



**Calhoun: The NPS Institutional Archive**

**DSpace Repository**

Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

2002-12

# Obstacle avoidance control for the REMUS autonomous underwater vehicle

Fodrea, Lynn

Monterey, Calif. Naval Postgraduate School

http://hdl.handle.net/10945/3959

*Downloaded from NPS Archive: Calhoun*


http://www.nps.edu/library

Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

# NAVAL POSTGRADUATE SCHOOL
# Monterey, California


# THESIS

**OBSTACLE AVOIDANCE CONTROL FOR THE REMUS AUTONOMOUS UNDERWATER VEHICLE**

by

Lynn Renee Fodrea

December 2002

Thesis Advisor: Anthony Healey

**Approved for public release; distribution is unlimited**

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 | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | December 2002 | Master's Thesis |

**4. TITLE AND SUBTITLE** Obstacle Avoidance Control for the REMUS Autonomous Underwater Vehicle

**5. FUNDING NUMBERS** N0001401AF00002

**6. AUTHOR (S)** Lynn Fodrea

**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)**
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217-5660

**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 U.S. Department of Defense or the U.S. Government.

**12a. DISTRIBUTION / AVAILABILITY STATEMENT**
Approved for public release; Distribution Unlimited.

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** ***(maximum 200 words)***

Future Naval operations necessitate the incorporation of autonomous underwater vehicles into a collaborative network. In future complex missions, a forward look capability will be required to map and avoid obstacles such as sunken ships. This thesis examines obstacle avoidance behaviors using a forward-looking sonar for the autonomous underwater vehicle REMUS. Hydrodynamic coefficients are used to develop steering equations that model REMUS through a track of specified points similar to a real-world mission track. Control of REMUS is accomplished using line of sight and state feedback controllers. A two-dimensional forward-looking sonar model with a 120° horizontal scan and a 110 meter radial range is modeled for obstacle detection. Sonar mappings from geographic range-bearing coordinates are developed for implementation in MATLAB simulations. The product of bearing and range weighting functions form the gain factor for a dynamic obstacle avoidance behavior. The overall vehicle heading error incorporates this obstacle avoidance term to develop a path around detected objects. REMUS is a highly responsive vehicle in the model and is capable of avoiding multiple objects in proximity along its track path.

| 14. SUBJECT TERMS | 15. NUMBER OF PAGES |
|---|---|
| Obstacle avoidance, REMUS, Underwater vehicle, AUV | 79 |
| | 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

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

# OBSTACLE AVOIDANCE CONTROL FOR THE REMUS AUTONOMOUS UNDERWATER VEHICLE

Lynn Fodrea
Lieutenant, United States Navy
B.S., U.S. Naval Academy, 1998

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN MECHANICAL ENGINEERING**

from the

**NAVAL POSTGRADUATE SCHOOL**
**December 2002**

Author: Lynn Fodrea

Approved by: Anthony J. Healey
Thesis Advisor

Young W. Kwon
Chairman, Department of Mechanical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

# ABSTRACT

Future Naval operations necessitate the incorporation of autonomous underwater vehicles into a collaborative network. In future complex missions, a forward look capability will be required to map and avoid obstacles such as sunken ships. This thesis examines obstacle avoidance behaviors using a forward-looking sonar for the autonomous underwater vehicle REMUS. Hydrodynamic coefficients are used to develop steering equations that model REMUS through a track of specified points similar to a real-world mission track. Control of REMUS is accomplished using line of sight and state feedback controllers. A two-dimensional forward-looking sonar model with a 120° horizontal scan and a 110 meter radial range is modeled for obstacle detection. Sonar mappings from geographic range-bearing coordinates are developed for implementation in MATLAB simulations. The product of bearing and range weighting functions form the gain factor for a dynamic obstacle avoidance behavior. The overall vehicle heading error incorporates this obstacle avoidance term to develop a path around detected objects. REMUS is a highly responsive vehicle in the model and is capable of avoiding multiple objects in proximity along its track path.

THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

## ACKNOWLEDGEMENTS

I would like to thank my thesis advisor, Professor Anthony J. Healey, for his expert insight, direction, and assistance during the development of this work. His ability to see beyond my thoughts and assumptions allowed this product to become what it is.

I would also like to thank CDR Bill Marr, an AUV team member, friend, and counselor who has continued to be there to listen to and advise me on career, academic, and social concerns.

