1993-12

Hopf bifurcation analysis for depth control of submersible vehicles

Bateman, Craig A.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

HOPF BIFURCATION ANALYSIS FOR DEPTH CONTROL OF SUBMERSIBLE VEHICLES

by

Craig A. Bateman

December, 1993

Thesis Advisor: Fotis A. Papoulias

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	30 November 1993	Master's Thesis

4. TITLE AND SUBTITLE HOPF BIFURCATION ANALYSIS FOR DEPTH CONTROL OF SUBMERSIBLE VEHICLES

5. FUNDING NUMBERS

6. AUTHOR(S) *BATEMAN, Craig A.*

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)

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
*A

13.
ABSTRACT

Control of a modern submarine is a multi-dimensional problem coupling initial stability, hydrodynamic and control system response. The loss of stability at moderate to high speeds is examined using a nonlinear Hopf bifurcation analysis. Complete linear state feedback is used for demonstration purposes for depth control at level attitude and for a fixed nominal speed. Control time constant, nominal and actual speeds, metacentric height, and stern to bow plane deflection ratio are used as the main bifurcation parameters. A complete local bifurcation mapping provides a systematic method for evaluating the bounds of controllability for control system design parameters for a submarine with a given set of hydrodynamic coefficients. The submarine and its potential design modifications are then verified with a nonlinear simulation program.

14. SUBJECT TERMS
Submarine, bifurcation, nonlinear dynamics, chaos

15. NUMBER OF PAGES 94

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI STD. 239-18

Approved for public release; distribution is unlimited.

HOPF BIFURCATION ANALYSIS FOR DEPTH CONTROL OF SUBMERSIBLE VEHICLES

by

Craig A. Bateman
Lieutenant Commander, United States Navy
B.S., University of Michigan, 1979

Submitted in partial fulfillment
of the requirements for the degree of

MASTER'S OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1993

Author: [redacted]
Craig A. Bateman

Approved by: [redacted]
Fotis A. Papoulias, Thesis Advisor

[redacted]
Matthew D. Kelleher, Chairman
Department of Mechanical Engineering

ABSTRACT

Control of a modern submarine is a multi-dimensional problem coupling initial stability, hydrodynamic and control system response. The loss of stability at moderate to high speeds is examined using a nonlinear Hopf bifurcation analysis. Complete linear state feedback is used for demonstration purposes for depth control at level attitude and for a fixed nominal speed. Control time constant, nominal and actual speeds, metacentric height, and stern to bow plane ratio are used as the main bifurcation parameters. A complete local bifurcation mapping provides a systematic method for evaluating the bounds of controllability for control system design parameters for a submarine with a given set of hydrodynamic coefficients. The submarine and its potential design modifications are then verified with a nonlinear simulation program.

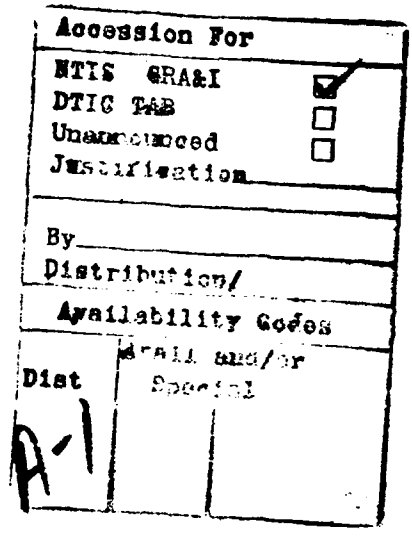

TABLE OF CONTENTS

NOMENCLATURE

Symbol	Definition
A	closed loop dynamics matrix for the linearized system
a	dummy independent variable
α	bow plane to stern plane deflection ratio (or, control surface coordination gain)
$b(x)$	local beam of the hull
B	vehicle buoyancy
C_D	quadratic drag coefficient
c_1, c_2	coupled heave and pitch terms
d_q, d_w	cross flow drag terms
δ_b	bow plane deflection
δ_s, δ	stern plane deflection
δ_{sat}	saturation value of δ
I_y	vehicle mass moment of inertia
k_1, k_2, k_3, k_4	controller gains in θ,w,q, and z, respectively
K	cubic stability coefficient
m	vehicle mass
M	pitch moment
M_a	derivative of M with respect to a
q	pitch rate
θ	vehicle pitch angle
R, θ	polar coordinates of transformed reduced system
T	matrix of eigenvectors of A
T_c	time constant
U	vehicle forward speed
U_0	nominal forward speed
w	heave velocity
W	vehicle weight
$\mathbf{x}$	state variable vector
(x_B, z_B)	body fixed coordinates of vehicle center of buoyancy
(x_G, z_G)	body fixed coordinates of vehicle center of gravity
$\mathbf{z}$	state variables vector in canonical form
z	deviation off the nominal depth
z_{GB}	vehicle metacentric height
Z	heave force
Z_a	derivative of Z with respect to a
ω_0	frequency at the bifurcation point

ACKNOWLEDGEMENT

First and foremost, I would like to give my deepest thanks to my wife, Sara, and my children, David and Kristen, for their patience and support during our two and a half year separation. There is no way that I can repay their sacrifice.

Second, my thanks to LT. Mary M. Zurowski, USN for her careful editing and proofreading of this thesis. Hers was a difficult task.

Finally, my highest appreciation to Professor Fotis A. Papoulias, for his expert guidance and excellent instruction during the course of this thesis.

I. INTRODUCTION

The fundamental goal of submarine control is to reach and maintain ordered depth. Any design that does not meet this goal, in the face of the inherent complexities, is not overly useful as a practical vessel. Current evaluation schemes involve extensive model testing such as those done on the DARPA SUBOFF Model (DTRC Model 5470) [ref. 8]. This is an expensive and time consuming evaluation method. The goal is to develop an analytic method to determine the stability of a potential design. Much work has been done on depth control and modelling of submarines in the vertical plane [ref's. 1, 2, & 3] but this work has assumed the existence of a stable platform.

The stability of a design will have a significant impact on its responsiveness. A vehicle with a large margin to instability will not be very responsive. The problem becomes one of determining how close to the margins we can get without a total loss of stability. Nonlinear dynamics and chaos provides us with the tools for solving this problem [ref's 9, & 10]. By expanding the scope of our research to include nonlinear terms we are able to define the limits of stability and therefore the margins. At the Naval Postgraduate School there has been extensive work on defining the nature of the instabilities that are encountered in the control of

submersibles in path keeping and the dive plane [ref's 4, 5, & 6].

At low forward velocities a submersible using stern planes for attitude and depth control can experience a loss of stability in the form of stern planes reversal. This is a pitchfork bifurcation that can be predicted [ref. 7] and can be accounted for in the control design. The purpose of this thesis is to develop a program for finding the limits of stability for a submarine at moderate and high speeds. Once these limits are mapped then the nature of the loss of stability must then be determined. For this we use a Hopf bifurcation analysis along with a nonlinear simulation program. Finally, after the limits are determined we are able to define the control system design parameters and evaluate the controllability of the design.

II. PROBLEM FORMULATION

A. EQUATIONS OF MOTION

By restricting the submersible's motion to the vertical, or dive plane, the motion can be modeled by the coupled nonlinear equations for pitch and heave. With a body fixed coordinate frame at the vehicles geometric center, we can express Newton's equations of motion as

$$\begin{aligned} m(\dot{w} - Uq - z_G q^2 - x_G \dot{q}) &= Z_{\dot{q}}\dot{q} + Z_{\dot{w}}\dot{w} + Z_q Uq + Z_w Uw \\ &\quad - C_D \int_{TAIL}^{NOSE} b(x) \frac{(w - xq)^3}{|w - xq|} dx \\ &\quad (W - B)\cos\theta + U^2 (Z_{\delta_s}\delta_s + Z_{\delta_b}\delta_b), \end{aligned} \tag{2.1}$$

and

$$\begin{aligned} I_y \dot{q} + m z_G w q - m x_G (\dot{w} - Uq) &= M_{\dot{q}}\dot{q} + M_{\dot{w}}\dot{w} + M_q Uq + M_w Uw \\ &\quad + C_D \int_{TAIL}^{NOSE} b(x) \frac{(w - xq)^3}{|w - xq|} x dx \\ &\quad - (x_G W - x_B B)\cos\theta - (z_G W - z_B B)\sin\theta \\ &\quad + U^2 (M_{\delta_s}\delta_s + M_{\delta_b}\delta_b). \end{aligned} \tag{2.2}$$

For simplicity we will assume that the drag coefficient, C_D, is constant over the length of the submersible. This does not have a significant effect on the results.

The pitch rate and depth rate for a submersible is given by

$$\dot{\theta} = q \ , \tag{2.3}$$

$$\dot{z} = -U\sin\theta + w\cos\theta \ , \tag{2.4}$$

respectively. In equations (2.3) and (2.4) θ is the pitch angle of the vehicle as measured from the horizontal plane. Figure (2.1) shows the geometry and definitions for most of the symbols. We assume that forward velocity, U, is maintained constant during maneuvers by an appropriate use of the propulsion system.

B. CONTROL LAW

Equations (2.1) through (2.4) can be written as a set of nonlinear differential equations in the form,

$$\dot{\theta} = q \ , \tag{2.5}$$

$$\begin{aligned} \dot{w} = a_{11}Uw + a_{12}Uq + a_{13}z_{GB}\sin\theta + b_{11}U^2\delta_s + b_{12}U^2\delta_b \\ + d_w(w,q) + c_1(w,q) \ , \end{aligned} \tag{2.6}$$

$$\dot{q} = a_{21}Uw + a_{22}Uq + a_{23}z_{GB}\sin\theta + b_{21}U^2\delta_s + b_{22}U^2\delta_b + d_q(w,q) + c_2(w,q) \quad , \tag{2.7}$$

$$\dot{z} = -U\sin\theta + w\cos\theta \quad , \tag{2.8}$$

where,

$$\begin{aligned}
D_v &= (m-Z_{\dot{w}})(I_y-M_{\dot{q}}) - (mx_G+Z_{\dot{q}})(mx_G+M_{\dot{w}}) \quad , \\
a_{11}D_v &= (I_y-M_{\dot{q}})Z_w + (mx_G+Z_{\dot{q}})M_w \quad , \\
a_{12}D_v &= (I_y-M_{\dot{q}})(m+Z_q) + (mx_G+Z_{\dot{q}})(M_q-mx_G) \quad , \\
a_{13}D_v &= -(mx_G+Z_{\dot{q}})W \quad , \\
b_{11}D_v &= (I_y-M_{\dot{q}})Z_{\delta_s} + (mx_G+Z_{\dot{q}})M_{\delta_b} \quad , \\
b_{12}D_v &= (I_y-M_{\dot{q}})Z_{\delta_b} + (mx_G+Z_{\dot{q}})M_{\delta_b} \quad , \\
a_{21}D_v &= (m-Z_{\dot{w}})M_w + (mx_G+M_{\dot{w}})Z_w \quad , \\
a_{22}D_v &= (m-Z_{\dot{w}})(M_q-mx_G) + (mx_G+M_{\dot{w}})(m+Z_q) \quad , \\
a_{23}D_v &= -(m-Z_{\dot{w}})W \quad , \\
b_{21}D_v &= (m-Z_{\dot{w}})M_{\delta_s} + (mx_G+M_{\dot{w}})Z_{\delta_s} \quad , \\
b_{22}D_v &= (m-Z_{\dot{w}})M_{\delta_b} + (mx_G+M_{\dot{w}})Z_{\delta_b} \quad , \\
d_w(w,q)D_v &= (I_y-M_{\dot{q}})I_w + (mx_G+Z_{\dot{q}})I_q \quad , \\
d_q(w,q)D_v &= (m-Z_{\dot{w}})I_q + (mx_G+M_{\dot{w}})I_w \quad , \\
c_1(w,q)D_v &= (I_y-M_{\dot{q}})mz_Gq^2 - (mx_G+Z_{\dot{q}})mz_Gwq \quad , \\
c_2(w,q)D_v &= -(m-Z_{\dot{w}})mz_Gwq + (mx_G+M_{\dot{w}})mz_Gq^2 \quad .
\end{aligned} \tag{2.9}$$

In equations (2.5) through (2.8), the vehicle is assumed to be neutrally buoyant (W = B), level ($x_G = x_B$), and statically stable ($z_G > z_B$). The terms I_w and I_q represent the cross flow drag integrals in equations (2.1) and (2.2), and $z_{GB} = z_G - z_B$ is the metacentric height. We can assume z_B to be zero, therefore $z_{GB} = z_G$.

The depth control system uses the linearized form of equations (2.5) through (2.8) with the linearization at the

nominal conditions of level attitude, ordered depth and forward speed. This gives the state space form of the depth control equations as

$$\dot{\theta} = q \ , \tag{2.10}$$

$$\dot{w} = a_{11}U_0 w + a_{12}U_0 q + a_{13}z_{GB}\theta + b_1 U_0^2 \delta \ , \tag{2.11}$$

$$\dot{q} = a_{21}U_0 w + a_{22}U_0 q + a_{23}z_{GB}\theta + b_2 U_0^2 \delta \ , \tag{2.12}$$

$$\dot{z} = -U_0\theta + w \ , \tag{2.13}$$

where U_0 is the nominal speed for gain selection, and the control inputs are defined as

$$\begin{aligned} \delta_s &= \delta \ , \\ \delta_b &= \alpha\delta \ , \\ b_1 &= b_{11} + \alpha b_{12} \ , \\ b_2 &= b_{21} + \alpha b_{22} \ , \end{aligned} \tag{2.14}$$

where α is the control surface coordination gain. This produces a linear full state feedback control law of

$$\delta = k_1\theta + k_2 w + k_3 q + k_4 z \ , \tag{2.15}$$

where the gains k_1, k_2, k_3, and k_4 are computed to give the closed loop system, equations (2.10) through (2.13), the

desired dynamics. Since the desired characteristic equation has the general form,

$$\lambda^4+\alpha_3\lambda^3+\alpha_2\lambda^2+\alpha_1\lambda+\alpha_0 = 0 \ , \qquad (2.16)$$

the controller gains are computed by equating coefficients of the actual and desired characteristic equations,

$$\begin{aligned}
&b_1U_0^2k_2+b_2U_0^2k_3 = -\alpha_3-(a_{11}+a_{22})\,U_0 \ , \\
&b_2U_0^2k_1+(b_2a_{12}-b_1a_{22})\,U_0^3k_2+(b_1a_{21}-b_2a_{11})\,U_0^3k_3+b_1U_0^2k_4 \\
&\qquad = -\alpha_2-a_{23}z_{GB}+(a_{11}a_{22}-a_{21}a_{12}) \ , \\
&(b_2a_{11}-b_1a_{21})\,U_0^3k_1+(b_1a_{23}-b_2a_{13})\,z_{GB}U_0^2k_2 \\
&\quad +(b_2+b_1a_{22}-b_2a_{12})\,U_0^3k_4 = \alpha_1+(a_{13}a_{21}-a_{23}a_{11})\,z_{GB}U_0 \ , \\
&[(b_1a_{21}-b_2a_{11})\,U_0^4+(b_1a_{23}-b_2a_{213})\,z_{GB}U_0^2]\,k_4 = \alpha_0 \ .
\end{aligned} \qquad (2.17)$$

[ref. 7]

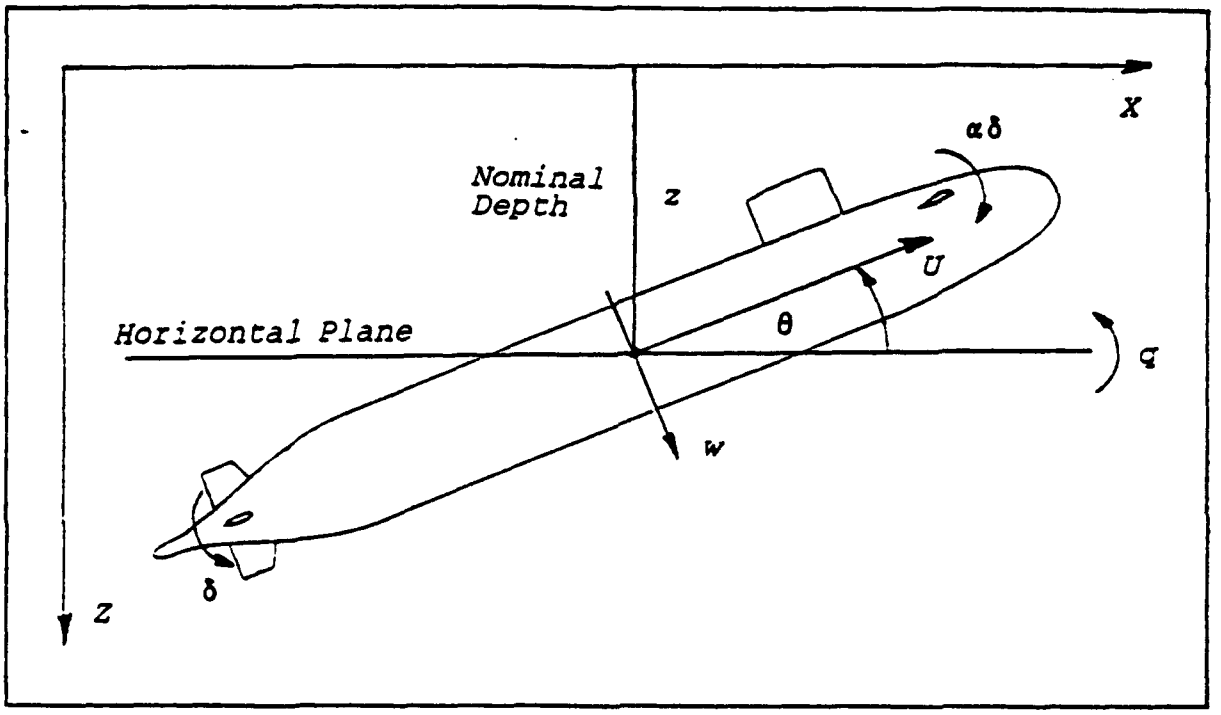

Figure (2.1) - Geometrical representations of the basic definitions for the equations of motion.

III. STABILITY

A. BIFURCATION ANALYSIS

In system dynamics, the classical definition of stability states, that the real part of all the eigenvalues of the system must be negative. Therefore, our initial investigations into the stability of the SUBOFF Model was to find those eigenvalues whose real parts cross the imaginary axis. We used the bifurcation analysis program, included as Appendix A, to calculate the eigenvalues of the system.

By linearizing the equations of motion, equations (2.1 through 2.4), the state space equations of the dynamic system can be written in the form

$$\dot{\boldsymbol{x}}=A\boldsymbol{x}+B\boldsymbol{u} \tag{3.1}$$

where

$$\boldsymbol{u}=-K\boldsymbol{x} \tag{3.2}$$

and K is the matrix of controller gains, as calculated by pole placement in equation (2.17). The eigenvalues of the system are found by solving

$$\det|A-BK-sI|=0. \tag{3.3}$$

In the bifurcation analysis program a pseudo-root locus method is employed where the time constant, T_c, is fixed. The constant T_c fixes to placement of the system poles at a given nominal forward speed U_0 and then the model speed, U, is varied incrementally with the system eigenvalues calculated at each speed increment. When the real part of an eigenvalue changes sign between the limits of a speed increment a bisection method is employed to find the speed where the real part of the eigenvalue equals 0.

For each point where the real part of an eigenvalue crosses the imaginary axis the associated T_c and U are plotted on a bifurcation map. This map delineates the regions of classical stability (all eigenvalues on the left hand half-plane) from the regions of instability (at least one eigenvalue in the right hand half-plane). A family of bifurcation maps were generated by varying nominal speed, U_0, initial stability, Z_{GB}, and control surface gain, α.

B. TYPICAL RESULTS

Figure (3.1) shows a typical bifurcation map with its five distinct regions. Region I is the area of classical stability. In region II there is one real positive eigenvalue which is indicative of a pitchfork bifurcation. Pitchfork bifurcations of this model were previously examined by Reidel [ref. 7]. Regions III, IV, and V have at least one pair of complex conjugate eigenvalues with a positive real part. This

would indicate that there should be an unstable oscillatory behavior for the model.

C. SIMULATIONS

An extensive set of simulations were run to verify the bifurcation map's prediction of system stability. While the results of simulations showed the demarcation between the stable and unstable regions, the simulations found that the linear bifurcation analysis failed to predict the method of departure from stability. Five points (a through e) as shown on fig. (3.1) were chosen to illustrate the model's behavior in the regions of interest. Table (3.1) lists the eigenvalues found at each of these points and Table (3.2) shows the eigenvalues associated with the exact bifurcation point near points (b through d). Note that the eigenvalues are given in dimensional terms while all other information in the tables are non-dimensionalized.

Point a is in the region of stability and fig (3.2) shows a rapid convergence to nominal stability. Simulations run at points b, c, and d are shown on figures (3.3), (3.4), and (3.5). These points are spaced less than 2.2 kts apart but they have a huge difference in their bounded oscillatory behavior. Table (3.1) also lists the measured and theoretical periods for these three points. The theoretical period is computed by dividing 2π by the imaginary part of the eigenvalue with the largest real part, and then non-

dimensionalize it by the ratio U/L, where U = speed of the vehicle (in feet per second) at each point, and L = vehicle length (13.9792 ft). An interesting point to note is the behavior at point d, where the measured period is approximately one half of the theoretical period. The dominant period at 80 nd sec is associated with the creation of the limit cycle at the bifurcation point while the 155.4 nd sec period is part of the chaotic response found at point d. Table (3.2) shows the eigenvalues, measured, and theoretical periods at the exact bifurcation points. Finally a simulation run at point is shown in figure (3.6) and demonstrates an unbounded departure from stability.

It is evident that a more detailed analysis had to be performed, and that a Hopf bifurcation analysis should be used.

