A(1,7)   = B2v+((B2v*xp2)/L);

```

```

A(1,8) = (B2v*xp1)/L;
A(2,1) = A2rv;
A(2,2) = A2rr;
A(2,5) = B2r/L;
A(2,6) = -B2r/L;
A(2,7) = B2r+(B2r*xp2)/L;
A(2,8) = (B2r*xp1)/L;
A(3,3) = Alvv-(Clv*Kv);
A(3,4) = Alvr-(Clv*Kr);
A(3,5) = B1v/L;
A(3,6) = -B1v/L-(Clv*Ky);
A(3,7) = (B1v*xp2)/L;
A(3,8) = B1v+((B1v*xp1)/L)-(Clv*Kpsi);
A(4,3) = Alrv-(Clr*Kv);
A(4,4) = Alrr-(Clr*Kr);
A(4,5) = B1r/L;
A(4,6) = -B1r/L-(Clr*Ky);
A(4,7) = (B1r*xp2)/L;
A(4,8) = B1r+((B1r*xp1)/L)-(Clr*Kpsi);
A(5,1) = 1;
A(5,7) = 1;
A(6,3) = 1;
A(6,8) = 1;
A(7,2) = 1;
A(8,4) = 1;
%
% Find eigenvalues
%
B=eig(A);
F=real(B);
H(index)=max(F);
TCv(index)=TC;
end

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX L. MATLAB CODE TO FIND THE REGION OF STABILITY

```
% Stability graph - TC vs. L - constant T
%
% Parameters:
%   T      = Nondimensional tension
%   L      = Towline length
%   TC     = Control law time constant
%   xp1    = Attachment point (leading ship - positive forward of
amidships)
%   xp2    = Attachment point (trailing ship - positive aft of
amidships)
%
% Constants
%
u1      = 1;
u2      = 1;
m1      = 0.018078;
m2      = 0.018;
Iz1     = 0.0007;
Iz2     = 0.00069412;
Yv1     = -0.07893;
Yv2     = -0.1183;
Yr1     = -0.004044;
Yr2     = -0.0042;
Nv1     = -0.016428;
Nv2     = -0.0187;
Nr1     = -0.010332;
Nr2     = -0.0176;
Yvdot1 = -0.051328;
Yvdot2 = -0.0184;
Yrdot1 = 0.005617;
Yrdot2 = -0.0011;
Nvdot1 = -0.001945;
Nvdot2 = -0.0008489;
Nrdot1 = -0.00564;
Nrdot2 = -0.0090;
Ydel    = 0.0103;
Ndel    = -0.0051;
indexL  = 0;
index   = 0;
xp1     = 0.5;
xp2     = 0.5;
%
% Enter constant T
%
T       = input('T = ');
%
% Start loop on length
%
for iL = 0.1:0.01:2.0;
    indexL    = indexL+1;
    indexTC   = 0;
    L         = iL
    L_v(indexL) = L;
    A3        = 1/L;
```

```

B3      = xp1/L;
C3      = xp2/L;
D3      = -1/L;
%
% Loop on TC
%
for iTC = 0.1:0.01:2.0;
    indexTC=indexTC+1;
    TC      = iTC;
    bpole  = -1/TC;
    TC_v(indexTC)=TC;
    %
    % Setup the matrix coefficients
    %
    A2vv   = [((Iz2-Nrdot2)*Yv2)+(Nv2*Yrdot2)]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
    A2vr   = [((Yr2-(m2*u2))*(Iz2-Nrdot2))+(Nr2*Yrdot2)]/(((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2));
    B2v    = [(-(Iz2-Nrdot2)*T)-(Yrdot2*T*xp2)]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
    A2rv   = [(Yv2*Nvdot2)+(Nv2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
    A2rr   = [((Yr2-(m2*u2))*Nvdot2)+(Nr2*(m2-Yvdot2))]/(((m2-
Yvdot2)*(Iz2-Nrdot2))-(Nvdot2*Yrdot2));
    B2r    = [-(T*Nvdot2)-(T*xp2*(m2-Yvdot2))]/(((m2-Yvdot2)*(Iz2-
Nrdot2))-(Nvdot2*Yrdot2));
    Alv    = [((Iz1-Nrdot1)*Yv1)+(Nv1*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
    Alvr   = [(Yr1-m1*u1)*(Iz1-Nrdot1)+(Nr1*Yrdot1)]/(((m1-
Yvdot1)*(Iz1-Nrdot1))-(Nvdot1*Yrdot1));
    Blv    = [((Iz1-Nrdot1)*T)-(Yrdot1*T*xp1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
    Clv    = [((Iz1-Nrdot1)*Ydel)+(Ndel*Yrdot1)]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
    Alrv   = [(Yv1*Nvdot1)+(Nv1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
    Alrr   = [((Yr1-(m1*u1))*Nvdot1)+(Nr1*(m1-Yvdot1))]/(((m1-
Yvdot1)*(Iz1-Nrdot1))-(Nvdot1*Yrdot1));
    Blr    = [(T*Nvdot1)-(T*xp1*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
    Clr    = [(Ydel*Nvdot1)+(Ndel*(m1-Yvdot1))]/(((m1-Yvdot1)*(Iz1-
Nrdot1))-(Nvdot1*Yrdot1));
    %
    % Find control gains
    %
    C      = zeros(4);
    C(1,3) = 1;
    C(2,2) = Alv;
    C(2,3) = Alvr;
    C(3,2) = Alrv;
    C(3,3) = Alrr;
    C(4,1) = 1;
    C(4,2) = 1;
    D      = zeros(4,1);
    D(2,1) = Clv;
    D(3,1) = Clr;
    poles = [bpole bpole-0.05 bpole-0.10 bpole-0.15];