Finally, I would like to thank my fiancé, Major Todd Woodrick, USAF, for listening to me talk through my thoughts, for being there to challenge me, and for being my truest fan and support over the last year and a half.

THIS PAGE INTENTIONALLY LEFT BLANK

# I. INTRODUCTION

## A. BACKGROUND

United States naval warfare strategy is constantly evolving and adapting to our ever-changing world. One of the most foreign and complex areas of naval warfare that requires a myriad of resources to explore and classify is that of the underwater world. With increased Amphibious Operations in the littoral environment and an increased need for Force Protection of our nation's ports, it is critical to be able to characterize the undersea battlefield and an enemy's coastal defenses. Recently, the undersea battlefield has undergone considerable change with the advent of improved mines, submarine quieting, and other littoral threats.

It has often been said that the best way to combat threats in a specific environment is to use assets in the same medium. A major area of development for combating this complex undersea battlefield from the surf zone to the shallow water regime is the Unmanned Underwater Vehicle (UUV). UUVs not only increase safety to our military forces by removing the human swimmer from the hostile minefield environment, but they also provide a more maneuverable asset in the random and turbulent waters of the littorals. The UUV Mission Priorities, as outlined in the Organic Off-board Mine Reconnaissance CONOPS, include programs that will extend knowledge and control of the undersea battle space through the employment of covert sensors capable of operating *reliably* in high-risk areas. The CONOPS states that there are four basic mission areas

for which the utility of unmanned undersea systems was substantiated: mine warfare, surveillance, intelligence collection, and tactical oceanography. To ensure success and reliability during these missions, it is imperative that the UUVs used are capable of obstacle avoidance. This thesis will focus on obstacle avoidance arguments for a specific type of UUV known as the Autonomous Underwater Vehicle (AUV). AUVs are unmanned, independent craft with respect to power and control and require no external interface. AUVs appeal to the underwater community in that they are able to:

- Provide their own power
- Provide data storage capabilities
- Make decisions based on inputs from onboard sensors

These capabilities alone set them apart from their well-known counterparts, ROVs or Remotely Operated Vehicles. ROVs are not only tethered, but require a human interface as well as sufficient cable to search the waters around the base platform (Ruiz, 2001)

## B. MOTIVATION

Advancements have been made in the area of robotics for underwater environments over the past several years. AUV development began as far back as 1960 with experimental prototypes available in the 1980's. For a history on AUV development, see (Blidberg, 2001). AUVs possess the unique ability to safely operate in littoral areas for search, detection, and classification of mines and for hydrographic reconnaissance and intelligence. To broaden the

capabilities of underwater vehicles for military, industrial and environmental applications in multiple vehicle operations, it is essential to design a robust robotic system that exhibits the maximum degree of autonomy, both through navigation and sensory processing. One of the greatest technological challenges facing AUVs and the robot community today is that of navigation around obstacles. While most underwater vehicles can solve the problem of localization and maneuvering, many do not possess the capability to move around obstacles that arise in their programmed path, specifically in unmapped areas near the littorals where mine-like objects or other potential hazards are prevalent. Land robots and crawling vehicles are capable of obstacle and collision avoidance using a "stop-back-turn" principle that swimming vehicles cannot (Healey, Kim, 1999). This thesis will present a solution to the obstacle avoidance problem for the Remote Environmental Measuring Unit System (REMUS) AUV.

## C. OBSTACLE AVOIDANCE FOR AUTONOMOUS UNDERWATER VEHICLES

The obstacle avoidance problem has been under research since the advent of underwater vehicle technology. Several approaches have been used to solve this problem for underwater robots. One approach is that of wall-following or obstacle contour following (Kamon, 1997). This method utilizes the obstacle boundaries to determine a close proximity path around the obstacle until reaching a position on the obstacle boundary where it can break away and return to its course. The boundary following continues until the obstacle no longer blocks the desired path.

Experimental results using Kamon's wall following algorithm show that this technique produces minimal path distances around obstacles.

The approach proposed by Moitie and Suebe [Moite & Suebe, 2000] uses an obstacle avoidance system consisting of four subsystems: a digital terrain manager used to estimate the sea floor altitude, a global planner used to generate waypoints to guide the AUV to a given target, a reflex planner to check the trajectories of the global planner, and an obstacle avoidance sonar for environmental mapping. All of these subsystems are used to determine a viable area of the state space from which a viable (or escape) trajectory can be used.