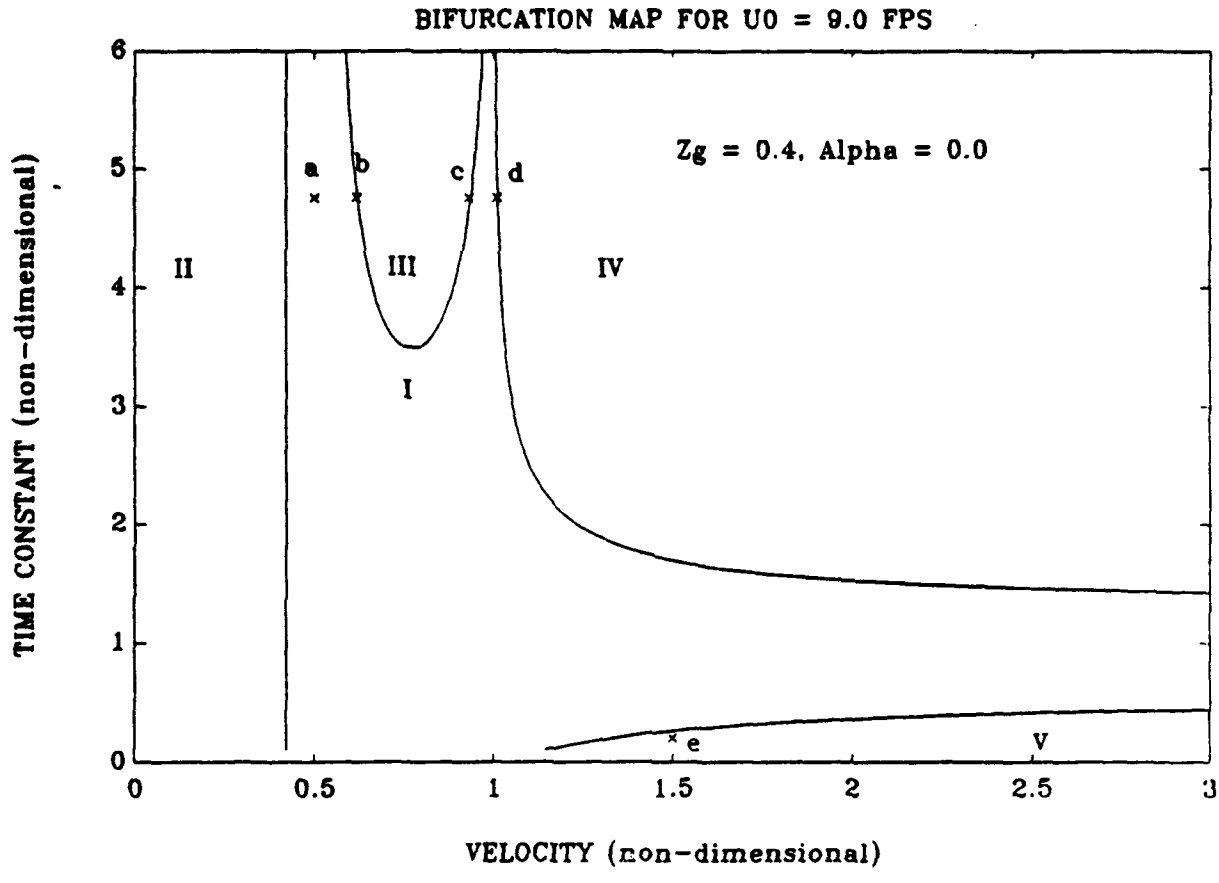

Figure (3.1) - A typical bifurcation map showing the five distinct regions.

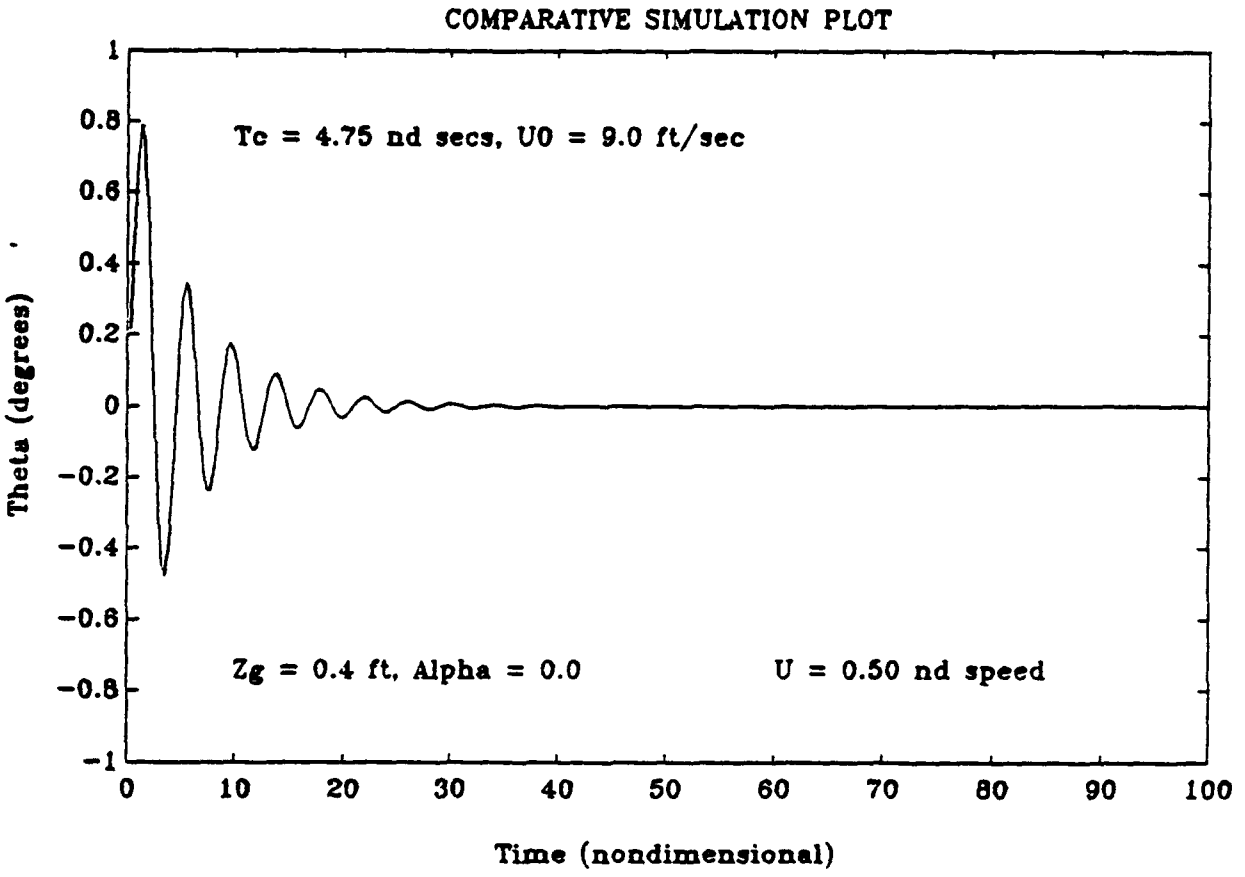

Figure (3.2) - An example of the stable response found in region I. This corresponds to Point a in fig. (3.1).

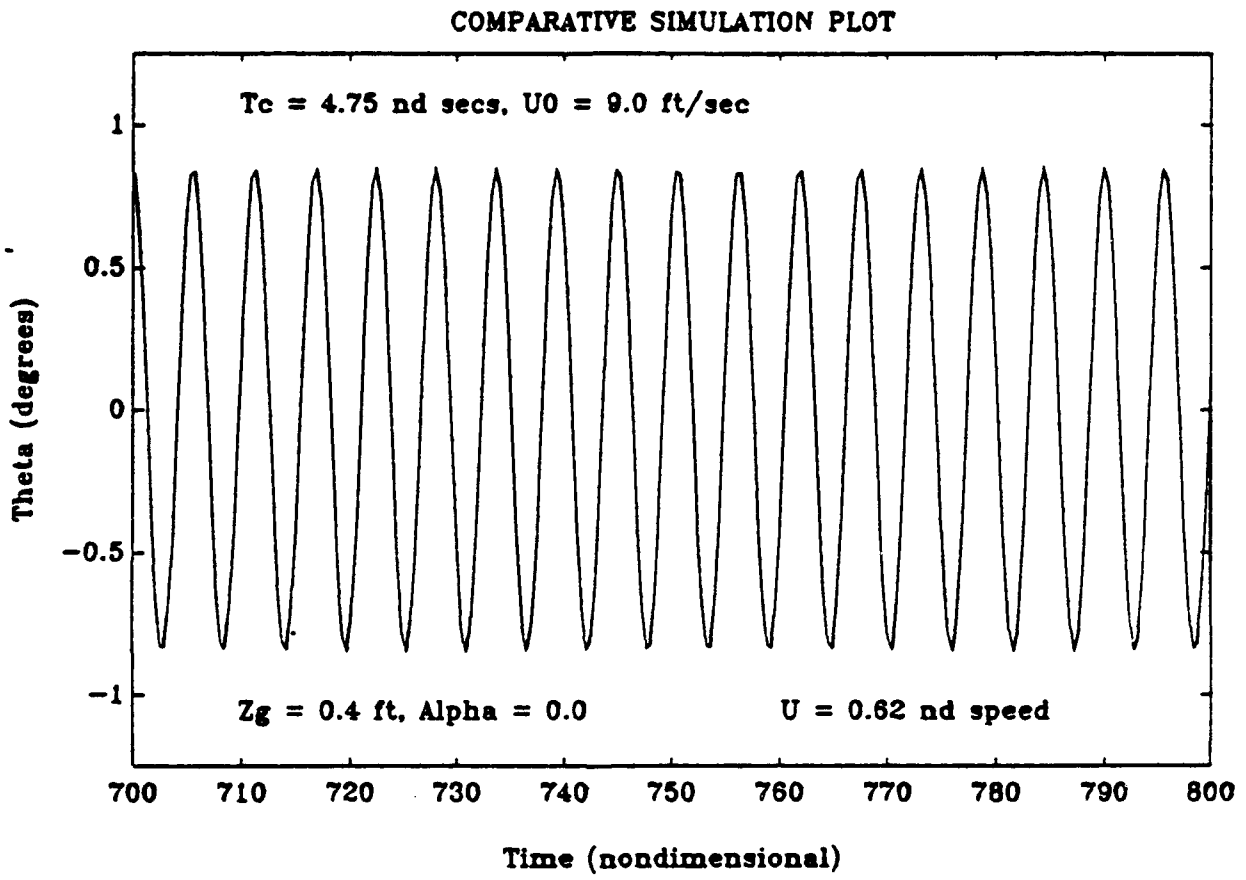

Figure (3.3) - This figure corresponds to Point b in fig (3.1) and shows the oscillatory behavior in region III.

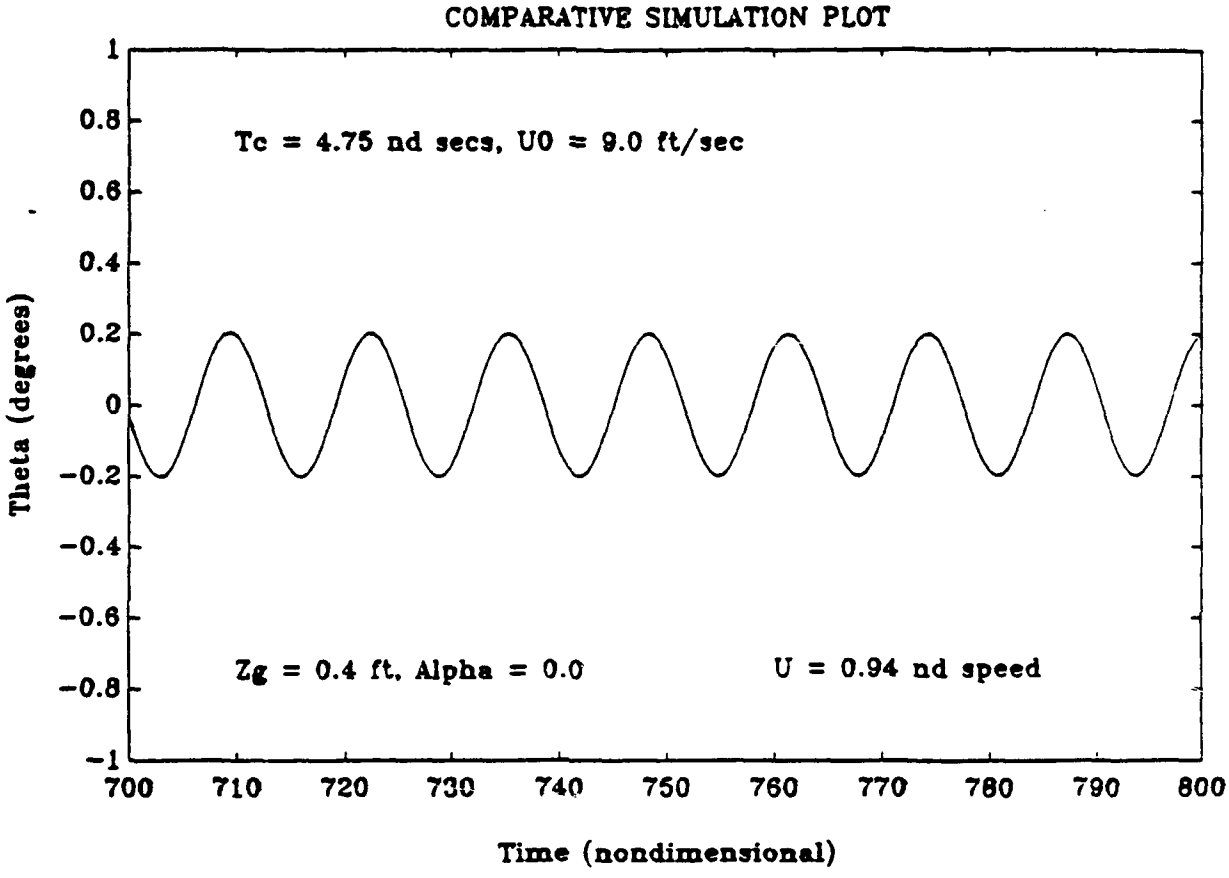

Figure (3.4) - This simulation corresponds to Point c in fig. (3.1). Note the distinct difference in the nature of oscillations between this point and Point b.

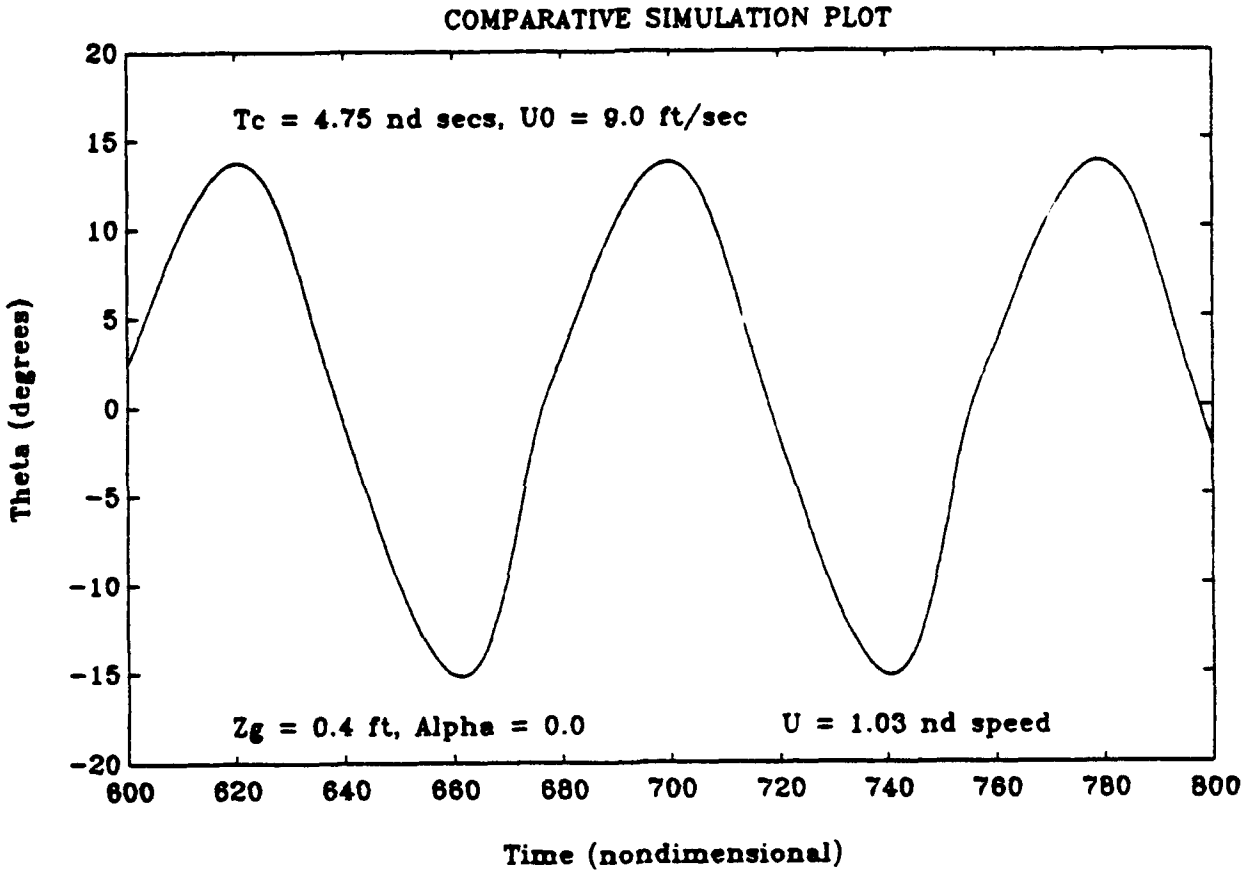

Figure (3.5) - This simulation corresponds to Point d in fig. (3.1). Note the magnitude of the oscillations associated with region IV.

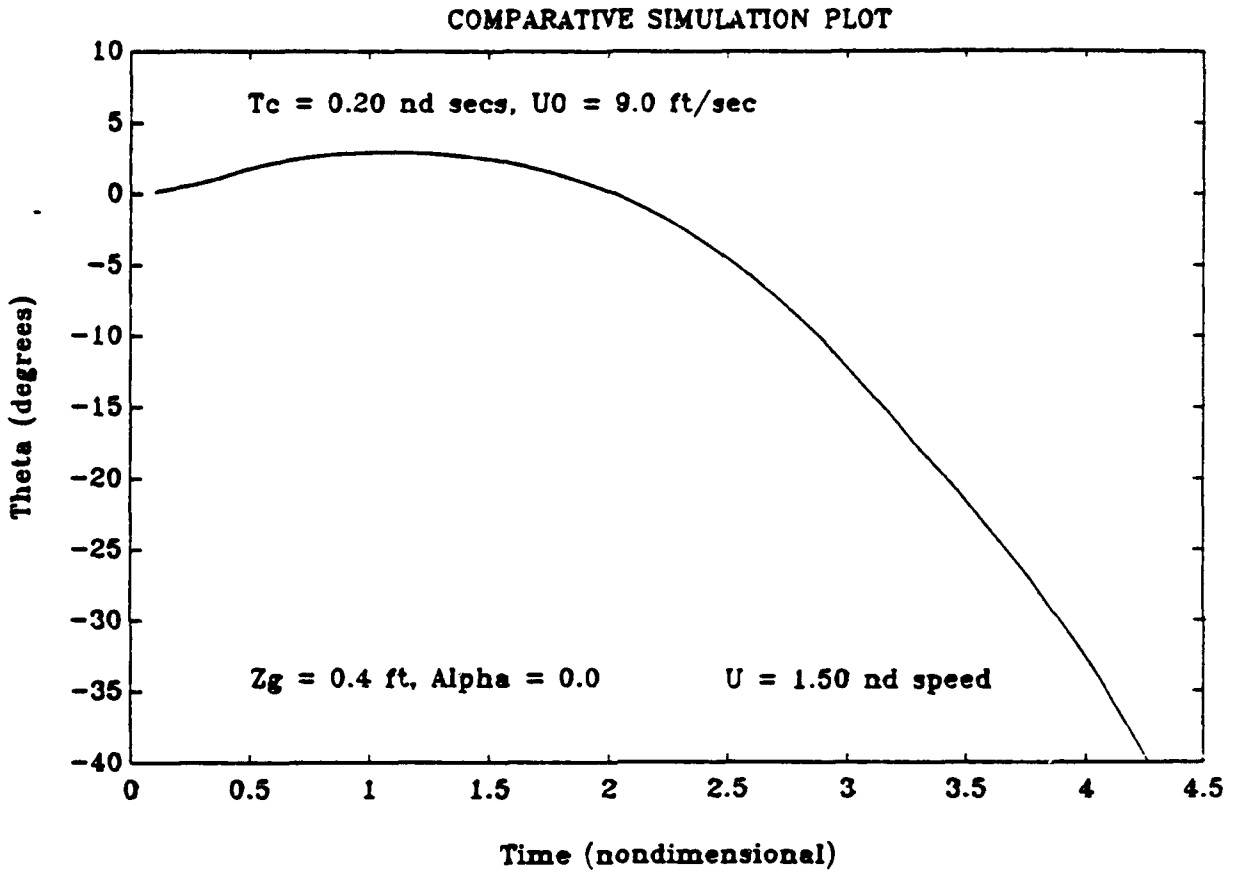

Figure (3.6) - The simulation run at Point e in region V of fig. (3.1) shows the unbounded behavior found in this region.

Points	Eigenvalues	U (fps)	Measured T (nd sec)	Theoretical T (nd sec)
a	-0.4581, -0.0001 -0.0515 ± 0.4938	$0.50U_0$	4.13	5.00
b	-0.6206, -0.0003 0.0052 ± 0.4463	$0.62U_0$	5.59	5.62
c	-0.5888, -0.0046 0.0067 ± 0.2983	$0.94U_0$	12.86	12.75
d	-0.3000 ± 0.2889 0.0387 ± 0.0268	$1.03U_0$	80.00	155.4
e	-27.890, -1.7300 0.7100 ± 4.1700	$1.50U_0$	NA	1.46

Table (3.1) - A listing of the points in fig. (3.1) and their associated eigenvalues, and periods.

Points	Eigenvalues	U (fps)	Measured T (nd sec)	Theoretical T (nd sec)
b	-0.6070, -0.0002 0.0000 ± 0.4506	$0.61U_0$	6.11	5.57
c	-0.5710, -0.0054 0.0000 ± 0.2884	$0.93U_0$	13.13	13.18
d	-0.2675 ± 0.2063 0.0000 ± 0.0558	$1.02U_0$	74.93	74.67

Table (3.2) - Listing of the eigenvalues and periods of the exact bifurcation points associated with the points in Fig. (3.1).