```

```

k      = place(C,D,poles);
Kpsi  = k(1,1);
Kv    = k(1,2);
Kr    = k(1,3);
Ky    = k(1,4);
%
% A matrix
%
A      = zeros(8);
A(1,1) = A2vv;
A(1,2) = A2vr;
A(1,5) = B2v/L;
A(1,6) = -B2v/L;
A(1,7) = B2v+((B2v*xp2)/L);
A(1,8) = (B2v*xp1)/L;
A(2,1) = A2rv;
A(2,2) = A2rr;
A(2,5) = B2r/L;
A(2,6) = -B2r/L;
A(2,7) = B2r+((B2r*xp2)/L);
A(2,8) = (B2r*xp1)/L;
A(3,3) = A1vv-(C1v*Kv);
A(3,4) = A1vr-(C1v*Kr);
A(3,5) = B1v/L;
A(3,6) = -B1v/L-(C1v*Ky);
A(3,7) = (B1v*xp2)/L;
A(3,8) = B1v+((B1v*xp1)/L)-(C1v*Kpsi);
A(4,3) = A1rv-(C1r*Kv);
A(4,4) = A1rr-(C1r*Kr);
A(4,5) = B1r/L;
A(4,6) = -B1r/L-(C1r*Ky);
A(4,7) = (B1r*xp2)/L;
A(4,8) = B1r+((B1r*xp1)/L)-(C1r*Kpsi);
A(5,1) = 1;
A(5,7) = 1;
A(6,3) = 1;
A(6,8) = 1;
A(7,2) = 1;
A(8,4) = 1;
%
% Find eigenvalues
%
B=eig(A);
F=real(B);
H_new=max(F);
%
% Detect if H changes its sign
%
if indexTC==1
    H_old=H_new;
else
    PROD=H_new*H_old;
    if PROD > 0
        %
        % H does not change sign - keep going
        %
        H_old=H_new;
    end
end

```

```

        else
            %
            % H changed its sign - find critical TC by linear
interpolation
            %
            index=index+1;
            TC_crit(index)=- (H_old*TC_v(indexTC)-
H_new*TC_v(indexTC-1))/(H_new-H_old);
            H_old=H_new;
        end
    end
end
end
%
% Plot (L, crit TC) curve
%
%plot(L_v,TC_crit),xlabel('L'),ylabel('{T_C}_{\rm
crit}'),title('Constant T'),grid

```

APPENDIX M. MATLAB CODE TO EVALUATE THE STABILITY CRITERIA BASED ON ONLY TRAILING SHIP

```

%Check the Stability Criteria in the paper
%For the Second Ship
%First Criteria  $x_{p2} > N_{v2}/Y_{v2}$ 
%Second Criteria  $T > -(Alfa2/Alfa1)$ 
m2=.018;
u2=1;
T=0.01;
L=1.8541;
xp2=0.5;
Iz2=.00069412;
Yv2=-0.1183;
Yr2=-0.0042;
Nv2=-0.0187;
Nr2=-0.0176;
Yvdot2=-0.0184;
Yrdot2=-0.0011;
Nvdot2=-0.0008489;
Nrdot2=-0.0090;
Xp2=(Nv2/Yv2)
A=[ ((Yvdot2-m2)*(Nrdot2-Iz2))-(Nvdot2*Yrdot2)]; %A0
B=[ ((Yvdot2-m2)*Nr2)+(Yv2*(Nrdot2-Iz2))+(Nvdot2*(m2-Yr2))-
(Yrdot2*Nv2)]; %B0
C=[ (Yv2*Nr2)+(Nv2*(m2-Yr2))]; %C0
C1=[ ((m2-Yvdot2)*xp2)+Nvdot2];
C2=[ -(Nrdot2-Iz2)+((m2-Yvdot2)*(xp2^2))+(xp2*(Nvdot2+Yrdot2))];
D1=[Nv2-(Yv2*xp2)];
D2=[ -(Yv2*(xp2^2))+(Nv2*xp2)+((Yr2-Yvdot2)*xp2)-Nr2+Nvdot2];
E1=0;
Alfa1=[ ((B*C1*D1)-(A*(D1^2)))+(1/L)*((B*C1*D2)+(B*C2*D1)-
(2*A*D1*D2)))+(1/(L^2))*((B*C2*D2)-(A*(D2^2)))]
Alfa2=[ (B*C*D1)+(1/L)*((B*C*D2)-(B^2*E1))]
T1=-(Alfa2/Alfa1)

```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. *Principles of Naval Architecture*, v III, Edward V. Lewis editor, Society of Naval Architects and Marine Engineers, 1989
2. Jones, Garrett D., *Semi-rigid Towing Model for Analysis of Maneuvering in the Horizontal Plane*, Thesis, Naval Postgraduate School, CA 2001
3. Bernitsas M.M and Kekridis N.S., *Simulation and Stability of Ship Towing*, ISP, vol 32, no 369, 1985

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, VA
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Chairman
Department of Mechanical Engineering
Naval Postgraduate School
Monterey, CA
4. Engineering and Technology Curricular Office
Naval Postgraduate School
Monterey, CA
5. Professor Fotis A. Papoulias
Naval Postgraduate School
Monterey, CA
6. Deniz Kuvvetleri Komutanligi
Personel Daire Baskanligi
Bakanliklar,Ankara,Turkey
7. LTJG Mersin Gokce
Ataturk Cad. Tabakhane Mah.
Ozgur Apt.Kat:2 D:2
P.K. 41500
Karamursel,Kocaeli,Turkey