The Vector Field Histogram (VHF) technique (Borenstien and Koren, 1991) consists of a two-stage data reduction process that uses a two-dimensional Cartesian histogram grid as a world model. The first stage is data reduction to a one–dimensional local polar histogram with each sector representing an obstacle density. The second stage involves a selection of the sector with the lowest obstacle density. The steering model is then reduced to calculating an avoidance-heading vector aligned with the selected sector.

## D. PATH PLANNING

Path planning is a tool used for devising collision free trajectories for robot vehicles in a structured world where mission specifications and environmental models are known. Path planning commonly occurs prior to mission execution for the existing environmental constraints.

Environmental data allows path planners to design paths around known physical obstacles such as trees and pillars or hazardous environments such as rough terrain or high turbulence areas. Path planning differs from obstacle avoidance in that obstacle avoidance is performed in a non-structured world that is initially assumed to be free of obstructions. However, due to the unpredictable nature of an underwater environment, path planning alone is insufficient to allow for safe vehicle navigation. Obstacle avoidance is a necessary tool for in situ response to unknown environmental conditions and hazards.

Several path planning techniques have been developed for both land based and subsurface robots. One that has received the most attention in recent years is the potential field approach in which an artificial potential field is defined to reflect the structure of the space around the vehicle (Thrope, 1985, Krogh, 1986). A repulsive field pushes the vehicle away form an indicated obstacle while an attractive field pulls a vehicle toward a goal. The path to the goal is minimized through the space. It is configured to have a global minimum at the desire terminal state of the vehicle. The main drawback to this approach lies in the fact that local minima may entrap the robot trajectory.

A second approach considered by Latombe (1991) is that of cell decomposition in which the workspace is divided into non-overlapping cells represented by nodes. The space is then searched from starting point to the end node using a graph search algorithm to determine the path of free cells.

Further progress has been made to incorporate path planning and obstacle avoidance in a more dynamic program. Stentz (1994) develops a path planning algorithm known as D* for partially known environments in which a sensor is also available to supplement a map of the environment. It combines what is known of the global environment prior to mission with acquired local environmental data during missions. The D* technique uses a cost based approach in which a directed graph of arcs is generated prior to mission with each arc having an associated cost. The robot's sensor can then measure arc costs in its local vicinity and generate known and estimated arc values that compromise a map.

Lane (2001) uses an approach known as dynamic programming. This method considers a modular system that handles different needs of the environment while the robot is in motion. These modules consist of a segmentation module that identifies regions of the sonar image containing obstacles, a feature extraction module, a tracking module that provides a dynamic model of the obstacle, a workspace representation that builds a symbolic representation of the vehicle's surroundings, and finally a path planning module that represents each obstacle as a constraint. The maneuvering solution is then based on minimizing the path length to the goal.

While several of the path planning techniques described above are designed for land robots vice underwater robots and involve much simpler dynamic motions, the challenge of underwater robot technology is in the

difficulty of ceasing or changing a forward motion given a short notice sonar return.

## E. SCOPE OF THIS THESIS – THE REMUS VEHICLE

The REMUS vehicle was developed at Wood's Hole Oceanographic Institute (WHOI) in the Oceanographic Systems Laboratory. It is designed to perform hydrographic reconnaissance in the Very Shallow Water (VSW) zone from 40 to 100 feet. As seen in Figure 1, it is 62 inches long and 7.5 inches in diameter. It weighs 80 pounds in air and can operate in depths up to 328 feet, but typically operates between 10 and 66 feet. The aft end propeller enables REMUS to reach a maximum speed is 5.6 knots. Its four fins, two horizontal and two vertical on either side and just forward of the propeller, allow pitch and yaw motions for maneuvering. Table 1 includes the remaining functional and physical characteristics.

Figure 1. REMUS VEHICLE

Currently, REMUS is equipped with a number of sensors that can generate hydrographic maps, maps of water currents, water clarity, temperature, and salinity profiles, as well as some acoustic profiles. While REMUS is fitted with two side-scan sonars that are used to detect objects on or near the sea floor, a forward-looking sonar

would give it the ability to detect objects in front of the vehicle.