IV. HOPF BIFURCATION

A. INTRODUCTION

By definition a Hopf bifurcation occurs when a pair of complex conjugate eigenvalues cross into the right hand half-plane. When this occurs the system will deviate from a steady state solution in an oscillatory manner. This deviation can be either supercritical or subcritical.

For the supercritical case shown in fig. (4.1), stable limit cycles form after straight line stability is lost. Assume that a parameter, w(t), is varying. When t is less than t_{crit} all of the eigenvalues of the system are located in the left hand half-plane and the system is nominally stable. At t equal to t_{crit}, a complex conjugate pair moves into the right hand half-plane and forms the stable limit cycle. As the distance w(t) increases from $w(t_{crit})$, where the difference $D = w(t) - w(t_{crit})$, the amplitude of the limit cycle will also increase. If D remains small then the system will remain near the nominal steady state solution.

Fig. (4.2) shows the subcritical case with unstable limit cycles being generated prior to the critical point being reached. Thus, as w(t) approaches $w(t_{crit})$ the system could deviate from the nominal steady state solution and converge to a large amplitude limit cycle before the nominal system loses

stability. Hence a random disturbance can cause a nominally stable system, one with all of the eigenvalues in the left hand half-plane, to exhibit oscillatory behavior. Once w(t) equals $w(t_{crit})$ the nominal system becomes unstable and a discontinuous increase in the amplitude of oscillation is seen. This is because there are no nearby stable attractors for the system to converge to.

A system design must make a distinction between these two types of bifurcations because of the disparate nature of the losses in stability. Thus the designer cannot rely on a linear approximation and must use higher order approximations (3^{rd} order) of the equations of motion to adequately analyze his/her system.

B. THIRD ORDER APPROXIMATIONS

The nonlinear equations of motion are expanded using a third order Taylor series approximation near the nominal steady state, $\mathbf{x} = [0]$. The control law is then modelled as,

$$\delta' = \delta_{sat}\tanh\left(\frac{\delta}{\delta_{sat}}\right) , \qquad (4.1)$$

where δ_{sat} is the saturation angle of the control plane input. Using the same approximation for the control law as the equation of motion, δ' becomes

$$\delta' = \delta - \frac{1}{3\delta_{sat}^2}\delta^3 \ . \tag{4.2}$$

Therefore, the state equations can now be written as

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + g(\boldsymbol{x}) \ , \tag{4.3}$$

where

$$\boldsymbol{x} = [\theta \ w \ q \ z]^T \ , \tag{4.4}$$

and the higher order terms are,

$$g_1 = 0 \ , \tag{4.5a}$$

$$g_2 = b_1 U^2 \delta_3(\theta, w, q, z) - \frac{1}{6} a_{13} Z_{GB} \theta^3 + c_{11} q^2 + c_{12} wq \ , \tag{4.5b}$$

$$g_3 = b_2 U^2 \delta_3(\theta, w, q, z) - \frac{1}{6} a_{23} Z_{GB} \theta^3 + c_{21} wq + c_{22} q^2 \ , \tag{4.5c}$$

$$g_4 = -\frac{1}{2} w\theta^2 + \frac{1}{6} U\theta^3 \ . \tag{4.5d}$$

The term δ_3 contains the third order terms derives from substituting δ into δ'.

Defining T as the matrix of eigenvectors of A evaluated at the Hopf bifurcation point, then the transformation,

$$\boldsymbol{x} = T\boldsymbol{z} \ , \tag{4.6}$$

transforms the system into a canonical form

$$\dot{\boldsymbol{z}} = T^{-1}AT\boldsymbol{z} + T^{-1}g(T\boldsymbol{z}) \ . \tag{4.7}$$

At the bifurcation point

$$T^{-1}AT = \begin{vmatrix} 0 & -\omega_0 & 0 & 0 \\ \omega_0 & 0 & 0 & 0 \\ 0 & 0 & p & 0 \\ 0 & 0 & 0 & q \end{vmatrix} , \tag{4.8}$$

with $\omega_0 > 0$ and p,q < 0. z_3 and z_4 correspond to p and q and are asymptotically stable. Center manifold theory states that the stable coordinates z_3, z_4 can be expressed as polynomials in the critical coordinates z_1, z_2 and this relationship is at least second order. Therefore, we can write,

$$z_3 = \alpha_1 z_1^2 + \alpha_2 z_1 z_2 + \alpha_3 z_2^2 \ , \tag{4.9a}$$

and

$$z_4 = \beta_1 z_1^2 + \beta_2 z_1 z_2 + \beta_3 z_2^2 \ . \tag{4.9b}$$

The coefficients α_i and β_i can be computed as follows: We differentiate equations (4.9) with respect to $\mathbf{z}$,

$$\dot{z}_3 = 2\alpha_1 z_1 \dot{z}_1 + \alpha_2(\dot{z}_1 z_2 + z_1 \dot{z}_2) + 2\alpha_3 z_2 \dot{z}_2 ,$$
$$\dot{z}_4 = 2\beta_1 z_1 \dot{z}_1 + \beta_2(\dot{z}_1 z_2 + z_1 \dot{z}_2) + 2\beta_3 z_2 \dot{z}_2 ,$$

and substitute $\dot{z}_1 = -\omega_0 z_2$, and $\dot{z}_2 = \omega_0 z_1$. Therefore,

$$\dot{z}_3 = \alpha_2 \omega_0 z_1^2 + (2\alpha_3 - 2\alpha_1)\omega_0 z_1 z_2 - \alpha_2 \omega_0 z_2^2 , \qquad (4.10a)$$

and

$$\dot{z}_4 = \beta_2 \omega_0 z_1^2 + (2\beta_3 - 2\beta_1)\omega_0 z_1 z_2 - \beta_2 \omega_0 z_2^2 . \qquad (4.10b)$$

The third and fourth equations of (4.7) are written as,

$$\begin{vmatrix} \dot{z}_3 \\ \dot{z}_4 \end{vmatrix} = \begin{vmatrix} p & 0 \\ 0 & q \end{vmatrix} \begin{vmatrix} z_3 \\ z_4 \end{vmatrix} + [D] , \qquad (4.11)$$

where,

$$[D] = T^{-1} g_2(T\mathbf{z}) ,$$

and $g_2(T\mathbf{z})$ contains the second order terms of $g(T\mathbf{z})$. We substitute equations (4.9) into (4.11) and equate coefficients with (4.10). In this way we get a linear system in α_i, and β_i. From this we can write the two dimensional state space equations as

$$\dot{z}_1 = -\omega_0 z_2 + r_{11} z_1^3 + r_{12} z_1^2 z_2 + r_{13} z_1 z_2^2 + r_{14} z_2^3 + p_{11} z_1^2 + p_{12} z_1 z_2 + p_{13} z_2^2 \,, \tag{4.12a}$$

and

$$\dot{z}_2 = \omega_0 z_1 + r_{21} z_1^3 + r_{22} z_1^2 z_2 + r_{23} z_1 z_2^2 + r_{24} z_2^3 + p_{21} z_1^2 + p_{22} z_1 z_2 + p_{23} z_2^2 \,, \tag{4.12b}$$

where the coefficients r_{ij} and p_{ij} are derived from equations (4.8).

These equations are only valid exactly at the Hopf bifurcation point. For speeds U in a region near the bifurcation point these equations become

$$\dot{z}_1 = \alpha' \epsilon z_1 - (\omega_0 + \omega' \epsilon) z_2 + F_1(z_1, z_2) \,, \tag{4.13a}$$

and

$$\dot{z}_2 = (\omega_0 + \omega' \epsilon) z_1 + \alpha' \epsilon z_2 + F_2(z_1, z_2) \,, \tag{4.13b}$$

where: α', ω' are the derivatives with respect to U of the real and imaginary parts of the critical complex conjugate pair of eigenvalues evaluated at the bifurcation point: ϵ is the difference in U from the critical value; and, the nonlinear functions F_1 and F_2 are

$$F_1 = r_{11}z_1^3 + r_{12}z_1^2z_2 + r_{13}z_1z_2^2 + r_{14}z_2^3 + p_{11}z_1^2 + p_{12}z_1z_2 + p_{13}z_2^2 \ , \tag{4.14a}$$

and

$$F_2 = r_{21}z_1^3 + r_{22}z_1^2z_2 + r_{23}z_1z_2^2 + r_{24}z_2^3 + p_{21}z_1^2 + p_{22}z_1z_2 + p_{23}z_2^2 \ . \tag{4.14b}$$

By transforming z_1 and z_2 to polar coordinates in the form

$$z_1 = R\cos\Theta \tag{4.15a}$$

$$z_2 = R\sin\Theta \tag{4.15b}$$

equations (4.13) become

$$\dot{R} = \alpha'\epsilon R + F_1(R,\Theta)\cos\Theta + F_2(R,\Theta)\sin\Theta \tag{4.16a}$$

and

$$R\dot{\Theta} = (\omega + \omega'\epsilon)R + F_2(R,\Theta)\cos\Theta - F_1(R,\Theta)\sin\Theta \ . \tag{4.16b}$$

Equation (4.16a) then yields

$$\dot{R} = \alpha'\epsilon R + P(\Theta)R^3 + Q(\Theta)R^2 \ . \tag{4.17}$$

By averaging equation (4.17) over one cycle we can obtain an equation with constant coefficients. Defining

$$K = \frac{1}{2\pi}\int_0^{2\pi} P(\theta)\, d\theta \ , \tag{4.18}$$

and

$$L = \frac{1}{2\pi}\int_0^{2\pi} Q(\theta)\, d\theta \ , \tag{4.19}$$

and carrying out the integration we obtain

$$L = 0 \ , \tag{4.20}$$

and

$$K = \frac{1}{8}(3r_{11} + r_{13} + r_{22} + 3r_{24}) \ , \tag{4.21}$$

which reduces equation (4.17) to

$$\dot{R} = \alpha'\epsilon R + KR^3 \ . \tag{4.22}$$

The existence and stability of the limit cycle is determined by analyzing the equilibrium points of the averaged equation (4.22), which correspond to periodic solutions in z_1 z_2 as seen in the coordinate transformation equations (4.15). From equation (4.22) we can see that two conditions exist,

1) If $\alpha' > 0$, then:

   (a) If K > 0 then unstable limit cycles coexist with the stable equilibrium for $\epsilon < 0$.

(b) If $K < 0$ then stable limit cycles coexist with the unstable equilibrium for $\epsilon > 0$.

2) If $\alpha' < 0$, then:

(a) If $K > 0$ the unstable limit cycles coexist with the stable equilibrium for $\epsilon > 0$.

(b) If $K < 0$ then stable limit cycles coexist with the unstable equilibrium for $\epsilon < 0$.

From this criteria, by computing the nonlinear coefficient K we can use it to distinguish the two different types of Hopf bifurcations:

- Supercritical if $K < 0$;
- Subcritical if $K > 0$.

[ref. 6]

C. RESULTS

A typical bifurcation map of stability for the SUBOFF Model is shown in fig. (4.3). This map is characterized by the Pitchfork bifurcation curve (P) and the three Hopf bifurcation curves (H1, H2, and H3). The nature of curve P was previously analyzed and those results (Riedel, 1993) are reconfirmed in this study.

Curve H1 is characterized by a weak supercritical branch (a → b) at low nominal speeds, U_0. As U_0 increases this branch develops a weak to moderate subcritical behavior with K between 0 and 10^2. The second branch of H1 (b → c) has a

consistent moderate subcritical behavior with K on the order of 10^2.

Cusp (C) marks the intersection of curve H2 with curve H3. The cusp is highly dependent on both U_0 and initial stability Z_{GB}.

As Z_{GB}, for a given U_0, increases curve H2 (d → e) shifts from a very weak subcritical nature with K between $+10^{-2}$ and 1 to a very weak supercritical nature with K between -1 and -10^{-2}. With a lower U_0 and/or higher Z_{GB}, point e moves down in the time constant and may not intersect curve H3.

Curve H3 (f → g) is a strong supercritical bifurcation with K values between -10^4 and -10^6. The position of H3 is independent of U_0, initial stability, and control surface coordination gain.

Finally, curve H4 is a strong subcritical branch with K having values between 10^3 and 10^6. Because of this highly subcritical behavior, H4 can dominate and obscure the stable region at speeds greater than $U/U_0 = 1$.

Figure (4.4) plots the K values for a representative bifurcation map shown in fig (3.1). Note the predicted super- and subcritical branches associated with fig (4.3). Point (a) is inside the stable region I and the numerical simulations seen in fig. (3.2) converge to zero. Point (b) is located in the unstable region, immediately after a supercritical bifurcation. As a result, small amplitude limit cycle oscillations have developed. The same is true as we move

towards point (c) although we expect a family of unstable limit cycles around this point as a result of the subcritical bifurcation. This point is further explored in the next section. As we approach point (d) a family of stable limit cycles is generated but its behavior is influenced by the previously developed unstable limit cycles. The real part of the critical pair of eigenvalues is becoming positive and relatively large, see fig. (3.1), which means that the amplitudes of these stable limit cycles are expected to be significantly higher, a result which is confirmed by fig. (3.5). The imaginary parts of the critical pair of eigenvalues are also changing very fast in this region. This means that the period of these limit cycles must be computed based on the value of the imaginary part right at the bifurcation point, rather than its value at the specific parameter point. This was observed previously in tables (3.1) and (3.2). Point (e) is in the strongly subcritical region V, thus we see the rapid divergence from stability as seen in fig. (3.6).

D. SIMULATIONS

The response of the submersible was simulated using an Adams-Bashforth integration scheme in Fortran program coding and is included in Appendix C. The control law (2.15) and gains (2.17) discussed in Chapters II and III were used. Non-dimensional ships speed and control system time constant,

nominal speed, initial stability, and control surface coordination gain were the input values to the system. A nominal 0.1 ft/sec vertical speed was used as the external initial disturbance.

The simulations were used to compare the Hopf bifurcation data in two ways:

1) By confirming sub-/ supercritical behavior predicted by the K factor, and;

2) By comparing the predicted to simulated period of oscillations.

In fig. (4.5) we have plotted the stable equilibria and the stable and unstable limit cycles from our example first used in fig. (3.1). Figure (4.5) clearly shows the predicted sub- and supercritical behavior that the K values predicted. The important features to note in fig. (4.5) are:

1. Unstable limit cycles found with the subcritical Hopf bifurcation;

2. The divergence of the amplitudes as the velocity moves away from the bifurcation point velocity, and;

3. The rapid divergence of the right most bifurcation and its quick and abrupt termination.

To see why the abrupt termination occurs we must look at the root locus plot as shown in fig. (4.6). The parameter in this the third bifurcation terminates when there is a break in point and the complex conjugate pair, which had given the oscillatory response, becomes two positive real eigenvalues.

These two eigenvalues now make the system completely unstable. The splitting of this complex conjugate pair into two real and positive eigenvalues can be better seen in fig (4.7).

SUPERCRITICAL HOPF BIFURCATION

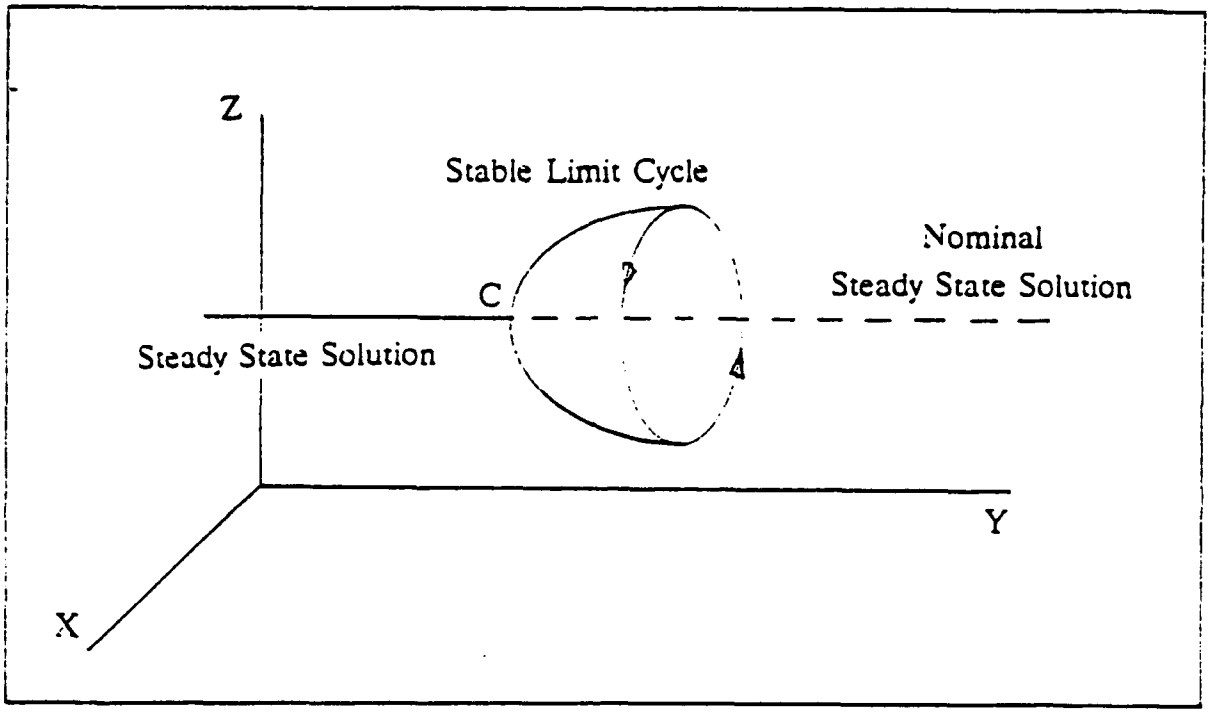

Figure (4.1) - An example of a supercritical Hopf bifurcation where the system has a nominal steady state solution until point C, which occurs at t_{crit}. After t_{crit} the system becomes unstable but forms a stable limit cycle.

SUBCRITICAL HOPF BIFURCATION

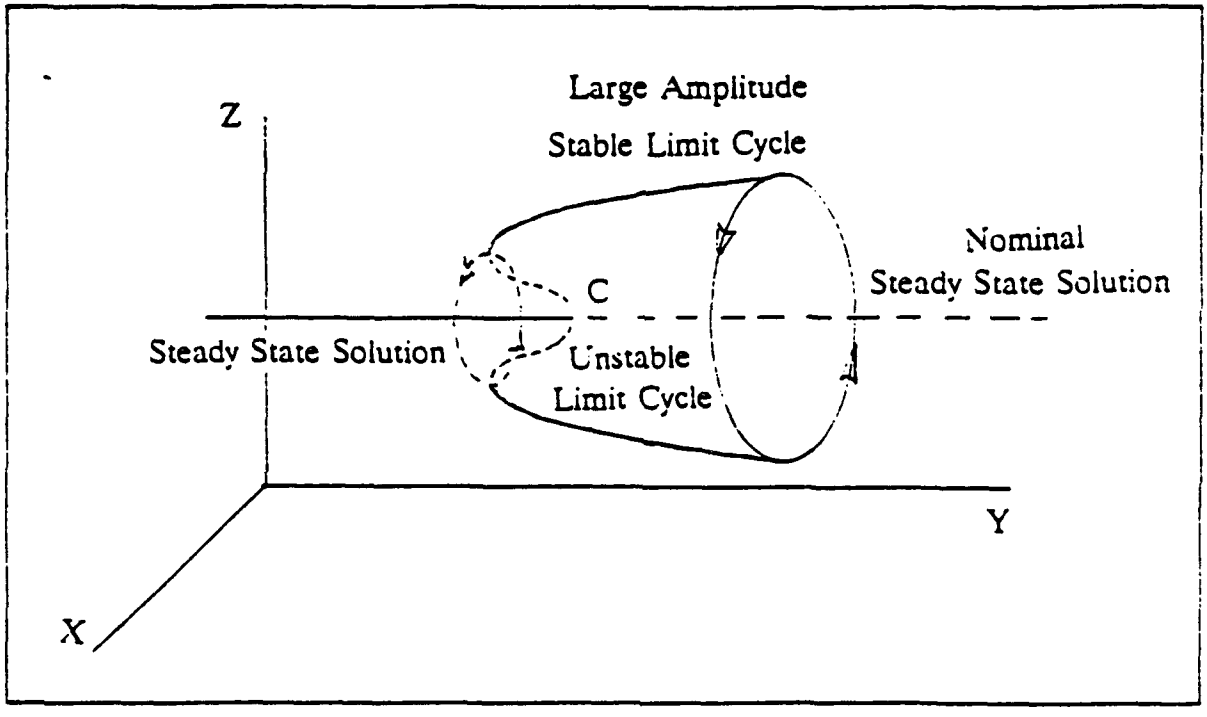

Figure (4.2) - This figure shows a subcritical Hopf bifurcation where the system loses stability prior to reaching point C, which again occurs at t_{crit}.

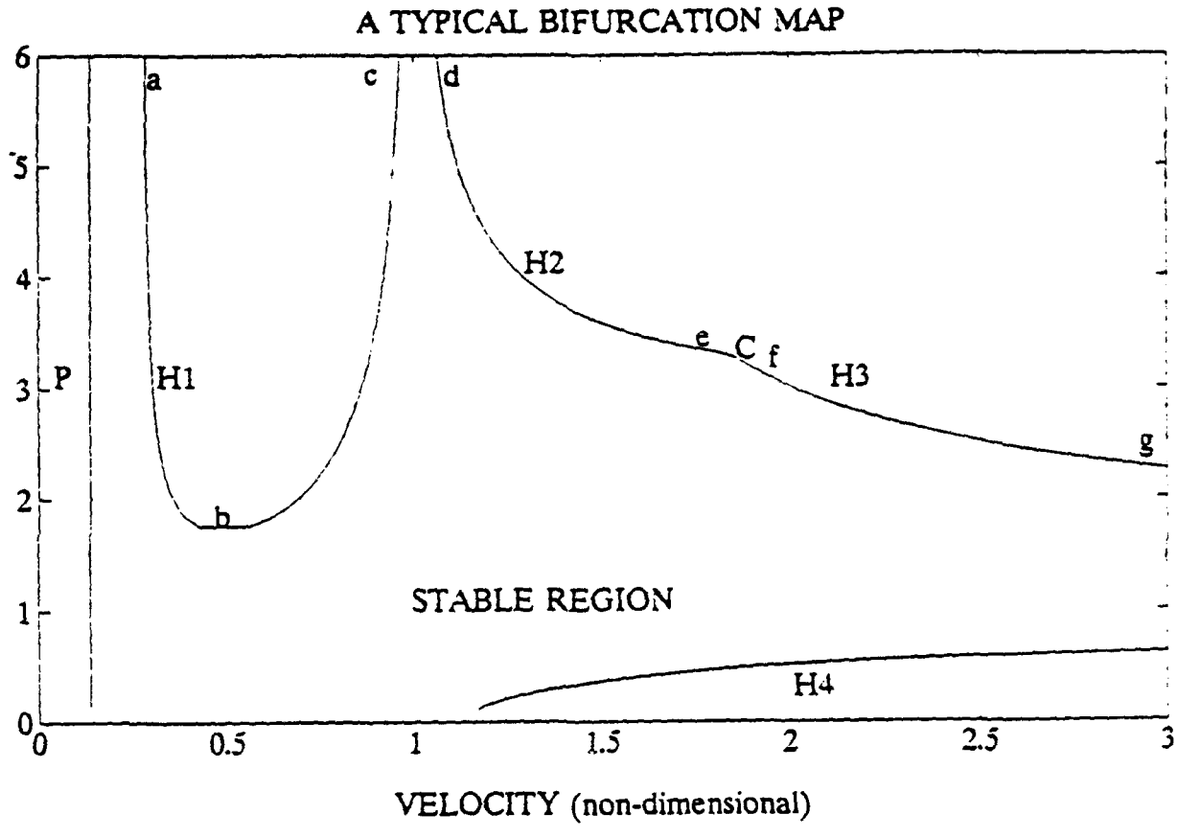

Figure (4.3) - An annotated bifurcation map for the SUBOFF Model.