Table 1. REMUS Functional and Physical Characteristics

| PHYSICAL/FUNCTIONAL AREA | CHARACTERISTIC |
|---|---|
| Vehicle Diameter | 7.5 in |
| Vehicle Length | 62 in |
| Weight in Air | 80 lbs |
| External Ballast Weight | 2.2 lbs |
| Operating Depth Range | 10 ft to 66 ft |
| Transit Depth Limits | 328 ft |
| Typical Search Area | 875 yds X 1093 yds |
| Typical Transponder Range | 1640 yds |
| Operational Temperature Range | +32F to +100F |
| Speed Range | 0.5 knots to 5.6 knots |
| Maximum Operating Water Current | 2 knots |
| Maximum Operating Sea State | Sea State 2 |
| Battery | 1 kW-hr internally rechargeable Lithium-ion |
| Endurance | 20 hours at 3 knots; 9 hours at 5 knots |

## F. THESIS STRUCTURE

The intent of this research is to develop a forward-looking sonar model that supports obstacle avoidance behaviors on the REMUS vehicle. This is a two step process accomplished through the following: firstly, develop a robust steering model for the REMUS vehicle as a necessary building block for obstacle avoidance behaviors; secondly, build obstacle avoidance control into the steering model to enable safe navigation of the very shallow water environment while gathering or verifying environmental and minefield data. To this end, the steering model designed for the REMUS vehicle is based on known hydrodynamic

coefficients and will incorporate an obstacle avoidance heading command for both single and multiple threat environments.

Chapter II will focus on the development of the equations of motion for the REMUS AUV. Chapter III will describe the steering control laws associated with the EOM for REMUS. Chapter IV will discuss the obstacle avoidance algorithm developed for REMUS. Chapter V will present simulation analysis for the obstacle avoidance behavior discussed in the previous chapter and Chapter VI will offer conclusions and recommendations for future study.

# II. STEERING MODEL

## A. GENERAL

Modeling of rigid body dynamics for underwater vehicles differs from modeling of other robots only in terms of the forces applied to produce motion. The approach taken with and underwater vehicle is that of a moving body in free space without constraint. The propulsion and maneuvering forces on the moving body are hydrodynamic and hydrostatic in origin and are caused by interactions with the ocean water particles local to the body, rather than interactions with the ground as those of land based robots. These forces are often controllable and can thus be studied from a perspective of stabilization. (Healey class notes)

## B. EQUATIONS OF MOTION IN THE HORIZONTAL PLANE

The following paragraphs describe the development of the steering model used to control the REMUS vehicle. This model was adapted from that of the ARIES AUV (Healey & Marco, 2001) and is based on the following assumptions:

- the vehicle behaves as a rigid body
- the earth's rotation is negligible for the purposed of acceleration components of the vehicle center of mass
- the primary forces that act on the vehicle are inertial and gravitational in origin and are derived from hydrostatic, propulsion, thruster, and hydrodynamic lift and drag forces

The equations of motion (EOM) for steering are derived using a Newton-Euler approach that relates the vehicle's

position and motions in the local plane to those in the global plane. The geometry of the global and local coordinate systems can be seen in Figure 3 below.

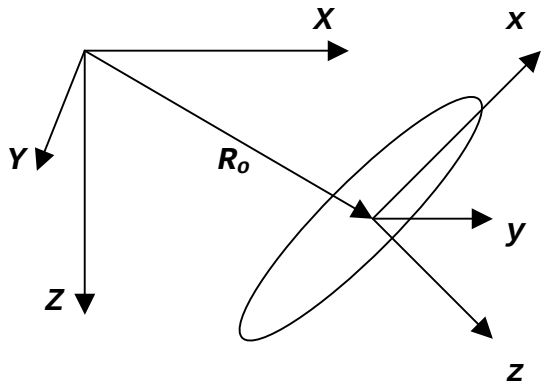



Figure 2. Local and Global Coordinate System (From: Marco and Healey, 2001)

Healey, (1995) shows that the local velocity vector $[u,v,w]^{-1}$ where *u* is forward speed (surge), *v* is side slip (sway) and *w* is any component velocity in the local Z direction (heave), can be easily transformed to the global velocity vector $\left[\dot{X},\dot{Y},\dot{Z}\right]^{-1}$ through the 'Euler' angles $\phi$, $\theta$, and $\psi$ as follows:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = T(\phi,\theta,\psi) \bullet \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (1)$$

Where T is the transformation matrix:

$$T^{-1}(\phi,\theta,\psi) = \begin{bmatrix} \cos\psi\cos\theta, & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi, & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta, & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi, & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta, & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad (2)$$

However, the connection between angular attitude and angular velocity is not as simple. Rate gyros in use today measure the components of inertial angular velocity of a vehicle that lie along the vehicle's body axes. Thus, Healey derives the inertial angular rates in terms of components that have angular velocities about the global axes and then transforms them as above to the final reference frame. The final transformation takes the form:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \sum T(\phi) \bullet T(\theta) \bullet T(\psi) \bullet \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} + T(\phi) \bullet T(\theta) \bullet \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + T(\phi) \bullet \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} \tag{3}$$

in which the rate components from each 'Euler' angle are viewed as follows:

- the change of rotation $\psi$ as a vector quantity lying along the original Z axis
- the rate of change of $\theta$ as a vector quantity lying along the Y axis of the first intermediate frame and
- the rate of change of $\phi$ as a vector lying along the X axis of the final body frame

with the result:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\psi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{4}$$

For small angular rotations, it is evident that:

$$\dot{\phi} = p;\ \dot{\theta} = q;\ \dot{\psi} = r.$$

The final EOM are developed in the body fixed frame coordinates using these inertial frame quantities of position, velocity, and acceleration of the vehicle's center of mass. The translational equation of motion is a vector equation relating the global acceleration of the center of mass to the net sum of all of the forces acting on the vehicle in three degrees of freedom (X,Y,Z) as:

$$F = m\{\dot{v} + \dot{\omega} \times \rho_g + \omega \times \omega \times \rho_g + \omega \times v\} \tag{5}$$

The rotational equation of motion is derived from equating the sum of the applied moments about the vehicle's center of mass to the rate of change of angular momentum of the vehicle about it's center of mass. The mass moment of inertia of the vehicle, I, about its center of gravity changes with loading. Thus, the mass moment of inertia is evaluated about the body-fixed frame that lies along the vehicle's axis of symmetry. The rotational equation of motion in vector form thus becomes:

$$M_o = I_o\dot{\omega} + \omega \times (\mathrm{I}_o\omega) + m\{\rho_g \times \dot{v} + \rho_g \times \omega \times v\} \tag{6}$$

With the addition of weight and buoyancy terms that act at the centers G and B, Healey, (1995) derives the equations of motion for a six degree of freedom model as:

SURGE EQUATION OF MOTION

$$m[\,\dot{u}_r - v_r r + w_r q - x_G\left(q^2 + r^2\right) + y_G\left(pq - \dot{r}\right) + z_G\left(pr + \dot{q}\right)\,] + (W - B)\sin\theta = X_f \tag{7}$$

SWAY EQUATION OF MOTION

$$m[\,\dot{v}_r + u_r r - w_r p + x_G\left(pq + \dot{r}\right) - y_G\left(p^2 + r^2\right) + z_G\left(qr - \dot{p}\right)\,] - (W - B)\cos\theta\sin\phi = Y_f \tag{8}$$

HEAVE EQUATION OF MOTION

$$m[\,\dot{w}_r - u_r q + v_r p + x_G\left(pr - \dot{q}\right) + y_G\left(qr + \dot{p}\right) - z_G\left(p^2 + q^2\right)\,] + (W - B)\cos\theta\cos\phi = Z_f \tag{9}$$

ROLL EQUATION OF MOTION

$$I_x \dot{p} + (I_z - I_y)qr + I_{xy}(pr - \dot{q}) - I_{yz}(q^2 - r^2) - I_{xz}(pq + \dot{r}) + m[y_G(\dot{w} - u_r q + v_r p) \quad (10)$$

$$- z_G(\dot{v}_r + u_r r - w_r p)] - (y_G W - y_B B)\cos\theta\cos\phi + (z_G W - z_B B)\cos\theta\sin\phi = K_f$$

PITCH EQUATION OF MOTION

$$I_y \dot{q} + (I_z - I_z)pr - I_{xy}(qr + \dot{p}) + I_{yz}(pq - \dot{r}) + I_{xz}(p^2 - r^2) - m[x_G(\dot{w} - u_r q + v_r p) \quad (11)$$

$$- z_G(\dot{u}_r - v_r r + w_r q)] + (x_G W - x_B B)\cos\theta\cos\phi + (z_G W - z_B B)\sin\theta = M_f$$