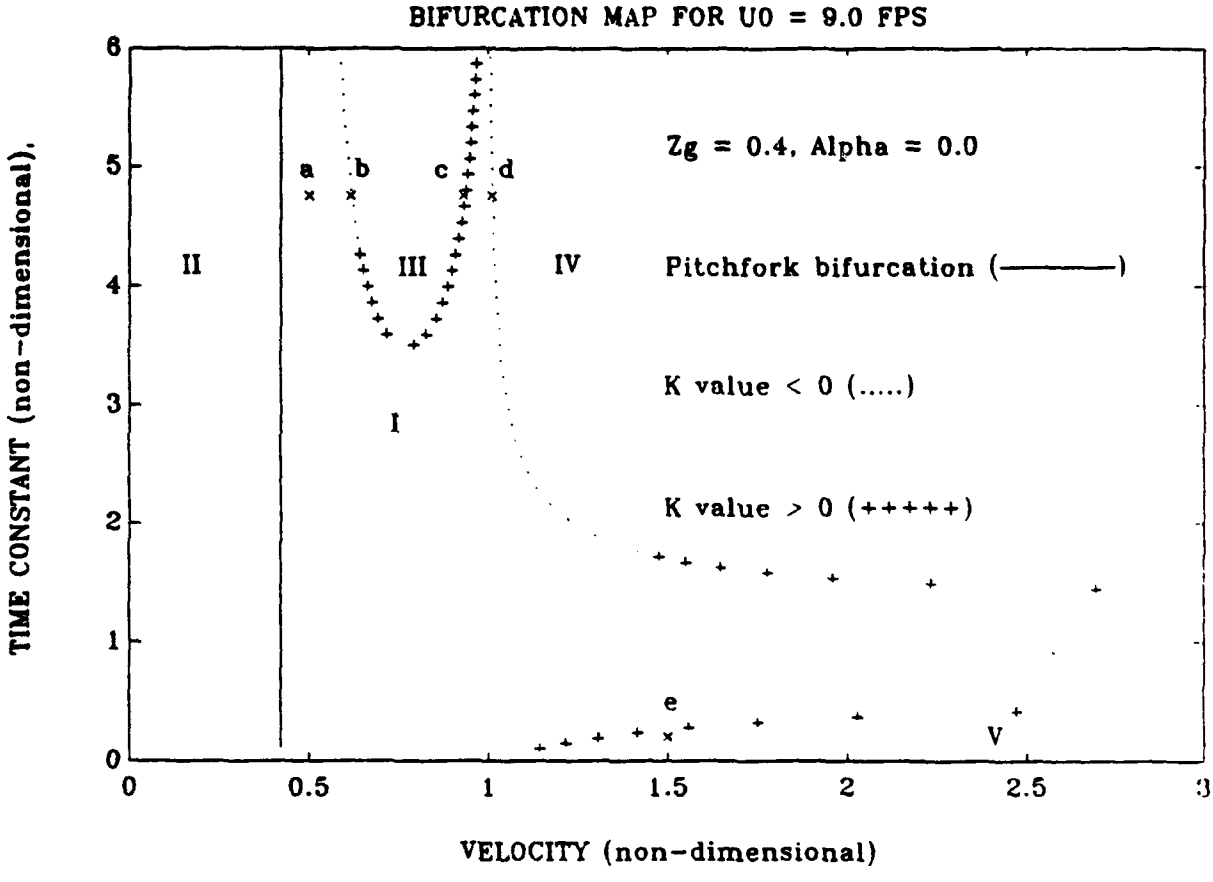

Figure (4.4) - A plot of the K values associated with fig. (3.1).

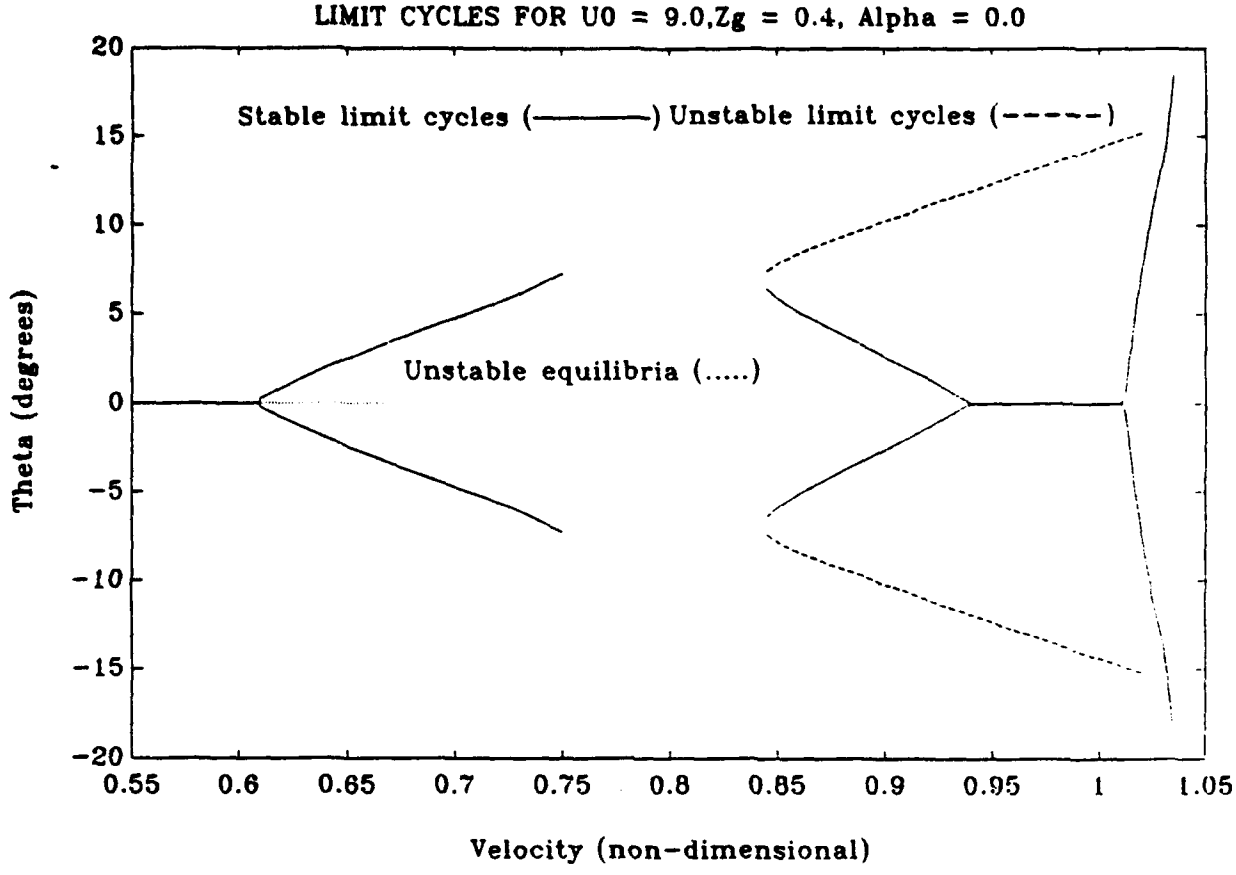

Figure (4.5) - This shows the amplitude response for a speed range encompassing the Hopf bifurcation points that are shown in fig. (3.1).

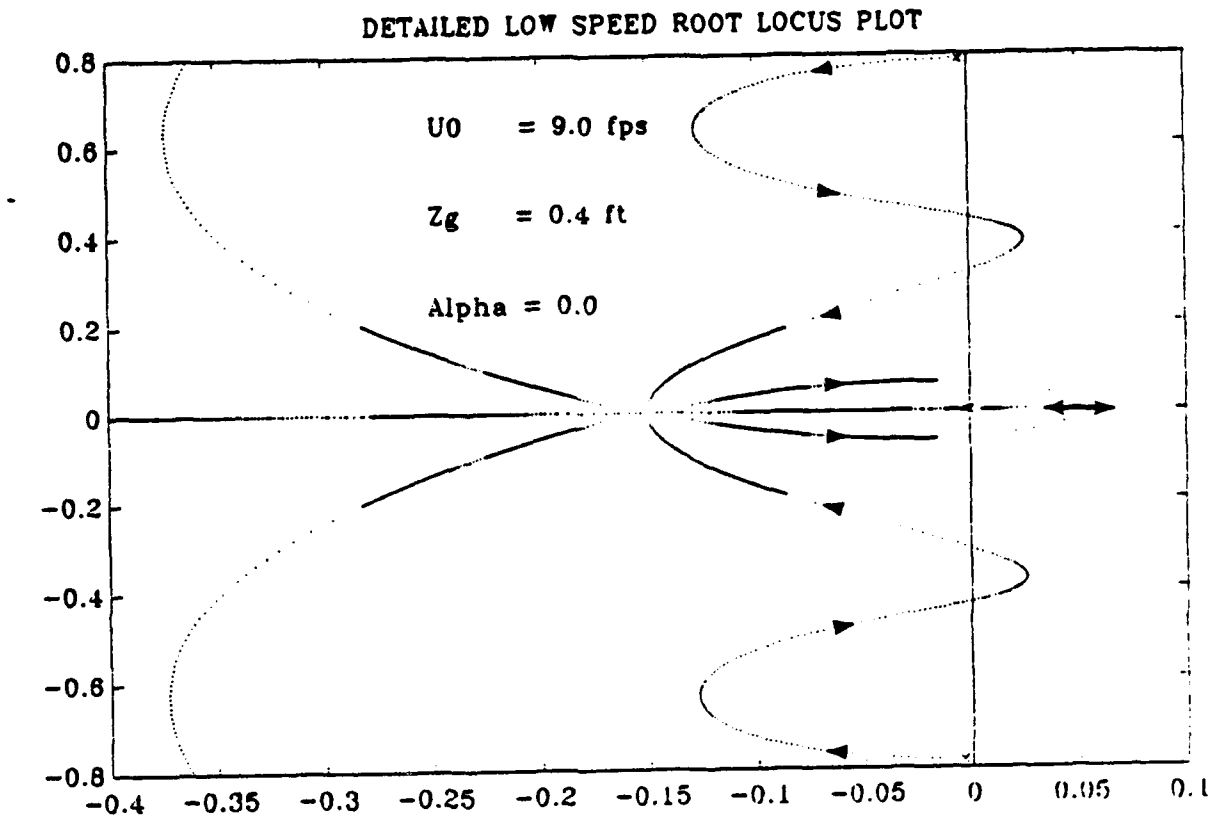

Figure (4.6) - The root locus plot of the eigenvalues associated with fig. (4.5).

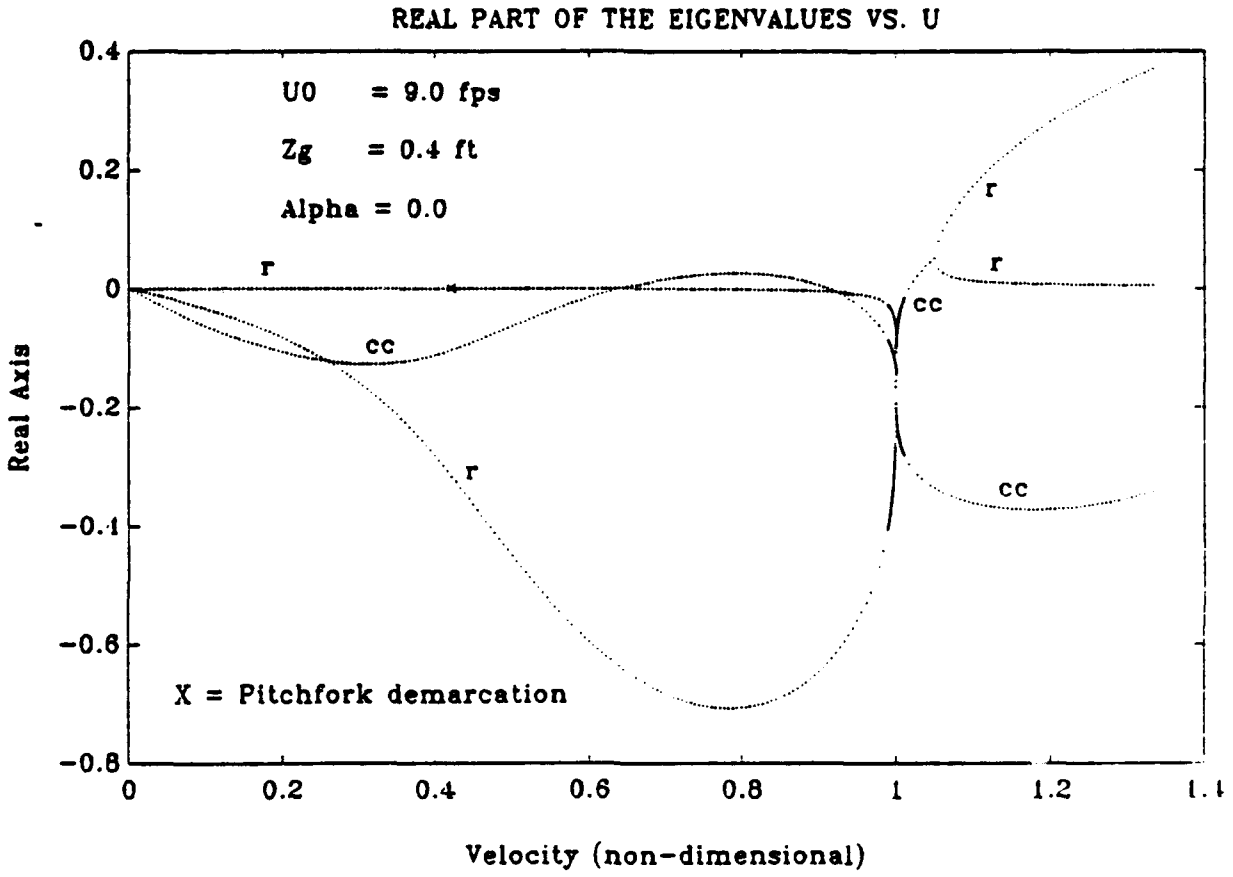

Figure (4.7) - A plot of the real parts of the eigenvalues plotted in fig. (4.6).

V. APPLICATIONS

A. CONTROL SYSTEM PARAMETERS

From the typical bifurcation maps we can see that a region of stability is created between the pitchfork and Hopf bifurcation boundaries. For the control system designer the limits or parameters must be defined prior to starting the design. By maximizing the region of stability we can give the designer the most leeway in his/her work. There are three parameters that we can use to change the bifurcation maps, nominal speed, initial stability, and the control surface gain coefficient.

First we will look at changing the nominal speed. In fig. (5.1) we have plotted three curves for nominal speeds of 3.0, 9.0, and 15.0 fps. We can see that although the pitchfork line moves to the left in dimensional speeds this line remains nearly constant with a dimensional stern planes reversal occurring at 1.2 kts. The high speed Hopf boundaries ($U/U_0 > 1.0$) move apart as the nominal speed increases. The effectiveness of increasing U_0 is limited in the upper branch by the fixed position of the H3 line with the maximum practical T_c achieved at a U_0 = 9.0 fps. In the lower arm there is no increase in the stability area after U_0 = 9.0 fps therefore any increase in U_0 is pointless. For the low speed

Hopf curves ($U/U_0 < 1.0$) we quickly lose our margin of stability as U_0 increases thus necessitating further changes to regain the lost area of stability.

The next parameter change we examined was varying the initial stability. Figure (5.2) shows the effect of increasing Z_g from 0.1 to 0.4 ft. The subcritical H4 branch remains constant while the upper high speed Hopf branch moves down effectively decreasing the area of stability. The low speed Hopf curves move up to increase the low speed area of stability. We can see that the additional loss in area is by the movement of the pitchfork line to the right. At a Z_g = 0.4 ft stern planes reversal occurs at a dimensional speed of 2.4 kts which is well within the currently accepted range of 1.0 to 3.0 kts for modern submarines. Therefore in this model we would want to balance the initial stability to maximize the low and high speed areas.

An increase in the control surface gain coefficient, α, is shown in fig. (5.3). Note that the low and high speed Hopf curves all move up in time constant. While the low speed Hopf curves give a large increase in stability, the high speed curves move up proportionally and there is no increase in stability area. This does allow the designer to shift the range of stable time constants without a loss of high speed stability. The pitchfork line will move to the left until it equals zero when α = 1.0.

In order to examine what happens at the extremes of the design options we can look at the low nominal speed (3 fps) bifurcation maps. Figure (5.4) shows the typical bifurcation map such as the one we have previously discussed. As the metacentric height is increased there are significant changes in the nature of the bifurcation curves. In fig. (5.5) we see that the pitchfork bifurcation line has moved significantly to the right and has intersected the low speed Hopf bifurcation curve. This intersection along with the merger of the H2 and H4 curves have combined to reduce the region of stability to a negligible portion of the map.

A further increase in the metacentric height as shown in fig. (5.6) demonstrates a dramatic change in the nature of the stability of the model. The low speed region has two hyperbolic-like Hopf bifurcation curves (the upper curve occurs well above the region of interest) bounding the lower and upper limits of stability. For the speeds $U/U_0 > 1.0$, the pitchfork bifurcation line now intersects the H2 curve and has changed from a supercritical to a subcritical pitchfork.

Figure (5.7) is an example of the effect just described and how it can occur at higher nominal speeds. This shows that although initial stability is necessary for overall stability, if the metacentric height becomes too large it can have an adverse affect on the performance of the submarine.

B. SUBMARINE DESIGN EVALUATION

For all of the simulations up to this point the moment and inertia coefficients of the control surfaces have been .5x of those listed in the SUBOFF Model report. By using the reduce effect control surface input we have been able to show stable simulations in all of the mapped stable regions of the bifurcation maps.

Linear bifurcation methods fail to predict the change in system response for changes in control surface coefficients. The bifurcation maps for .1x to 1.0x the control surface coefficients are exactly the same, in other words the bifurcation points are independent of the size and effectiveness of the control surfaces. Therefore we must examine the K values in order to predict the response of the model.

Figure (5.8) shows the change in stability for the model with and increase in the control surface coefficients. The area of lost stability is indicated by the shaded portion of the map. This loss of stability is caused by the shift of the H2 curve from weak to moderately supercritical to a strongly subcritical curve. With two strongly subcritical curves in the high speed region ($U/U_0 > 1.0$) the possibility of subcritical capture is greatly increased. This effect is confirmed by running an extensive set of simulations and mapping the change in stable to unstable response. We must note that this instability occurs in a region that has four

eigenvalues with negative real parts where linear control system design would not predict an instability.

With this method a design can be tested and then modified using the design goals as determined by the nonlinear response in the simulations. This can change a completely unstable model such as the SUBOFF Model into one where a large margin of stability and control system latitude is design into it.

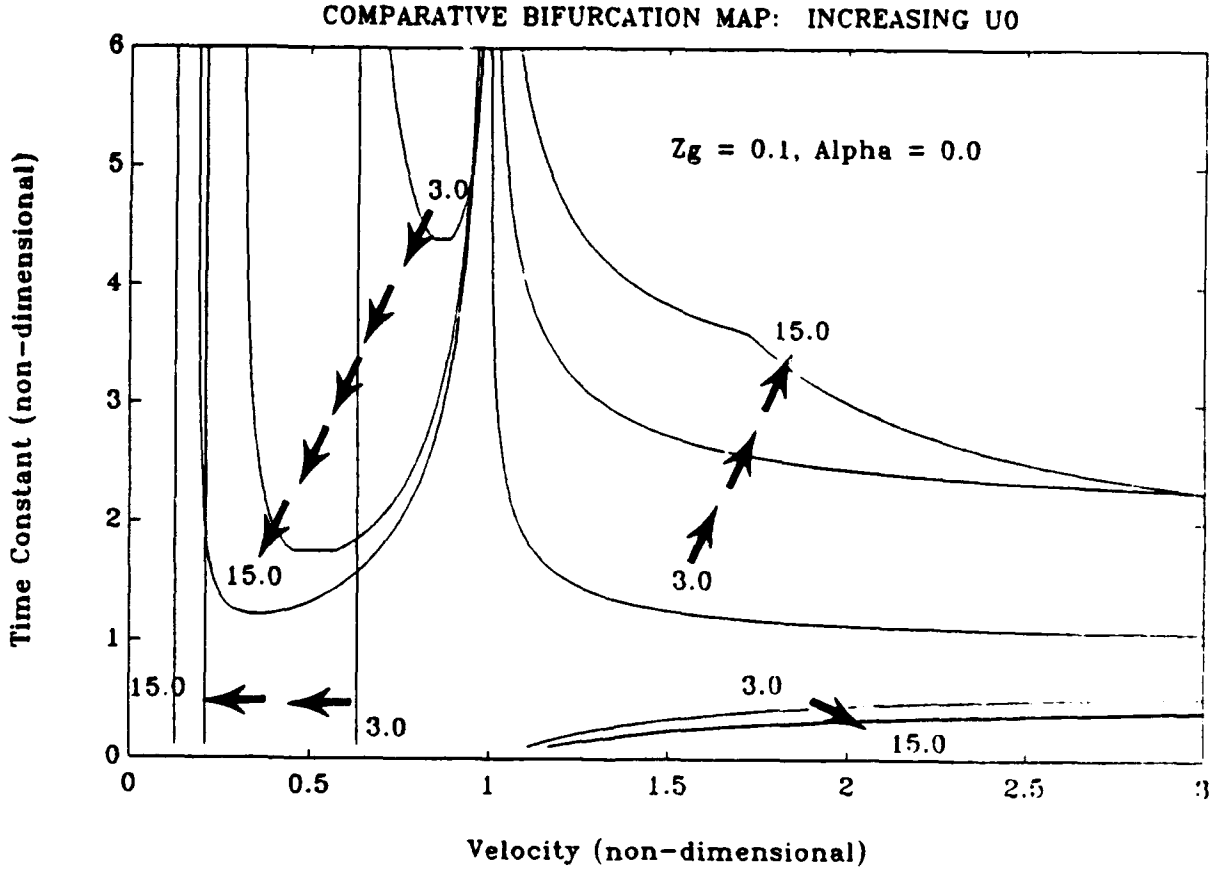

Figure (5.1) - The effects of changing the nominal speed, U_0, on the bifurcation maps.

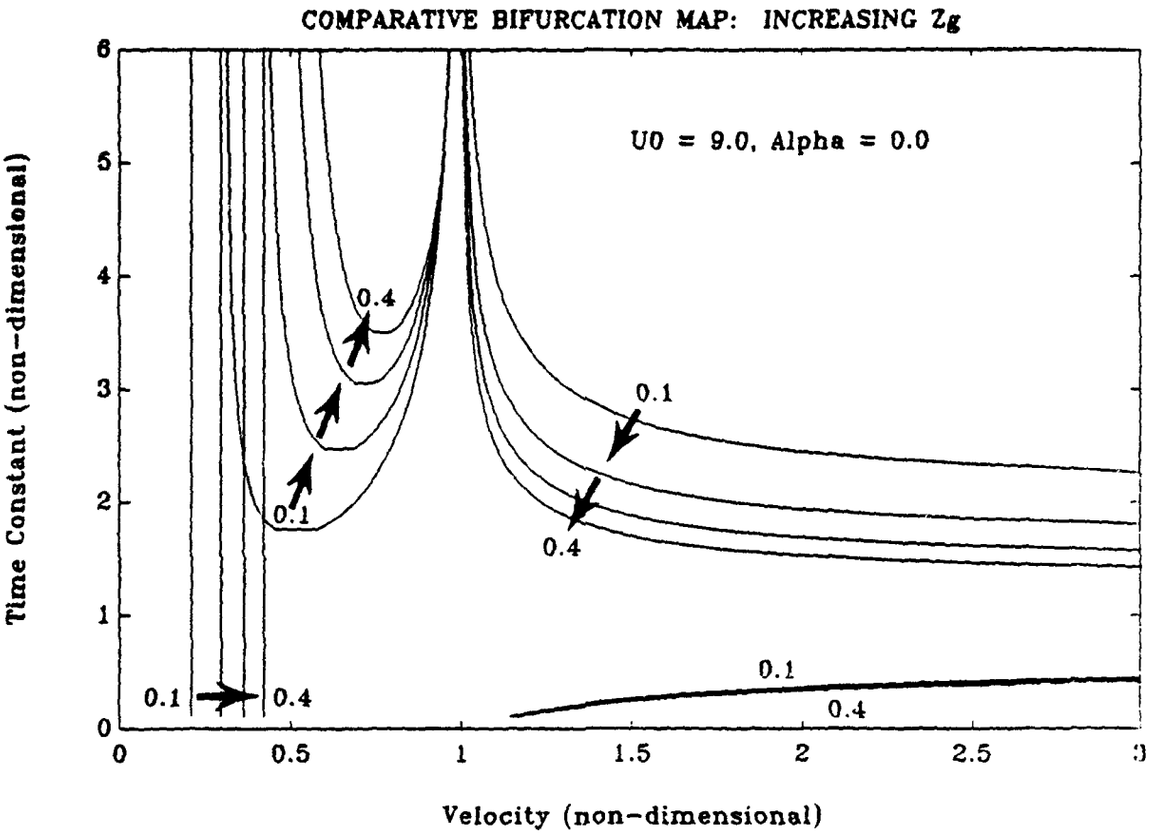

Figure (5.2) - The effects of changing initial stability, Z_g, on the bifurcation maps.

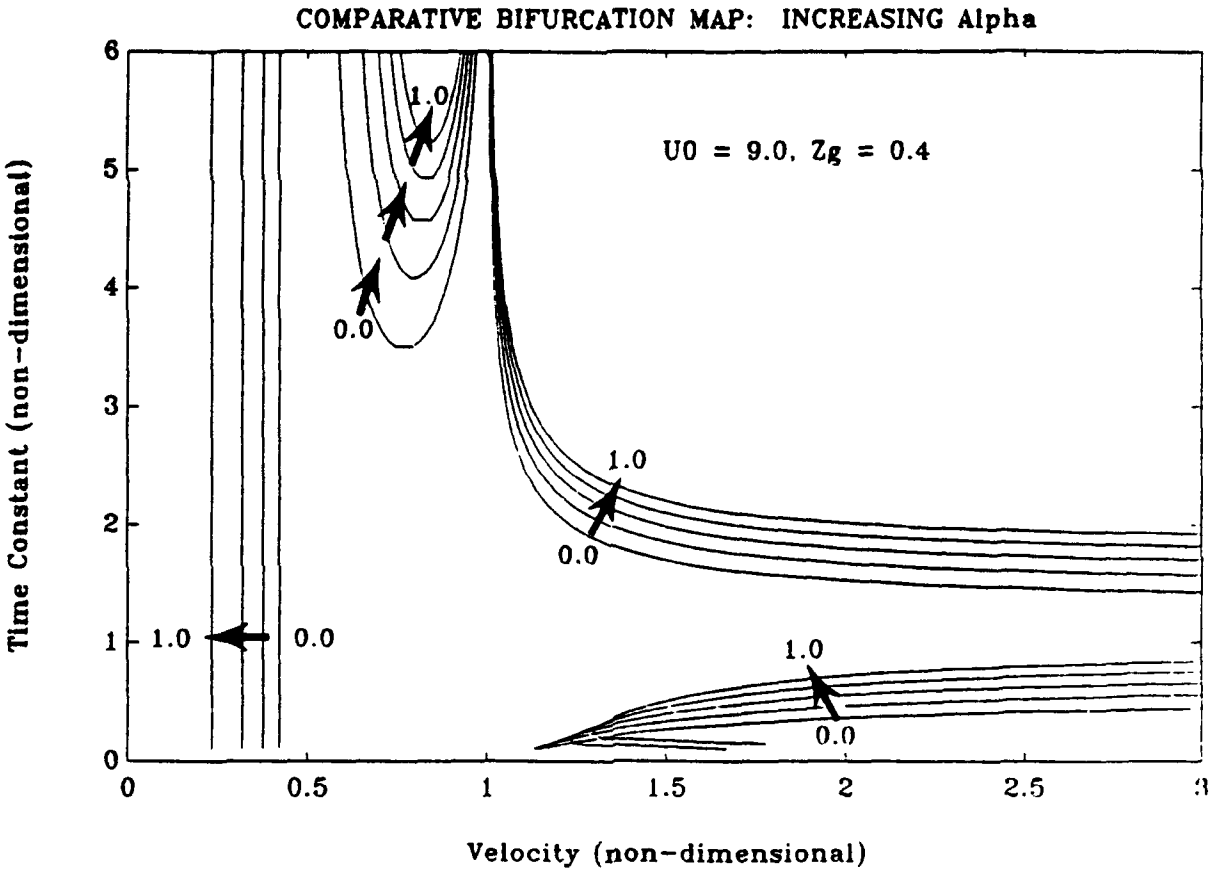

Figure (5.3) - The effects of changing the control surface gain coefficient, α, on the bifurcation maps.

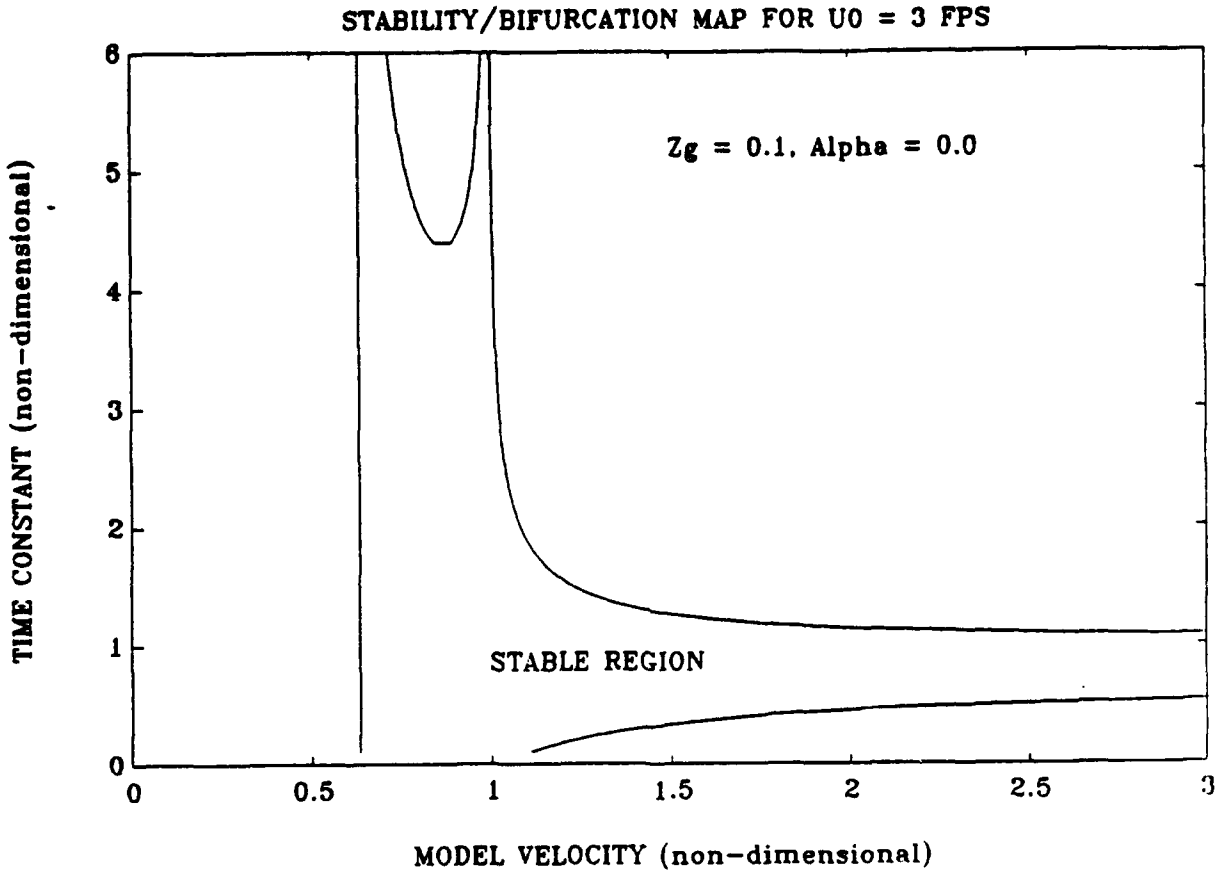

Figure (5.4) - A low nominal speed bifurcation map.

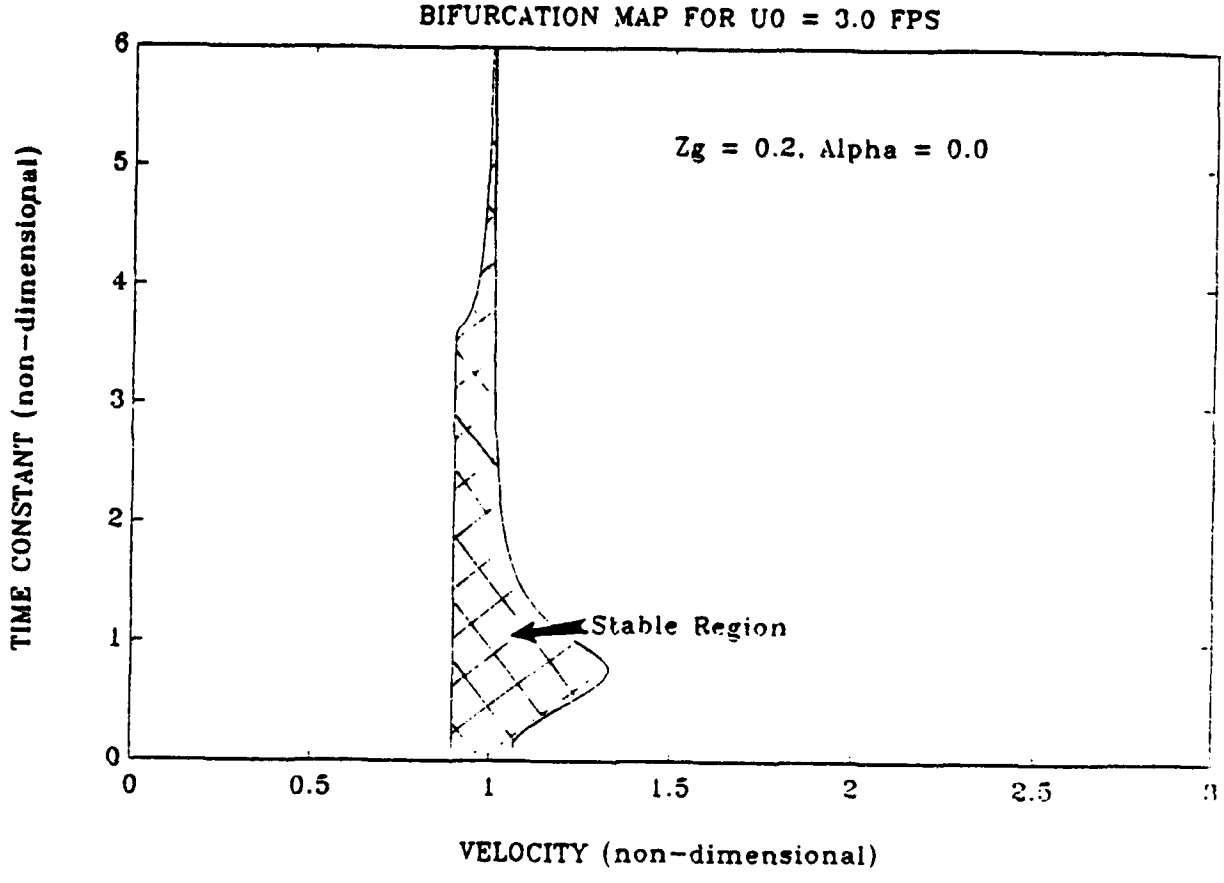

Figure (5.5) - This demonstrates the effects of an increase in the metacentric height for low nominal speeds.

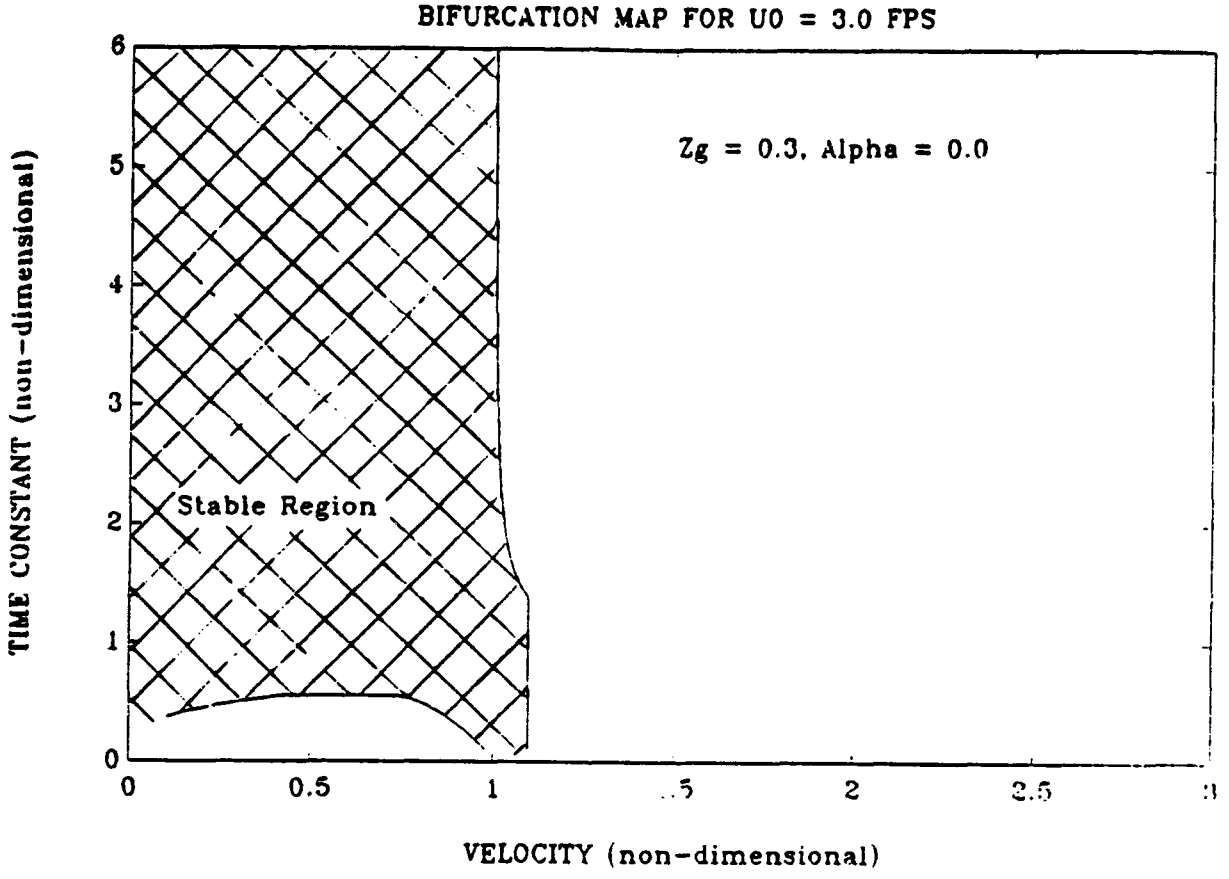

Figure (5.6) - As the metacentric height increases it has an adverse effect on the stability of the submarine.

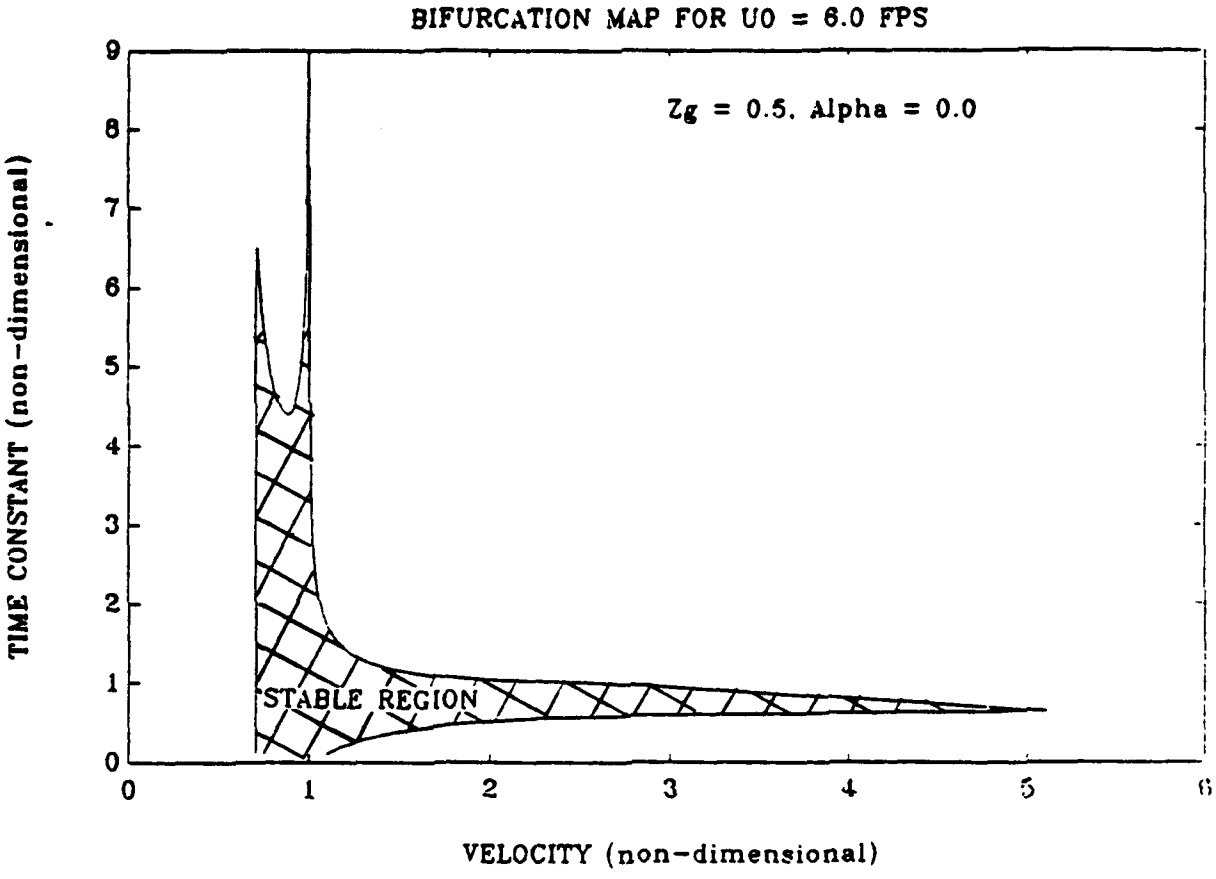

Figure (5.7) - An example of increasing the metacentric height for a higher nominal speed model.