YAW EQUATION OF MOTION

$$I_z \dot{r} + (I_y - I_x)pq - I_{xy}(p^2 - q^2) - I_{yz}(pr + \dot{q}) + I_{xz}(qr - \dot{p}) + m[x_G(\dot{v}_r + u_r r - w_r p) \quad (12)$$

$$- y_G(\dot{u}_r - v_r r + w_r q)] - (x_G W - x_B B)\cos\theta\sin\phi - (y_G W - y_B B)\sin\theta = N_f$$

Where:

W = weight
B = buoyancy
I = mass moment of inertia terms

$u_r$, $v_r$, $w_r$ = component velocities for a body fixed system with respect to the water

$p$, $q$, $r$ = component angular velocities for a body fixed system

$x_B$, $y_B$, $z_B$ = position difference between geometric center and center of buoyancy

$x_G$, $y_G$, $z_G$ = position difference between geometric center and center of gravity

$X_f$, $Y_f$, $Z_f$, $K_F$, $M_f$, $N_f$ = sums of all external forces acting in the particular body fixed direction

Healey (1995) further simplifies Equations 7 thru 12 with the following assumptions:

- The center of mass of the vehicle lies below the origin ($z_G$ is positive)

- $x_G$ and $y_G$ are zero
- The vehicle is symmetric in its inertial properties
- The motions in the vertical are negligible (i.e. $[w_r, p, q, r, Z, \phi, \theta] = 0$)
- $u_r$ equals the forward speed, $U_o$.

The simplified equations of motion are thus:

$$u_r = U_o \tag{13}$$

$$m\dot{v}_r = -mU_o r + \Delta Y_f(t) \tag{14}$$

$$I_{zz}\dot{r} = \Delta N_f(t) \tag{15}$$

$$\dot{\psi} = r \tag{16}$$

$$\dot{X} = U_o \cos\psi - v_r \sin\psi + U_{cx} \tag{17}$$

$$\dot{Y} = U_o \sin\psi - v_r \cos\psi + U_{cy} \tag{18}$$

## C. HYDRODYNAMIC COEFFICIENTS

The modeling of submerged vehicles assumes small forward motions at nominal speeds in a straight line transit. Under steady motion conditions, there is a balance between the hydrodynamic drag and propulsion forces as well as the weight and buoyancy forces. The predominant forces from lift that arise in directions other than the longitudinal direction are caused from small angles of attack and side slip. Hydrodynamic forces are related to relative velocities and accelerations of the fluid and vehicle that result from any motions that deviate from the straight line path assumed above. Due to the symmetry of vehicles about their longitudinal axis, the components of fluid motion in the transverse direction are often

independent of motions in the longitudinal direction. Healey proposes that due to the symmetry of the vehicle, one can heuristically determine that only a subset of motions would affect the loading in any particular direction (Healey class notes) and uses the following expressions to describe the hydrodynamic forces of sway and yaw respectively:

$$\Delta Y_f = f(v_r, dv_r/dt, r, dr/dt, p, dp/dt, t) \tag{19}$$

$$\Delta \mathrm{N}_f = f(p, dp/dt, v_r, dv_r/dt, r, dr/dt, t) \tag{20}$$

It is evident that the sway and yaw motions are coupled in horizontal plane steering. Roll motion coupling is common but is often one way and is thus not considered. The fluid forces above are often linearized using Tayor series expansion terms in individual motion components. These expansion terms are termed ‘hydrodynamic coefficients’ and depend on the shape characteristics of the vehicle. Errors in these coefficients will have a significant affect on the natural stability of the vehicle as they are the building blocks of the dynamics matrix. Through the assumption of ‘small’ motions, the expression for the transverse (sway) force is:

$$Y_f = Y_{\dot{v}_r}\dot{v}_r + Y_{v_r}v_r + Y_{\dot{r}}\dot{r} + Y_r r \tag{21}$$

and for the expression for the rotational (yaw) force is:

$$N_f = N_{\dot{v}_r}\dot{v}_r + N_{v_r}v_r + N_{\dot{r}}\dot{r} + N_r r \tag{22}$$