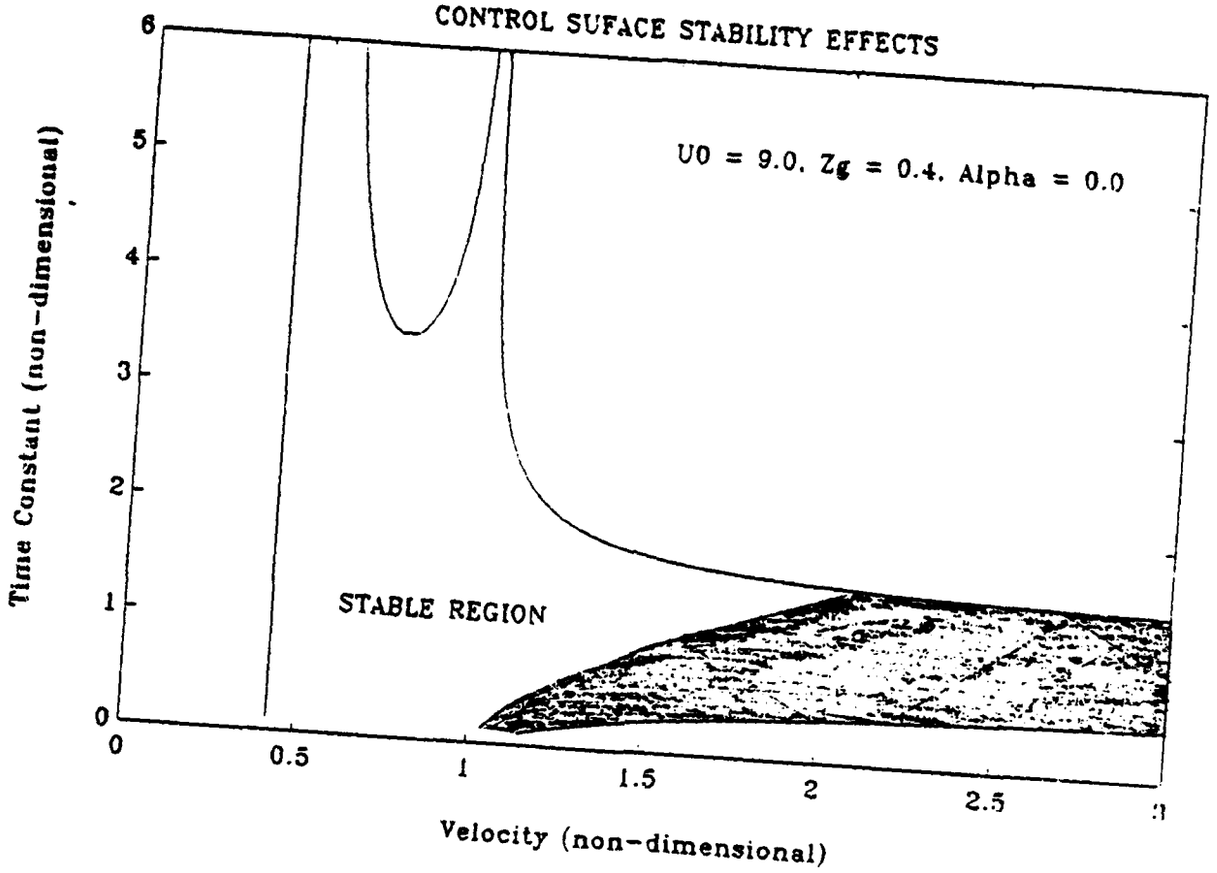

Figure (5.8) - The loss of stability due to using full moment and inertia coefficients vice the same coefficients x0.5.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The application of a Hopf bifurcation analysis to a submarine design can be an effective tool in the design evaluation and modification phase. These methods when paired with programs that generate hydrodynamic coefficients for a submarine will save time, effort, and money by reducing the amount of model testing necessary to validate a design. An effective set of control system design parameters will be generated in the process that will be optimal for the final design of a submersible.

Secondly, this system will give the limits of the range of metacentric heights that will maintain stability for the full range of speeds of the design. As was shown changes in the metacentric height can have a dramatic affect on stability.

Finally, an evaluation of the need for bow planes or forward control surfaces are actually to maintain control of the submarine. If the forward planes can be eliminated a potential source of noise would also be eliminated along with simplifying the design of the control system.

B. RECOMMENDATIONS

First, modify the programs to evaluate the performance of the submarine for the effects of external forces such as, wave effects, currents, and free surface effects.

Second, perform a systematic study of the bifurcations at conditions other than nominal in trim conditions, such as out of neutral buoyancy (heavy/light), or out-of-trim fore and aft conditions.

APPENDIX A - BIFURCATION ANALYSIS PROGRAM

```
C      PROGRAM BIFUR1.FOR
C      BIFURCATION ANALYSIS FOR THE DARPA SUBOFF MODEL
C
C      PARAMETERS ARE: TC VS. U
C
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       DOUBLE PRECISION K1,K2,K3,K4,L,MQDOT,
      &                 MWDOT,MQ,MW,MDS,MDB,MD,
      &                 MASS,IY,WF,pi
       DIMENSION A(4,4),FV1(4),IV1(4),ZZZ(4,4),WR(4),WI(4)
C
       OPEN (11,FILE='BIF1.RES',STATUS='NEW')
       OPEN (12,FILE='BIF2.RES',STATUS='NEW')
       OPEN (13,FILE='BIF3.RES',STATUS='NEW')
C
       WEIGHT=1556.2363
       BUO    =1556.2363
       L      =13.9792
       IY     =561.32
       G      =32.2
       MASS   =WEIGHT/G
       RHO    =1.94
       XB     =0.0
       ZB     =0.0
       pi     =3.14159
C
C      WRITE (*,*) 'ENTER MIN, MAX, AND # OF INCREMENTS IN TIME
                    CONSTANT'
C      READ  (*,*)  TCMIN,TCMAX,ITC
C      WRITE (*,*) 'ENTER MIN, MAX, AND # OF INCREMENTS IN
                    VEHICLE SPEED'
C      READ  (*,*)  UMIN,UMAX,IU
C      WRITE (*,*) 'ENTER NOMINAL SPEED'
C      READ  (*,*)  U0
C      WRITE (*,*) 'ENTER XG AND ZG'
C      READ  (*,*) XG,ZG
       TCMIN = 0.1
       TCMAX = 9.0
       ITC   = 200
       IU    = 200
       U0    = 12.0
       XG    = 0.0
       WRITE (*,*) 'ENTER MIN, MAX, OF VEHICLE SPEED'
       READ  (*,*)  UMIN,UMAX
       WRITE (*,*) 'ENTER ZG'
       READ  (*,*)  ZG
```

```
C
      ZGB=ZG-ZB
      TCMIN=TCMIN*L/U0
      TCMAX=TCMAX*L/U0
      UMIN =UMIN*U0
      UMAX =UMAX*U0
C
      ZQDOT=-6.3300E-04*0.5*RHO*L**4
      ZWDOT=-1.4529E-02*0.5*RHO*L**3
      ZQ    = 7.5450E-03*0.5*RHO*L**3
      ZW    =-1.3910E-02*0.5*RHO*L**2
      ZDS   =0.5*(-5.6030E-03*0.5*RHO*L**2)
      ZDB   =0.5*(-5.6030E-03*0.5*RHO*L**2)
      MQDOT=-8.8000E-04*0.5*RHO*L**5
      MWDOT=-5.6100E-04*0.5*RHO*L**4
      MQ    =-3.7020E-03*0.5*RHO*L**4
      MW    = 1.0324E-02*0.5*RHO*L**3
      MDS   =0.5*(-2.4090E-03*0.5*RHO*L**3)
      MDB   =0.5*( 2.4090E-03*0.5*RHO*L**3)
C
      ALPHA=-0.0
      ZD=ZDS+ALPHA*ZDB
      MD=MDS+ALPHA*MDB
C
      DV    =(MASS-ZWDOT)*(IY-MQDOT)
     &       -(MASS*XG+ZQDOT)*(MASS*XG+MWDOT)
      A11DV=(IY-MQDOT)*ZW+(MASS*XG+ZQDOT)*MW
      A12DV=(IY-MQDOT)*(MASS+ZQ)+(MASS*XG+ZQDOT)*(MQ-MASS*XG)
      A13DV=-(MASS*XG+ZQDOT)*WEIGHT
      B1DV =(IY-MQDOT)*ZD+(MASS*XG+ZQDOT)*MD
      A21DV=(MASS-ZWDOT)*MW+(MASS*XG+MWDOT)*ZW
      A22DV=(MASS-ZWDOT)*(MQ-MASS*XG)
     &      +(MASS*XG+MWDOT)*(MASS+ZQ)
      A23DV=-(MASS-ZWDOT)*WEIGHT
      B2DV =(MASS-ZWDOT)*MD+(MASS*XG+MWDOT)*ZD
C
      A11=A11DV/DV
      A12=A12DV/DV
      A13=A13DV/DV
      A21=A21DV/DV
      A22=A22DV/DV
      A23=A23DV/DV
      B1 =B1DV /DV
      B2 =B2DV /DV
C
      EPS  =1.D-5
      ILMAX=1500
```

```
C
      DO 1 I=1,ITC
        WRITE (*,2001) I,ITC
        TC=TCMIN+(I-1)*(TCMAX-TCMIN)/(ITC-1)
        POLE=1.0/TC
        ALPHA3=4.0*POLE
        ALPHA2=6.0*POLE**2
        ALPHA1=4.0*POLE**3
        ALPHA0=    POLE**4
        K4=ALPHA0/((B1*A21-B2*A11)*U0**4
     &             +(B1*A23-B2*A13)*ZGB*U0**2)
        A2M=B1*U0**2
        A3M=B2*U0**2
        A0M=-(A11+A22)*U0-ALPHA3
        B1M=B2*U0**2
        B2M=(B2*A12-B1*A22)*U0**3
        B3M=(B1*A21-B2*A11)*U0**3
        B0M=(A11*A22-A21*A12)*U0**2-A23*ZGB-ALPHA2-B1*U0*U0*K4
        C1M=(B2*A11-B1*A21)*U0**3
        C2M=(B1*A23-B2*A13)*ZGB*U0**2
        C0M=(A13*A21-A23*A11)*ZGB*U0+ALPHA1
     &      -(B2+B1*A22-B2*A12)*K4*U0**3
        K2=C1M*B0M*A3M-B1M*C0M*A3M-C1M*B3M*A0M
        K2=K2/(C1M*B2M*A3M-B1M*C2M*A3M-C1M*B3M*A2M)
        K1=(C0M-C2M*K2)/C1M
        K3=(A0M-A2M*K2)/A3M
C
        DO 2 J=1,IU
          U=UMIN+(J-1)*(UMAX-UMIN)/(IU-1)
          A(1,1)=0.0D0
          A(1,2)=0.0D0
          A(1,3)=1.0D0
          A(1,4)=0.0D0
          A(2,1)=A13*ZGB+B1*U*U*K1
          A(2,2)=A11*U  +B1*U*U*K2
          A(2,3)=A12*U  +B1*U*U*K3
          A(2,4)=        B1*U*U*K4
          A(3,1)=A23*ZGB+B2*U*U*K1
          A(3,2)=A21*U  +B2*U*U*K2
          A(3,3)=A22*U  +B2*U*U*K3
          A(3,4)=        B2*U*U*K4
          A(4,1)=-U
          A(4,2)=1.0D0
          A(4,3)=0.0D0
          A(4,4)=0.0D0
C
          CALL RG(4,4,A,WR,WI,0,ZZZ,IV1,FV1,IERR)
          CALL DSTABL(DEOS,WR,WI,FREQ)
```

```
C
      IF (J.GT.1) GO TO 10
      DEOSOO=DEOS
      UOO  =U
      LL=0
      GO TO 2
10    DEOSNN=DEOS
      UNN  =U
      PR=DEOSNN*DEOSOO
      IF (PR.GT.0.D0) GO TO 3
      LL=LL+1
      IF (LL.GT.3) STOP 1000
      IL=0
      UO=UOO
      UN=UNN
      DEOSO=DEOSOO
      DEOSN=DEOSNN
 6    UL=UO
      UR=UN
      DEOSL=DEOSO
      DEOSR=DEOSN
      U=(UL+UR)/2.D0
      A(1,1)=0.0D0
      A(1,2)=0.0D0
      A(1,3)=1.0D0
      A(1,4)=0.0D0
      A(2,1)=A13*ZGB+B1*U*U*K1
      A(2,2)=A11*U  +B1*U*U*K2
      A(2,3)=A12*U  +B1*U*U*K3
      A(2,4)=        B1*U*U*K4
      A(3,1)=A23*ZGB+B2*U*U*K1
      A(3,2)=A21*U  +B2*U*U*K2
      A(3,3)=A22*U  +B2*U*U*K3
      A(3,4)=        B2*U*U*K4
      A(4,1)=-U
      A(4,2)=1.0D0
      A(4,3)=0.0D0
      A(4,4)=0.0D0
C
      CALL RG(4,4,A,WR,WI,0,ZZZ,IV1,FV1,IERR)
      CALL DSTABL(DEOS,WR,WI,FREQ)
C
      DEOSM=DEOS
      UM=U
      PRL=DEOSL*DEOSM
      PRR=DEOSR*DEOSM
      IF (PRL.GT.0.D0) GO TO 5
      UO=UL
      UN=UM
      DEOSO=DEOSL
      DEOSN=DEOSM
```

```
            IL=IL+1
            IF (IL.GT.ILMAX) STOP 3100
            DIF=DABS(UL-UM)
            IF (DIF.GT.EPS) GO TO 6
            U=UM
            GO TO 4
    5       IF (PRR.GT.0.D0) STOP 3200
            UO=UM
            UN=UR
            DEOSO=DEOSM
            DEOSN=DEOSR
            IL=IL+1
            IF (IL.GT.ILMAX) STOP 3100
            DIF=DABS(UM-UR)
            IF (DIF.GT.EPS) GO TO 6
            U=UM
    4       LLL=10+LL
            WRITE (LLL,2002) U/U0,TC*U0/L
    3       UOO=UNN
            DEOSOO=DEOSNN
    2     CONTINUE
    1   CONTINUE
C
 2001   FORMAT (2I5)
 2002   FORMAT (4(3X,F10.6))
        END
C
        SUBROUTINE DSTABL(DEOS,WR,WI,OMEGA)
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DIMENSION WR(4),WI(4)
        DEOS=-1.0D+20
        DO 1 I=1,4
          IF (WR(I).LT.DEOS) GO TO 1
          DEOS=WR(I)
          IJ=I
    1   CONTINUE
        OMEGA=WI(IJ)
        OMEGA=DABS(OMEGA)
        RETURN
        END
```

APPENDIX B - HOPF BIFURCATION PROGRAM

```
C      PROGRAM HOPF1A.FOR
C     EVALUATION OF HOPF BIFURCATION FORMULAE USING THE SUBOFF
C      SUBMARINE MODEL
C
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       DOUBLE PRECISION L,IY,MASS,MQDOT,MWDOT,
      1                 MQ,MW,MD,MDS,MDB,K1,K2,K3,K4,
      2                 ALFA1,ALFA2,ALFA3,
      3                 BETA1,BETA2,BETA3
       DOUBLE PRECISION M11,M12,M13,M14,M21,M22,M23,M24,
      1                 M31,M32,M33,M34,M41,M42,M43,M44,
      2                 N11,N12,N13,N14,N21,N22,N23,N24,
      3                 N31,N32,N33,N34,N41,N42,N43,N44,
      4                 L21,L22,L23,L24,L31,L32,L33,L34,
      5                 L41,L42,L43,L44,L25,L26,L27,L35,
      6                 L36,L37,L21A,L22A,L23A,L24A,L31A,
      7                 L32A,L33A,L34A
C
       DIMENSION A(4,4),T(4,4),TINV(4,4),FV1(4),IV1(4),YYY(4,4)
       DIMENSION WR(4),WI(4),TSAVE(4,4),
      1          TLUD(4,4),IVLUD(4),SVLUD(4)
       DIMENSION ASAVE(4,4),A1(4,4),A2(4,4)
C
       OPEN (10,FILE='HOPF.DAT',STATUS='OLD')
       OPEN (20,FILE='HOPF.RES',STATUS='NEW')
C
       WEIGHT=1556.2363
       IY     = 561.32
       L      =  13.9792
       RHO    =   1.94
       G      =  32.2
       XG     =   0.0
       ZB     =   0.0
       MASS   =WEIGHT/G
       BOY    =WEIGHT
       ZQDOT =-6.3300E-04*0.5*RHO*L**4
       ZWDOT =-1.4529E-02*0.5*RHO*L**3
       ZQ     = 7.5450E-03*0.5*RHO*L**3
       ZW     =-1.3910E-02*0.5*RHO*L**2
       ZDS    =0.5*(-5.6030E-03*0.5*RHO*L**2)
       ZDB    =0.5*(-5.6030E-03*0.5*RHO*L**2)
       MQDOT =-8.8000E-04*0.5*RHO*L**5
       MWDOT =-5.6100E-04*0.5*RHO*L**4
       MQ     =-3.7020E-03*0.5*RHO*L**4
       MW     = 1.0324E-02*0.5*RHO*L**3
       MDS    =0.5*(-2.4090E-03*0.5*RHO*L**3)
```

```
      MDB    =0.5*( 2.4090E-03*0.5*RHO*L**3)
C
      WRITE (*,1001)
      READ  (*,*)     ITOTAL
C     WRITE (*,1002)
C     READ  (*,*)     U0,ZG,DSAT
      WRITE (*,1004)
      READ  (*,*)     ZG
      WRITE (*,1003)
      READ  (*,*)     ALPHA
      ZGB=ZG-ZB
      U0     = 9.0
      DSAT   = 0.4
C
      ZD=ZDS+ALPHA*ZDB
      MD=MDS+ALPHA*MDB
C
C     DETERMINE [A] AND [B] COEFFICIENTS

C
      DV    =(MASS-ZWDOT)*(IY-MQDOT)
     &        -(MASS*XG+ZQDOT)*(MASS*XG+MWDOT)
      A11DV=(IY-MQDOT)*ZW+(MASS*XG+ZQDOT)*MW
      A12DV=(IY-MQDOT)*(MASS+ZQ)+(MASS*XG+ZQDOT)*(MQ-MASS*XG)
      A13DV=-(MASS*XG+ZQDOT)*WEIGHT
      B1DV =(IY-MQDOT)*ZD+(MASS*XG+ZQDOT)*MD
      A21DV=(MASS-ZWDOT)*MW+(MASS*XG+MWDOT)*ZW
      A22DV=(MASS-ZWDOT)*(MQ-MASS*XG)
     &       +(MASS*XG+MWDOT)*(MASS+ZQ)
      A23DV=-(MASS-ZWDOT)*WEIGHT
      B2DV =(MASS-ZWDOT)*MD+(MASS*XG+MWDOT)*ZD

C
      A11=A11DV/DV
      A12=A12DV/DV
      A13=A13DV/DV
      A21=A21DV/DV
      A22=A22DV/DV
      A23=A23DV/DV
      B1 =B1DV/DV
      B2 =B2DV/DV
C
      C11DV=(IY-MQDOT)*MASS*ZG
      C12DV=-(MASS*XG+ZQDOT)*MASS*ZG
      C21DV=-(MASS-ZWDOT)*MASS*ZG
      C22DV=(MASS*XG+MWDOT)*MASS*ZG
C
      C11=C11DV/DV
      C12=C12DV/DV
      C21=C21DV/DV
      C22=C22DV/DV
```

```
C
      DO 1 IT=1,ITOTAL
        WRITE (*,3001) IT,ITOTAL
        READ (10,*) U,TC
        TC=TC*L/U0
        U=U*U0
C
C       CALCULATE THE GAINS
C
        POLE=1.0/TC
        ALPHA3=4.0*POLE
        ALPHA2=6.0*POLE**2
        ALPHA1=4.0*POLE**3
        ALPHA0=    POLE**4
        K4=ALPHA0/((B1*A21-B2*A11)*U0**4
     &             +(B1*A23-B2*A13)*ZGB*U0**2)
        A2M=B1*U0**2
        A3M=B2*U0**2
        A0M=-(A11+A22)*U0-ALPHA3
        B1M=B2*U0**2
        B2M=(B2*A12-B1*A22)*U0**3
        B3M=(B1*A21-B2*A11)*U0**3
        B0M=(A11*A22-A21*A12)*U0**2-A23*ZGB-ALPHA2-B1*U0*U0*K4
        C1M=(B2*A11-B1*A21)*U0**3
        C2M=(B1*A23-B2*A13)*ZGB*U0**2
        C0M=(A13*A21-A23*A11)*ZGB*U0+ALPHA1
     &      -(B2+B1*A22-B2*A12)*K4*U0**3
        K2=C1M*B0M*A3M-B1M*C0M*A3M-C1M*B3M*A0M
        K2=K2/(C1M*B2M*A3M-B1M*C2M*A3M-C1M*B3M*A2M)
        K1=(C0M-C2M*K2)/C1M
        K3=(A0M-A2M*K2)/A3M

C
C       EVALUATE NONLINEAR RUDDER EXPANSION COEFFICIENTS
C
        DTTW=-(1.D0/(3.D0*DSAT**2))*3.D0*K1*K1*K2
        DTWW=-(1.D0/(3.D0*DSAT**2))*3.D0*K1*K2*K2
        DTTQ=-(1.D0/(3.D0*DSAT**2))*3.D0*K1*K1*K3
        DTQQ=-(1.D0/(3.D0*DSAT**2))*3.D0*K1*K3*K3
        DTTZ=-(1.D0/(3.D0*DSAT**2))*3.D0*K1*K1*K4
        DTZZ=-(1.D0/(3.D0*DSAT**2))*3.D0*K1*K4*K4
        DWWQ=-(1.D0/(3.D0*DSAT**2))*3.D0*K2*K2*K3
        DWQQ=-(1.D0/(3.D0*DSAT**2))*3.D0*K2*K3*K3
        DWWZ=-(1.D0/(3.D0*DSAT**2))*3.D0*K4*K2*K2
        DWZZ=-(1.D0/(3.D0*DSAT**2))*3.D0*K4*K4*K2
        DQQZ=-(1.D0/(3.D0*DSAT**2))*3.D0*K4*K3*K3
        DQZZ=-(1.D0/(3.D0*DSAT**2))*3.D0*K4*K4*K3
        DTWQ=-(1.D0/(3.D0*DSAT**2))*6.D0*K1*K2*K3
        DTWZ=-(1.D0/(3.D0*DSAT**2))*6.D0*K1*K4*K2
        DTQZ=-(1.D0/(3.D0*DSAT**2))*6.D0*K1*K4*K3
        DWQZ=-(1.D0/(3.D0*DSAT**2))*6.D0*K4*K2*K3
```

```
      DTTT=-(1.D0/(3.D0*DSAT**2))*1.D0*K1*K1*K1
      DWWW=-(1.D0/(3.D0*DSAT**2))*1.D0*K2*K2*K2
      DQQQ=-(1.D0/(3.D0*DSAT**2))*1.D0*K3*K3*K3
      DZZZ=-(1.D0/(3.D0*DSAT**2))*1.D0*K4*K4*K4
C
C     EVALUATE TRANSFORMATION MATRIX OF EIGENVECTORS
C
      A(1,1)=0.0D0
      A(1,2)=0.0D0
      A(1,3)=1.0D0
      A(1,4)=0.0D0
      A(2,1)=A13*ZGB+B1*U*U*K1
      A(2,2)=A11*U  +B1*U*U*K2
      A(2,3)=A12*U  +B1*U*U*K3
      A(2,4)=        B1*U*U*K4
      A(3,1)=A23*ZGB+B2*U*U*K1
      A(3,2)=A21*U  +B2*U*U*K2
      A(3,3)=A22*U  +B2*U*U*K3
      A(3,4)=        B2*U*U*K4
      A(4,1)=-U
      A(4,2)=1.0D0
      A(4,3)=0.0D0
      A(4,4)=0.0D0
      DO 11 I=1,4
        DO 12 J=1,4
          ASAVE(I,J)=A(I,J)
   12   CONTINUE
   11 CONTINUE
      CALL RG(4,4,A,WR,WI,1,YYY,IV1,FV1,IERR)
      CALL DSOMEG(IEV,WR,WI,OMEGA,CHECK)
      OMEGA0=OMEGA
      DO 5 I=1,4
        T(I,1)= YYY(I,IEV)
        T(I,2)=-YYY(I,IEV+1)
    5 CONTINUE
      IF (IEV.EQ.1) GO TO 13
      IF (IEV.EQ.2) GO TO 14
      IF (IEV.EQ.3) GO TO 15
      STOP 3004
   14 DO 6 I=1,4
        T(I,3)=YYY(I,1)
        T(I,4)=YYY(I,4)
    6 CONTINUE
      GO TO 17
   15 DO 7 I=1,4
        T(I,3)=YYY(I,1)
        T(I,4)=YYY(I,2)
    7 CONTINUE
      GO TO 17
   13 DO 16 I=1,4
        T(I,3)=YYY(I,3)
```

```
            T(I,4)=YYY(I,4)
   16     CONTINUE
   17     CONTINUE
C
C     CHECK THE EIGENVALUES
C
          WRITE(21,2003) TC*U0/L,WI(1),WI(2),WI(3),WI(4),
     &                               WR(1),WR(2),WR(3),WR(4)
C
C         NORMALIZATION OF THE CRITICAL EIGENVECTOR
C
          INORM=1
          IF (INORM.NE.0) CALL NORMAL(T)
C
C         INVERT TRANSFORMATION MATRIX
C
          DO 2 I=1,4
            DO 3 J=1,4
              TINV(I,J)=0.0
              TSAVE(I,J)=T(I,J)
    3       CONTINUE
    2     CONTINUE
          CALL DLUD(4,4,TSAVE,4,TLUD,IVLUD)
          DO 4 I=1,4
            IF (IVLUD(I).EQ.0) STOP 3003
    4     CONTINUE
          CALL DILU(4,4,TLUD,IVLUD,SVLUD)
          DO 8 I=1,4
            DO 9 J=1,4
              TINV(I,J)=TLUD(I,J)
    9       CONTINUE
    8     CONTINUE
C
C         CHECK Inv(T)*A*T
C
          IMULT=1
          IF (IMULT.EQ.1) CALL MULT(TINV,ASAVE,T,A2)
          IF (IMULT.EQ.0) STOP
          P=A2(3,3)
          Q=A2(4,4)
C         WRITE(21,*) P,Q
C
C         DEFINITION OF Nij
C
          N11=TINV(1,1)
          N21=TINV(2,1)
          N31=TINV(3,1)
          N41=TINV(4,1)
          N12=TINV(1,2)
          N22=TINV(2,2)
          N32=TINV(3,2)
```

```
      N42=TINV(4,2)
      N13=TINV(1,3)
      N23=TINV(2,3)
      N33=TINV(3,3)
      N43=TINV(4,3)
      N14=TINV(1,4)
      N24=TINV(2,4)
      N34=TINV(3,4)
      N44=TINV(4,4)
C
C     DEFINITION OF Mij
C
      M11=T(1,1)
      M21=T(2,1)
      M31=T(3,1)
      M41=T(4,1)
      M12=T(1,2)
      M22=T(2,2)
      M32=T(3,2)
      M42=T(4,2)
      M13=T(1,3)
      M23=T(2,3)
      M33=T(3,3)
      M43=T(4,3)
      M14=T(1,4)
      M24=T(2,4)
      M34=T(3,4)
      M44=T(4,4)
C
C     DEFINITION OF Lij
C
      L25=C11*M31*M31+C12*M21*M31
C
      L26=2*C11*M31*M32+C12*(M21*M32+M22*M31)
C
      L27=C11*M32*M32+C12*M22*M32
C
      L35=C22*M31*M31+321*M21*M31
C
      L36=2*C22*M31*M32+C21*(M21*M32+M22*M31)
C
      L37=C22*M32*M32+C21*M22*M32
C
      L21= DTTW*M11*M11*M21 + DTWW*M11*M21*M21 +
     &     DTTQ*M11*M11*M31 + DTQQ*M11*M31*M31 +
     &     DTTZ*M11*M11*M41 + DTZZ*M11*M41*M41 +
     &     DWWQ*M21*M21*M31 + DWQQ*M21*M31*M31 +
     &     DWWZ*M21*M21*M41 + DWZZ*M21*M41*M41 +
     &     DQQZ*M31*M31*M41 + DQZZ*M31*M41*M41 +
     &     DTWQ*M11*M21*M31 + DTWZ*M11*M21*M41 +
     &     DTQZ*M11*M41*M31 + DWQZ*M21*M41*M31 +
```

```
     &        DTTT*M11*M11*M11 + DWWW*M21*M21*M21 +
     &        DQQQ*M31*M31*M31 + DZZZ*M41*M41*M41
C
         TTW=M11*M11*M22+2.0*M11*M12*M21
         TWW=M12*M21*M21+2.0*M11*M21*M22
         TTQ=M11*M11*M32+2.0*M11*M12*M31
         TQQ=M12*M31*M31+2.0*M11*M31*M32
         TTZ=M11*M11*M42+2.0*M11*M12*M41
         TZZ=M41*M41*M12+2.0*M11*M41*M42
         WWQ=M21*M21*M32+2.0*M31*M21*M22
         WQQ=M22*M31*M31+2.0*M31*M32*M21
         WWZ=M21*M21*M42+2.0*M41*M21*M22
         WZZ=M22*M41*M41+2.0*M41*M42*M21
         QQZ=M31*M31*M42+2.0*M41*M31*M32
         QZZ=M32*M41*M41+2.0*M41*M42*M31
         TWQ=M11*M21*M32+M31*(M11*M22+M12*M21)
         TWZ=M11*M21*M42+M41*(M11*M22+M12*M21)
         TQZ=M11*M41*M32+M31*(M11*M42+M12*M41)
         WQZ=M21*M41*M32+M31*(M21*M42+M22*M41)
         TTT=3.0*M11*M11*M12
         WWW=3.0*M21*M21*M22
         QQQ=3.0*M31*M31*M32
         ZZZ=3.0*M41*M41*M42
C
         L22=DTTW*TTW+DTWW*TWW+DTTQ*TTQ+DTQQ*TQQ+
     &       DTTZ*TTZ+DTZZ*TZZ+DWWQ*WWQ+DWQQ*WQQ+
     &       DWWZ*WWZ+DWZZ*WZZ+DQQZ*QQZ+DQZZ*QZZ+
     &       DTWQ*TWQ+DTWZ*TWZ+DTQZ*TQZ+DWQZ*WQZ+
     &       DTTT*TTT+DWWW*WWW+DQQQ*QQQ+DZZZ*ZZZ
C
         TTW=M12*M12*M21+2.0*M11*M12*M22
         TWW=M11*M22*M22+2.0*M12*M21*M22
         TTQ=M12*M12*M31+2.0*M11*M12*M32
         TQQ=M11*M32*M32+2.0*M12*M31*M32
         TTZ=M12*M12*M41+2.0*M11*M12*M42
         TZZ=M11*M42*M42+2.0*M12*M41*M42
         WWQ=M22*M22*M31+2.0*M21*M22*M32
         WQQ=M21*M32*M32+2.0*M22*M31*M32
         WWZ=M22*M22*M41+2.0*M21*M22*M42
         WZZ=M21*M42*M42+2.0*M22*M41*M42
         QQZ=M32*M32*M41+2.0*M31*M32*M42
         QZZ=M31*M42*M42+2.0*M32*M41*M42
         TWQ=M12*M22*M31+M32*(M11*M22+M12*M21)
         TWZ=M12*M22*M41+M42*(M11*M22+M12*M21)
         TQZ=M12*M42*M31+M32*(M11*M42+M12*M41)
         WQZ=M22*M42*M31+M32*(M21*M42+M22*M41)
         TTT=3.0*M11*M12*M12
         WWW=3.0*M21*M22*M22
         QQQ=3.0*M31*M32*M32
         ZZZ=3.0*M41*M42*M42
C
```

```
      L23=DTTW*TTW+DTWW*TWW+DTTQ*TTQ+DTQQ*TQQ+
     &    DTTZ*TTZ+DTZZ*TZZ+DWWQ*WWQ+DWQQ*WQQ+
     &    DWWZ*WWZ+DWZZ*WZZ+DQQZ*QQZ+DQZZ*QZZ+
     &    DTWQ*TWQ+DTWZ*TWZ+DTQZ*TQZ+DWQZ*WQZ+
     &    DTTT*TTT+DWWW*WWW+DQQQ*QQQ+DZZZ*ZZZ
C
      L24= DTTW*M12*M12*M22 + DTWW*M12*M22*M22 +
     &     DTTQ*M12*M12*M32 + DTQQ*M12*M32*M32 +
     &     DTTZ*M12*M12*M42 + DTZZ*M12*M42*M42 +
     &     DWWQ*M22*M22*M32 + DWQQ*M22*M32*M32 +
     &     DWWZ*M22*M22*M42 + DWZZ*M22*M42*M42 +
     &     DQQZ*M32*M32*M42 + DQZZ*M32*M42*M42 +
     &     DTWQ*M12*M22*M32 + DTWZ*M12*M22*M42 +
     &     DTQZ*M12*M42*M32 + DWQZ*M22*M42*M32 +
     &     DTTT*M12*M12*M12 + DWWW*M22*M22*M22 +
     &     DQQQ*M32*M32*M32 + DZZZ*M42*M42*M42
C
      L31=L21
      L32=L22
      L33=L23
      L34=L24
C
C     DEFINITION OF ALFA(I) AND BETA(I)
C

      D1 =N32*L25 + N33*L35
      D2 =N32*L26 + N33*L36
      D3 =N32*L27 + N33*L37
C
      D11=-P
      D12=OMEGA0
      D21=-2*OMEGA0
      D22=-P
      D23=2*OMEGA0
      D32=-OMEGA0
      D33=-P
C
      ALFA2=(D2-D21*D1/D11-D23*D3/D33)/
     &      (D22-D21*D12/D11-D23*D32/D33)
      ALFA1=(D1-D12*ALFA2)/D11
      ALFA3=(D3-D32*ALFA2)/D33
C
      D1 =N42*L25 + N43*L35
      D2 =N42*L26 + N43*L36
      D3 =N42*L27 + N43*L37
C
      D11=-Q
      D12=OMEGA0
      D21=-2*OMEGA0
```

```
      D22=-Q
      D23=2*OMEGA0
      D32=-OMEGA0
      D33=-Q
C
      BETA2=(D2-D21*D1/D11-D23*D3/D33)/
     &      (D22-D21*D12/D11-D23*D32/D33)
      BETA1=(D1-D12*BETA2)/D11
      BETA3=(D3-D32*BETA2)/D33
C
      L21A=2*C11*(ALFA1*M31*M33+BETA1*M31*M34)+
     &      C12*(ALFA1*(M21*M33+M23*M31)+
     &      BETA1*(M21*M34+M24*M31))
C
      L22A=2*C11*(ALFA1*M32*M33+BETA1*M32*M34)
     &     +2*C11*(ALFA3*M31*M33+BETA3*M31*M34)
     &     +C12*(ALFA1*(M22*M33+M32*M23)
     &     +BETA1*(M22*M34+M24*M32))
     &     +C12*(ALFA3*(M21*M33+M23*M31)
     &     +BETA3*(M21*M34+M24*M31))
C
      L23A=2*C11*(ALFA2*M31*M33+BETA2*M31*M34)
     &     +2*C11*(ALFA3*M32*M33+BETA3*M32*M34)
     &     +C12*(ALFA2*(M21*M33+M23*M31)
     &     +BETA2*(M21*M34+M24*M31))
     &     +C12*(ALFA3*(M22*M33+M23*M32)
     &     +BETA3*(M22*M34+M24*M32))
C
      L24A=2*C11*(ALFA2*M32*M33+BETA2*M32*M34)
     &      +C12*(ALFA2*(M22*M33+M23*M32)
     &      +BETA2*(M22*M34+M24*M32))
C
      L31A=2*C22*(ALFA1*M31*M33+BETA1*M31*M34)
     &     +C21*(ALFA1*(M21*M33+M23*M31)
     &     +BETA1*(M21*M34+M24*M31))
C

      L32A=2*C22*(ALFA1*M32*M33+BETA1*M32*M34)
     &     +2*C22*(ALFA3*M31*M33+BETA3*M31*M34)
     &     +C21*(ALFA1*(M22*M33+M32*M23)
     &     +BETA1*(M22*M34+M24*M32))
     &     +C21*(ALFA3*(M21*M33+M23*M31)
     &     +BETA3*(M21*M34+M24*M31))
C
```

```
      L33A=2*C22*(ALFA2*M31*M33+BETA2*M31*M34)
     &     +2*C22*(ALFA3*M32*M33+BETA3*M32*M34)
     &     +C21*(ALFA2*(M21*M33+M23*M31)
     &     +BETA2*(M21*M34+M24*M31))
     &     +C21*(ALFA3*(M22*M33+M23*M32)
     &     +BETA3*(M22*M34+M24*M32))
C
      L34A=2*C22*(ALFA2*M32*M33+BETA2*M32*M34)
     &      +C21*(ALFA2*(M22*M33+M23*M32)
     &      +BETA2*(M22*M34+M24*M32))
C
      L21=L21A+L21*B1*U*U-A13*ZGB*M11**3/6.D0
      L22=L22A+L22*B1*U*U-A13*ZGB*M12*M11**2/2.D0
      L23=L23A+L23*B1*U*U-A13*ZGB*M11*M12**2/2.D0
      L24=L24A+L24*B1*U*U-A13*ZGB*M12**3/6.D0
      L31=L31A+L31*B2*U*U-A23*ZGB*M11**3/6.D0
      L32=L32A+L32*B2*U*U-A23*ZGB*M12*M11**2/2.D0
      L33=L33A+L33*B2*U*U-A23*ZGB*M11*M12**2/2.D0
      L34=L34A+L34*B2*U*U-A23*ZGB*M12**3/6.D0
      L41=-0.5*M11*M11*(M21-U*M11/3.0)
      L42=-M11*(M12*M21+0.5*M11*M22-0.5*U*M11*M12)
      L43=-M12*(M11*M22+0.5*M12*M21-0.5*U*M11*M12)
      L44=-0.5*M12*M12*(M22-U*M12/3.0)
C
      R11=N12*L21+N13*L31+N14*L41
      R12=N12*L22+N13*L32+N14*L42
      R13=N12*L23+N13*L33+N14*L43
      R14=N12*L24+N13*L34+N14*L44
      R21=N22*L21+N23*L31+N24*L41
      R22=N22*L22+N23*L32+N24*L42
      R23=N22*L23+N23*L33+N24*L43
      R24=N22*L24+N23*L34+N24*L44
C
C     EVALUATE DALPHA AND DOMEGA
C
      UINC=0.001
      UR =U+UINC
      UL =U-UINC
      U  =UR
C
      A(1,1)=0.0D0
      A(1,2)=0.0D0
      A(1,3)=1.0D0
      A(1,4)=0.0D0
      A(2,1)=A13*ZGB+B1*U*U*K1
      A(2,2)=A11*U  +B1*U*U*K2
      A(2,3)=A12*U  +B1*U*U*K3
      A(2,4)=        B1*U*U*K4
      A(3,1)=A23*ZGB+B2*U*U*K1
      A(3,2)=A21*U  +B2*U*U*K2
```

```
      A(3,3)=A22*U  +B2*U*U*K3
      A(3,4)=        B2*U*U*K4
      A(4,1)=-U
      A(4,2)=1.0D0
      A(4,3)=0.0D0
      A(4,4)=0.0D0
C
      CALL RG(4,4,A,WR,WI,0,YYY,IV1,FV1,IERR)
      CALL DSTABL(DEOS,WR,WI,FREQ)
      ALPHR=DEOS
      OMEGR=FREQ
C
      U=UL
C
      A(1,1)=0.0D0
      A(1,2)=0.0D0
      A(1,3)=1.0D0
      A(1,4)=0.0D0
      A(2,1)=A13*ZGB+B1*U*U*K1
      A(2,2)=A11*U  +B1*U*U*K2
      A(2,3)=A12*U  +B1*U*U*K3
      A(2,4)=        B1*U*U*K4
      A(3,1)=A23*ZGB+B2*U*U*K1
      A(3,2)=A21*U  +B2*U*U*K2
      A(3,3)=A22*U  +B2*U*U*K3
      A(3,4)=        B2*U*U*K4
      A(4,1)=-U
      A(4,2)=1.0D0
      A(4,3)=0.0D0
      A(4,4)=0.0D0
C
      CALL RG(4,4,A,WR,WI,0,YYY,IV1,FV1,IERR)
      CALL DSTABL(DEOS,WR,WI,FREQ)
      ALPHL=DEOS
      OMEGL=FREQ
C
      DALPHA=(ALPHR-ALPHL)/(UR-UL)
      DOMEGA=(OMEGR-OMEGL)/(UR-UL)
C
C     EVALUATION OF HOPF BIFURCATION COEFFICIENTS
C
      COEF1=3.0*R11+R13+R22+3.0*R24
      COEF2=3.0*R21+R23-R12-3.0*R14
      AMU2 =-COEF1/(8.0*DALPHA)
      BETA2=0.25*COEF1
C     TAU2 =-(COEF2-DOMEGA*COEF1/DALPHA)/(8.0*OMEGA0)
      PER  =2.0*3.1415927/OMEGA0
      PER  =PER*U/L
      WRITE (20,2001) TC*U0/L,COEF1,DALPHA,
     &                PER,AMU2,TAU2,DOMEGA
    1 CONTINUE
```

```
      STOP
 1001 FORMAT (' ENTER NUMBER OF DATA LINES')
 1002 FORMAT (' ENTER U0, ZG, AND DSAT')
 1003 FORMAT (' ENTER BOW PLANE TO STERN PLANE RATIO')
 1004 FORMAT (' ENTER ZG')
 2001 FORMAT (7E14.5)
 2002 FORMAT (6E14.5)
 2003 FORMAT (9F11.5)
 3001 FORMAT (2I5)
      END
C=============================================================
      SUBROUTINE DSOMEG(IJK,WR,WI,OMEGA,CHECK)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION WR(4),WI(4)
      CHECK=-1.0D+25
      DO 1 I=1,4
        IF (WR(I).LT.CHECK) GO TO 1
        CHECK=WR(I)
        IJ=I
    1 CONTINUE
      OMEGA=DABS(WI(IJ))
      IF (WI(IJ).GT.0.D0) IJK=IJ
      IF (WI(IJ).LT.0.D0) IJK=IJ-1
      RETURN
      END
C=============================================================
      SUBROUTINE DSTABL(DEOS,WR,WI,OMEGA)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION WR(4),WI(4)
      DEOS=-1.0D+20
      DO 1 I=1,4
        IF (WR(I).LT.DEOS) GO TO 1
        DEOS=WR(I)
        IJ=I
    1 CONTINUE
      OMEGA=WI(IJ)
      OMEGA=DABS(OMEGA)
      RETURN
      END
C=============================================================
      SUBROUTINE NORMAL(T)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION T(4,4),TNOR(4,4)
      CFAC=T(1,1)**2+T(1,2)**2
      IF (DABS(CFAC).LE.(1.D-10)) STOP 4001
      TNOR(1,1)=1.D0
      TNOR(2,1)=(T(1,1)*T(2,1)+T(2,2)*T(1,2))/CFAC
      TNOR(3,1)=(T(1,1)*T(3,1)+T(3,2)*T(1,2))/CFAC
      TNOR(4,1)=(T(1,1)*T(4,1)+T(4,2)*T(1,2))/CFAC
      TNOR(1,2)=0.D0
      TNOR(2,2)=(T(2,2)*T(1,1)-T(2,1)*T(1,2))/CFAC
```

```
      TNOR(3,2)=(T(3,2)*T(1,1)-T(3,1)*T(1,2))/CFAC
      TNOR(4,2)=(T(4,2)*T(1,1)-T(4,1)*T(1,2))/CFAC
      DO 1 I=1,4
        DO 2 J=1,2
          T(I,J)=TNOR(I,J)
    2   CONTINUE
    1 CONTINUE
      RETURN
      END
C=======================================================
      SUBROUTINE MULT(TINV,A,T,A2)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION TINV(4,4),A(4,4),T(4,4),A1(4,4),A2(4,4)
      DO 1 I=1,4
        DO 2 J=1,4
          A1(I,J)=0.D0
          A2(I,J)=0.D0
    2   CONTINUE
    1 CONTINUE
      DO 3 I=1,4
        DO 4 J=1,4
          DO 5 K=1,4
            A1(I,J)=A(I,K)*T(K,J)+A1(I,J)
    5     CONTINUE
    4   CONTINUE
    3 CONTINUE
      DO 6 I=1,4
        DO 7 J=1,4
          DO 8 K=1,4
            A2(I,J)=TINV(I,K)*A1(K,J)+A2(I,J)
    8     CONTINUE
    7   CONTINUE
    6 CONTINUE
      DO 11 I=1,4
C        WRITE (*,101) (A(I,J),J=1,4)
   11 CONTINUE
      DO 12 I=1,4
C        WRITE (*,101) (T(I,J),J=1,4)
   12 CONTINUE
      DO 10 I=1,4
C       WRITE (*,101) (A2(I,J),J=1,4)
   10 CONTINUE
C      WRITE (*,101) A2(1,1)
      RETURN
  101 FORMAT (4E15.5)
      END
```