This leads to:

$$Y_{\dot{v}_r} = \frac{\partial Y_f}{\partial \dot{v}_r};\quad \mathrm{Y}_{\mathrm{v_r}} = \frac{\partial \mathrm{Y_f}}{\partial \mathrm{v_r}};\quad \mathrm{Y}_{\dot{\mathrm{r}}} = \frac{\partial \mathrm{Y_f}}{\partial \dot{\mathrm{r}}};\quad \mathrm{Y_r} = \frac{\partial \mathrm{Y_f}}{\partial \mathrm{r}}; \tag{23-26}$$

and

$$N_{\dot{v}_r} = \frac{\partial N_f}{\partial \dot{v}_r}; \quad N_{v_r} = \frac{\partial N_f}{\partial v_r}; \quad N_{\dot{r}} = \frac{\partial N_f}{\partial \dot{r}}; \quad N_r = \frac{\partial N_f}{\partial r}; \qquad (27\text{-}30)$$

Where:

$Y_{\dot{v}_r}$ = coefficient for added mass in sway
$Y_{\dot{r}}$ = coefficient for added mass in yaw
$Y_{v_r}$ = coefficient of sway force induced by side slip
$Y_r$ = coefficient of sway force induced by yaw
$N_{\dot{v}_r}$ = coefficient for added mass moment of inertia in sway
$N_{\dot{r}}$ = coefficient for added mass moment of inertia in yaw
$N_{v_r}$ = coefficient of sway moment from side slip
$N_r$ = coefficient of sway moment from yaw

The hydrodynamic coefficients for steering for the REMUS vehicle were adapted from thesis work performed by MIT (Prestero, 2001) establishing estimates of all vehicle coefficients. Force contributions from lift, drag and added mass are summed to provide a set of combined force coefficients for both locally linearized and large angle motions. With modification, Table 2 below includes the coefficients of interest to the discussion above. The value for $Y_{v_r}$ was determined by adding the linearized combined coefficients for crossflow drag, $Z_{wc}$, body lift, $Z_{wl}$, and fin lift, $Z_{wf}$. The value for $Y_r$ was similarly determined by adding the linearized combined coefficients for crossflow drag, $Z_{qc}$, added mass, $Z_{qa}$, and fin lift, $Z_{qf}$. The value for $N_{v_r}$ was determined from first principles using Hoerner's (1965) equation for body lift moment

$$M_{uwl} = -N_{uwl} = -\frac{1}{2}\rho d^2 c_{yd\beta} x_{cp} \qquad (31)$$

where the center of pressure, $x_{cp}$, is centered at a point between 0.6 and 0.7 of the total body length from the nose. The moment coefficients for the rudder, $N_d$ and $Y_d$, were scaled from those in Appendix D by 3.5 to account for variation in experimental data. Figure 7-7 of Prestero shows a turn rate of 10 deg/sec with 4 degrees of rudder. This is approximately 3.5 times what the REMUS model predicts (33.69 deg/sec).

Table 2. REMUS Hydrodynamic Coefficients for Steering

| | |
|---|---|
| $Y_{\dot{v}_r}$ | -3.55e01 kg |
| $Y_{\dot{r}}$ | 1.93 kg m/rad |
| $Y_{v_r}$ | -6.66e01 kg/s (Same as Zw) |
| $Y_r$ | 2.2 kg m/s (Same as Zq) |
| $N_{\dot{v}_r}$ | 1.93 kg m |
| $N_{\dot{r}}$ | -4.88 kg $m^2$/rad |
| $N_{v_r}$ | -4.47 kg m/s |
| $N_r$ | -6.87 kg $m^2$/s (Same as Mq) |
| $N_d$ | -3.46e01/3.5 kg $m/s^2$ |
| $Y_d$ | 5.06e01/3.5 kg $m/s^2$ |