APPENDIX C - ADAMS-BASHFORTH SIMULATION PROGRAM

```
C      PROGRAM ABSIM1.FOR
C
C      SUBOFF SIMULATION PROGRAM USING A FOURTH ORDER
C      ADAMS-BASHFORTH INTEGRATION SCHEME
C
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       INTEGER LDA,NA
       PARAMETER (NA=4,LDA=4)
       DOUBLE PRECISION L,MW,MASS,IY,MQDOT,MWDOT,MQ,ICP,LEN,IW,
      *                 MDS,MDB,MDELTA,DELTA,U0,TIME,TC,K1,K2,
      *                  K3,K4,F1(4),F2(4),F3(4),F4(4),
      *                  A(LDA,NA),ISAT
       COMPLEX EVAL(NA)
C
       DIMENSION XL(25),BR(25),VEC1(25),VEC2(25)
C
       OPEN (10,FILE='SIM.DAT',STATUS='OLD')
       OPEN (30,FILE='ANG.DAT',STATUS='OLD')
       OPEN (20,FILE='SIM.RES',STATUS='NEW')
C
C      ENTER SPEEDS AND TIME DATA
C
       READ(10,*) U0,U,TC,TSIM,DELT,IPRNT,ALPHA
       U=U*U0
C
C      GEOMETRIC PROPERTIES AND HYDRODYNAMIC COEFFICIENTS
C
       DSAT  =0.4000D0
       PI    =4.0*DATAN(1.0D0)
       RHO   =1.94D0
       CDZ   =0.5000D0
       L     =13.9792D0
       WEIGHT=1556.2363D0
       IY    =561.32D0
       MASS  =WEIGHT/32.2
       TC    =TC*L/U0
       ZQDOT=-6.3300E-04*0.5*RHO*L**4
       ZWDOT=-1.4529E-02*0.5*RHO*L**3
       ZQ    = 7.5450E-03*0.5*RHO*L**3
       ZW    =-1.3910E-02*0.5*RHO*L**2
       ZDS   =(-5.6030E-03*0.5*RHO*L**2)
       ZDB   =(-5.6030E-03*0.5*RHO*L**2)
       MQDOT=-8.8000E-04*0.5*RHO*L**5
       MWDOT=-5.6100E-04*0.5*RHO*L**4
       MQ    =-3.7020E-03*0.5*RHO*L**4
       MW    = 1.0324E-02*0.5*RHO*L**3
```

```
      MDS   =(-2.4090E-03*0.5*RHO*L**3)
      MDB   =( 2.4090E-03*0.5*RHO*L**3)
C
      ZG     =0.4D0
      ZB     =0.0D0
      XG     =0.0D0
      XB     =0.0D0
      ZGB    =ZG-ZB
      ZD=ZDS+ALPHA*ZDB
      MD=MDS+ALPHA*MDB
C
C     SET INITIAL CONDITIONS
C
      THETA=0.0D0
C     READ(30,*) THETA
      W      =0.1D0
      Q      =0.0D0
      Z      =0.0D0
      Z      =Z*L
C
C     DETERMINE [A] AND [B] COEFFICIENTS
C
      DV     =(MASS-ZWDOT)*(IY-MQDOT)
     &         -(MASS*XG+ZQDOT)*(MASS*XG+MWDOT)
      A11DV=(IY-MQDOT)*ZW+(MASS*XG+ZQDOT)*MW
      A12DV=(IY-MQDOT)*(MASS+ZQ)+(MASS*XG+ZQDOT)*(MQ-MASS*XG)
      A13DV=-(MASS*XG+ZQDOT)*WEIGHT
      B1DV  =(IY-MQDOT)*ZD+(MASS*XG+ZQDOT)*MD
      A21DV=(MASS-ZWDOT)*MW+(MASS*XG+MWDOT)*ZW
      A22DV=(MASS-ZWDOT)*(MQ-MASS*XG)
     &         +(MASS*XG+MWDOT)*(MASS+ZQ)
      A23DV=-(MASS-ZWDOT)*WEIGHT
      B2DV  =(MASS-ZWDOT)*MD+(MASS*XG+MWDOT)*ZD
C
      A11=A11DV/DV
      A12=A12DV/DV
      A13=A13DV/DV
      A21=A21DV/DV
      A22=A22DV/DV
      A23=A23DV/DV
      B1  =B1DV/DV
      B2  =B2DV/DV
C
C     CALCULATE THE CONTROL GAINS
C
      POLE=1.0/TC
      ALPHA3=4.0*POLE
      ALPHA2=6.0*POLE**2
      ALPHA1=4.0*POLE**3
      ALPHA0=    POLE**4
```

```
      K4=ALPHA0/((B1*A21-B2*A11)*U0**4
     &           +(B1*A23-B2*A13)*ZGB*U0**2)
      A2M=B1*U0**2
      A3M=B2*U0**2
      A0M=-(A11+A22)*U0-ALPHA3
      B1M=B2*U0**2
      B2M=(B2*A12-B1*A22)*U0**3
      B3M=(B1*A21-B2*A11)*U0**3
      B0M=(A11*A22-A21*A12)*U0**2-A23*ZGB-ALPHA2-B1*U0*U0*K4
      C1M=(B2*A11-B1*A21)*U0**3
      C2M=(B1*A23-B2*A13)*ZGB*U0**2

      C0M=(A13*A21-A23*A11)*ZGB*U0+ALPHA1
     &    -(B2+B1*A22-B2*A12)*K4*U0**3
      K2=C1M*B0M*A3M-B1M*C0M*A3M-C1M*B3M*A0M
      K2=K2/(C1M*B2M*A3M-B1M*C2M*A3M-C1M*B3M*A2M)
      K1=(C0M-C2M*K2)/C1M
      K3=(A0M-A2M*K2)/A3M
C
C      WRITE(*,*) K1,K2,K3,K4
C
C     DETERMINE THE EIGENVALUES
C
      A(1,1)=0.0D0
      A(1,2)=0.0D0
      A(1,3)=1.0D0
      A(1,4)=0.0D0
      A(2,1)=A13*ZGB+B1*U*U*K1
      A(2,2)=A11*U  +B1*U*U*K2
      A(2,3)=A12*U  +B1*U*U*K3
      A(2,4)=        B1*U*U*K4
      A(3,1)=A23*ZGB+B2*U*U*K1
      A(3,2)=A21*U  +B2*U*U*K2
      A(3,3)=A22*U  +B2*U*U*K3
      A(3,4)=        B2*U*U*K4
      A(4,1)=-U
      A(4,2)=1.0D0
      A(4,3)=0.0D0
      A(4,4)=0.0D0
C
      CALL DEVLRG(NA,A,LDA,EVAL)
C
      CALL DWRCRN ('EVAL',1,NA,EVAL,1,0)
C
C     DEFINE THE LENGTH X AND BREADTH B TERMS FOR THE
C     INTEGRATION
C
      XL( 1)=  0.0000D0
      XL( 2)=  0.1000D0
      XL( 3)=  0.2000D0
      XL( 4)=  0.3000D0
```

```
      XL( 5)=  0.4000D0
      XL( 6)=  0.5000D0
      XL( 7)=  0.6000D0
      XL( 8)=  0.7000D0
      XL( 9)=  1.0000D0
      XL(10)=  2.0000D0
      XL(11)=  3.0000D0
      XL(12)=  4.0000D0
      XL(13)=  7.7143D0
      XL(14)= 10.0000D0
      XL(15)= 15.1429D0
      XL(16)= 16.0000D0
      XL(17)= 17.0000D0
      XL(18)= 18.0000D0
      XL(19)= 19.0000D0
      XL(20)= 20.0000D0
      XL(21)= 20.1000D0
      XL(22)= 20.2000D0
      XL(23)= 20.3000D0
      XL(24)= 20.4000D0
      XL(25)= 20.4167D0

      DO 102 N = 1,25
            XL(N) = (L/20)*XL(N) - L/2
 102  CONTINUE

C
      BR( 1)= 0.000D0
      BR( 2)= 0.485D0
      BR( 3)= 0.658D0
      BR( 4)= 0.778D0
      BR( 5)= 0.871D0
      BR( 6)= 0.945D0
      BR( 7)= 1.010D0
      BR( 8)= 1.060D0
      BR( 9)= 1.180D0
      BR(10)= 1.410D0
      BR(11)= 1.570D0
      BR(12)= 1.660D0
      BR(13)= 1.670D0
      BR(14)= 1.670D0
      BR(15)= 1.670D0
      BR(16)= 1.630D0
      BR(17)= 1.370D0
      BR(18)= 0.919D0
      BR(19)= 0.448D0
      BR(20)= 0.195D0
      BR(21)= 0.188D0
      BR(22)= 0.168D0
      BR(23)= 0.132D0
      BR(24)= 0.053D0
```

```
      BR(25)= 0.000D0
C
      PISIM =TSIM/DELT
      ISIM  =PISIM
C
C     SIMULATION BEGINS
C
      DO 100 I=1,ISIM
C
C       CALCULATE THE DRAG FORCE, INTEGRATE THE DRAG OVER THE
C       VEHICLE
C
        DO 101 K=1,25
          UCF=W-XL(K)*Q
          VEC1(K)=BR(K)*UCF*DABS(UCF)
          VEC2(K)=BR(K)*UCF*DABS(UCF)*XL(K)
  101   CONTINUE
        CALL TRAP(25,VEC1,XL,HEAVE)
        CALL TRAP(25,VEC2,XL,PITCH)
        HEAVE=0.5*RHO*CDZ*HEAVE
        PITCH=0.5*RHO*CDZ*PITCH
C
C       COUPLING EQUATIONS
C
        DWDV=(IY-MQDOT)*HEAVE+(MASS*XG+ZQDOT)*PITCH
        DQDV=(MASS-ZWDOT)*PITCH+(MASS*XG+MWDOT)*HEAVE
        C1DV=(IY-MQDOT)*MASS*ZG*Q**2
     &       -(MASS*XG+ZQDOT)*MASS*ZG*W*Q
        C2DV=-(MASS-ZWDOT)*MASS*ZG*W*Q
     &       +(MASS*XG+MWDOT)*MASS*ZG*Q**2
C
        DW=DWDV/DV
        DQ=DQDV/DV
        C1=C1DV/DV
        C2=C2DV/DV
C
C       LIMITING CONDITION
C
        IF (ABS(THETA) .GT. PI/4) THEN
           WRITE(*,*) 'BUBBLE EXCEEDED +/- 45, BOAT WAS LOST'
           STOP
        ENDIF
C
C       CONTROL EQUATIONS
C
        DELTA = (K1*THETA+K2*W+K3*Q+K4*Z)
        DELTA = DSAT*DTANH(DELTA/DSAT)
C       IF (DELTA .GT. 0.4) DELTA=0.4
C       IF (DELTA .LT. -0.4) DELTA=-0.4
C
        THEDOT=Q
```

```
      WDOT  =A11*U*W+A12*U*Q+A13*ZGB*DSIN(THETA)
     &        +B1*U*U*DELTA+DW+C1
      QDOT  =A21*U*W+A22*U*Q+A23*ZGB*DSIN(THETA)
     &        +B2*U*U*DELTA+DQ+C2
      ZDOT  =-U*DSIN(THETA)+W*DCOS(THETA)
      IF (I.GT.3) GO TO 150
C
C     INITIAL FIRST ORDER INTEGRATION
C
      THETA = THETA + DELT*THEDOT
      W     = W     + DELT*WDOT
      Q     = Q     + DELT*QDOT
      Z     = Z     + DELT*ZDOT
C
      F1(I)=THEDOT
      F2(I)=WDOT
      F3(I)=QDOT
      F4(I)=ZDOT
C
C     ADAMS-BASHFORTH INTEGRATION
C
  150 F1(4)=Q
      F2(4)=A11*U*W+A12*U*Q+A13*ZGB*DSIN(THETA)
     &       +B1*U*U*DELTA+DW+C1
      F3(4)=A21*U*W+A22*U*Q+A23*ZGB*DSIN(THETA)
     &       +B2*U*U*DELTA+DQ+C2
      F4(4)=-U*DSIN(THETA)+W*DCOS(THETA)
C
      THETA=THETA+(DELT/24.0)*(55.0*F1(4)
     1        -59.0*F1(3)+37.0*F1(2)-9.0*F1(1))
      W    =W+(DELT/24.0)*(55.0*F2(4)
     1       -59.0*F2(3)+37.0*F2(2)-9.0*F2(1))
      Q    =Q+(DELT/24.0)*(55.0*F3(4)
     1       -59.0*F3(3)+37.0*F3(2)-9.0*F3(1))
      Z    =Z+(DELT/24.0)*(55.0*F4(4)
     1       -59.0*F4(3)+37.0*F4(2)-9.0*F4(1))
C
      F1(1)=F1(2)
      F1(2)=F1(3)
      F1(3)=F1(4)
      F2(1)=F2(2)
      F2(2)=F2(3)
      F2(3)=F2(4)
      F3(1)=F3(2)
      F3(2)=F3(3)
      F3(3)=F3(4)
      F4(1)=F4(2)
      F4(2)=F4(3)
      F4(3)=F4(4)
C
      TIME=I*DELT
```

```
            J=J+1
            IF (J.NE.IPRNT) GO TO 100
            ALFA=W/U
            ALFA=(-DATAN(ALFA))*180/PI
            WRITE (20,991) TIME*U0/L,Z/L,THETA*180/PI,
     &                     DELTA,W,Q,ALFA
            WRITE (*,991) TIME*U0/L,Z/L,THETA*180/PI,
     &                    DELTA,W,Q,ALFA
            J=0
C
  100 CONTINUE
      STOP
  991 FORMAT (2X,F8.2,6(2X,F8.4))
      END
C
      SUBROUTINE TRAP(N,A,B,OUT)
C
C     NUMERICAL INTEGRATION ROUTINE USING THE TRAPEZOIDAL RULE
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(1),B(1)
      N1=N-1
      OUT=0.0
      DO 1 I=1,N1
        OUT1=0.5*(A(I)+A(I+1))*(B(I+1)-B(I))
        OUT =OUT+OUT1
    1 CONTINUE
      RETURN
      END
```

APPENDIX D - ROOT LOCUS PROGRAM

```
% THIS MATLAB PROGRAM FINDS THE ROOT LOCUS FOR A GIVEN U0, Zg,
% Alpha, AND Tc.
W=1556.2363;B1=1556.2363;
Iy=561.32;
g=32.2;
m=W/g;
rho=1.94;
L=13.9792;
xg=0;zg=.4;
zgb=.4;
nd1=.5*rho*L^2;
nd2=.5*rho*L^3;
nd3=.5*rho*L^4;
nd4=.5*rho*L^5;

e = zeros(100,4);
X = zeros(4,1);
Y = zeros(4,1);
Z = zeros(4,1);

Alpha=0;

Zqdnd=-6.33e-4;Zwdnd=-1.4529e-2;Zqnd=7.545e-3;Zwnd=-1.391e-2;
Zds=.5*(-5.603e-3);Zdb=-.5*5.603e-3;Zdltnd=(Zds+Alpha*Zdb);
Mqdnd=-8.8e-4;Mwdnd=-5.61e-4;Mqnd=-3.702e-3;Mwnd=1.0324e-2;
Mds=.5*(-0.002409);Mdb=.5*0.002409;Mdltnd=(Mds+Alpha*Mdb);

Zqd=nd3*Zqdnd;Zwd=nd2*Zwdnd;Zq=nd2*Zqnd;Zw=nd1*Zwnd;
Zdlt=nd1*Zdltnd;

Mqd=nd4*Mqdnd;Mwd=nd3*Mwdnd;Mq=nd3*Mqnd;Mw=nd2*Mwnd;
Mdlt=nd2*Mdltnd;

Dv=(m-Zwd)*(Iy-Mqd)-(m*xg+Zqd)*(m*xg+Mwd);
a11Dv=(Iy-Mqd)*Zw+(m*xg+Zqd)*Mw;
a12Dv=(Iy-Mqd)*(m+Zq)+(m*xg+Zqd)*(Mq-m*xg);
a13Dv=-(m*xg+Zqd)*W;
b1Dv=(Iy-Mqd)*Zdlt+(m*xg+Zqd)*Mdlt;
a21Dv=(m-Zwd)*Mw+(m*xg+Mwd)*Zw;
a22Dv=(m-Zwd)*(Mq-m*xg)+(m*xg+Mwd)*(m+Zq);
a23Dv=-(m-Zwd)*W;
b2Dv=(m-Zwd)*Mdlt+(m*xg+Mwd)*Zdlt;

a11=a11Dv/Dv;a12=a12Dv/Dv;a13=a13Dv/Dv;
a21=a21Dv/Dv;a22=a22Dv/Dv;a23=a23Dv/Dv;
b1=b1Dv/Dv;b2=b2Dv/Dv;
```

```
u = 9;
for j = 1:240
U = j/20;
s1(j)= U;
Fn=sqrt(U^2/(zgb*g));
tconst=4.7512;
pole=-1/(tconst*L/u);
p1=[pole pole-0.01 pole-0.02 pole-0.03];
a=[0 0 1 0;a13*zgb a11*u a12*u 0;a23*zgb a21*u a22*u 0;...
  -u 1 0 0];
aU=[0 0 1 0;a13*zgb a11*U a12*U 0;a23*zgb a21*U a22*U 0;-U
1 0 0];


b=[0;b1*u^2;b2*u^2;0];
bU=[0;b1*U^2;b2*U^2;0];

K1=place(a,b,p1);

Y = eig(aU-bU*K1);
e(j,:) = Y';

end

for k = 1:200

U = 8.9 + k/1000;
s2(k) = U;
Fn=sqrt(U^2/(zgb*g));

tconst=4.7512;

pole=-1/(tconst*L/u);

p1=[pole pole-0.01 pole-0.02 pole-0.03];

a=[0 0 1 0;a13*zgb a11*u a12*u 0;a23*zgb a21*u a22*u 0;...

  -u 1 0 0];

aU=[0 0 1 0;a13*zgb a11*U a12*U 0;...
    a23*zgb a21*U a22*U 0;-U 1 0 0];


b=[0;b1*u^2;b2*u^2;0];

bU=[0;b1*U^2;b2*U^2;0];

K1=place(a,b,p1);

Z = eig(aU-bU*K1);
```

```
    f(k,:) = Z';

    end


axis([-.4 .1 -.8 .8])
plot(real(e(1,1)),imag(e(1,1)),'x',...
     real(e(1,2)),imag(e(1,2)),'x',...
     real(e(1,3)),imag(e(1,3)),'x',...
     real(e(1,4)),imag(e(1,4)),'x')
hold
plot(real(e(:,1)),imag(e(:,1)),'.',...
     real(e(:,2)),imag(e(:,2)),'.',...
     real(e(:,3)),imag(e(:,3)),'.',...
     real(e(:,4)),imag(e(:,4)),'.')
plot(real(f(:,1)),imag(f(:,1)),'.',...
     real(f(:,2)),imag(f(:,2)),'.',...
     real(f(:,3)),imag(f(:,3)),'.',...
     real(f(:,4)),imag(f(:,4)),'.')
grid,title('EIGENVALUES FOR SPEEDS 1 TO 100 FPS')
%title('DETAILED LOW SPEED ROOT LOCUS PLOT')
pause
hold
```

LIST OF REFERENCES

1. Cristi, R., Papoulias, F.A., and Healey, A.J., (1991) Adaptive sliding mode control of autonomous underwater vehicles in the dive plane. *IEEE Journal of Oceanic Engineering*, **15**, 3.

2. Fossen, T.I., (1987) Nonlinear modelling and control of underwater vehicles. Doctoral Thesis, Division of Engineering, Cybernetics, Department of Electrical Engineering, Norwegian Institute of Technology.

3. Gertler, M., and Hagen, G.R., (1967) Standard equations of motion for submarine simulation. David Taylor Research Center, Report 2510.

4. Papoulias, F.A., (1992) Stability and bifurcations of towed underwater vehicles in the dive plane. *Journal of, Ship Research*, **36**, 3.

5. Papoulias, F.A., (1993) Dynamics and bifurcations of pursuit guidance for vehicle path keeping in the dive plane. *Journal of Ship Research*, **37**, 2.

6. Oral, Z.O., (June 1993) Hopf bifurcations and nonlinear studies of gain margins in path control of marine vehicles. Masters Thesis, Department of Mechanical Engineering, Naval Postgraduate School.

7. Riedel, J.S., (June 1993) Pitchfork bifurcations and dive plane reversal of submarines at low speeds. Mechanical Engineer's Thesis, Department of Mechanical Engineering, Naval Postgraduate School.

8. Roddy, R.F. (1990) Investigation of the stability and control characteristics of several configurations of the DARPA SUBOFF Model (DTRC Model 5470) from captive-model experiments. David Taylor Research Center, Report DTRC/SHD-1298-08.

9. Stewart, H.B., and Thompson, J.M.T., (1986) Towards a classitication of generic bifurcations in a dissipative dynamical systems. *Dynamics and Stability in Systems*, **1**, 1.

10. Stewart, H.B., and Thompson, J.M.T., (1986) Nonlinear dynamics and chaos, geometrical methods for engineers and scientists. John Wiley and Sons, New York.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria VA 22304-6145	2
2.	Library, Code 052 Naval Postgraduate School Monterey CA 93943-5002	2
3.	Chairman, Code ME Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Professor F.A. Papoulias, Code ME/Pa Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5000	5
5.	LCDR C.A. Bateman, USN 1324 Bolling Ave. Norfolk, VA 23508	2
6.	Naval Engineering Curricular Office, Code 34 Naval Postgraduate School Monterey, CA 93943-5000	1