Finally, Johnson (2001) determined that rudder action produces forces that when linearized are: $Y_\delta \delta_r(t)$ and $N_\delta \delta_r(t)$. The dynamics of the vehicle are thus defined as:

$$m\dot{v}_r = -mU_o r + Y_{\dot{v}_r}\dot{v}_r + Y_{v_r}v_r + Y_{\dot{r}}\dot{r} + Y_r r + Y_\delta \delta_r(t) \quad (32)$$

$$I_{zz}\dot{r} = N_{\dot{v}_r}\dot{v}_r + N_{v_r}v_r + N_{\dot{r}}\dot{r} + N_r r + N_\delta \delta_r(t) \quad (33)$$

$$\dot{\psi} = r \quad (34)$$

## D. VEHICLE KINEMATICS

The kinematics of the vehicle are described by Equations (32) and (33) where $U_{cx}$ and $U_{cy}$ are the current velocities in the associated direction. These two equations, as well as the simple relation of heading to its derivative, compose the steering dynamics of REMUS in matrix form, $\boldsymbol{M}\dot{\mathrm{x}} = \boldsymbol{A}\mathrm{x} + \boldsymbol{B}\mathrm{u}$, and can be expressed as follows:

$$\begin{bmatrix} m - Y_{\dot{v}_r} & -Y_{\dot{r}} & 0 \\ -N_{\dot{v}_r} & I_{zz} - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v}_r \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} Y_{v_r} & Y_r - mU_o & 0 \\ N_{v_r} & N_r & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v_r \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} Y_\delta \\ N_\delta \\ 0 \end{bmatrix} \delta_r(t) \quad (35)$$

where $\delta_r(t)$ is a generalized command that represents the control input to both rudders.

## E. VEHICLE DYNAMICS

The final assumption made for vehicle dynamics (Johnson, 2001) is that the cross coupling terms in the mass matrix are zero. This is based on the vehicle's symmetry and the rudders being very close to equidistant from the body center. Thus, in matrix form, the final vehicle dynamics are defined as:

$$\begin{bmatrix} m - Y_{\dot{v}_r} & 0 & 0 \\ 0 & I_{zz} - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v}_r \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} Y_{v_r} & Y_r - mU_o & 0 \\ N_{v_r} & N_r & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v_r \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} Y_\delta \\ N_\delta \\ 0 \end{bmatrix} \delta_r(t) \quad (36)$$

# III. CONTROL METHODS AND ARCHITECTURE

## A. GENERAL CONTROL THEORY

Obstacle avoidance maneuvers for robots are complex in that they must be performed as a reaction to a stimulus from a sensor. They become an issue of even greater interest and concern for underwater robots that must execute local reflexive maneuvers, or maneuvers in which the vehicle must process a sonar return, determine if that return is a threat along its proposed path, and further navigate around the threat before regaining its original path. Through sensor measurements, nonlinear path deviations can be developed to avoid these threats, while still scanning the underwater environment for possible mines and other environmental data.

Due to their autonomy, control of AUVs is relatively difficult. However, in spite of the uncertainty of hydrodynamic forces, feedback control has been a suitable solution used to provide commands to actuators that control and stabilize the motion of underwater vehicles (Healey and Marco, 2001). Riedel (1999) asserts that the single most important fact contributing to the difficulty in the control of underwater vehicles is the desire to control them along or about two or more axes. This leads to stronger coupling, larger nonlinearities and more state equations in the equations of motion. Additional factors that contribute to the control problem are as follows:

- A small AUV may be controllable in all six DOF
- Actuator dynamics are much smaller on underwater vehicles
- Power and control for the vehicle is limited by the onboard capacity of the vehicle
- Human intervention for fault processes is not possible

These same factors contribute to the obstacle avoidance problem due to the fact that nonlinear control is necessary during avoidance maneuvers. REMUS has a very high turn rate and is a very responsive vehicle. Thus, REMUS requires more robust control. This type of control can be achieved with both sliding mode theory and through a simple dead reckoning or “follow the rabbit” track guidance technique.

The REMUS steering model uses autopilot controls for maneuvering based on the NPS ARIES state variable time domain model (Marco, Healey, 2001). Autopilot is the name associated with the control systems that stabilize the motion of vehicles. As described by Marco, there are four different autopilots for flight maneuvering control. These consist of independent diving, steering/heading, altitude above bottom, and cross-track error controllers. All four modes are de-coupled for ease of design and are based on sliding mode control (SMC) theory. Sliding mode control is a robust technique, or one that provides high performance through widely varied operating conditions, used for compensation of nonlinear systems as well as for systems whose parameters vary in a predictable way with speed

(Healey, 1992). Sliding mode controls are ideal in that they effectively replace an $n^{th}$ order system with an equivalent $1^{st}$ order system. They are simple to use and easy to implement with minimal tuning, making them ideal for use in control design. Two tuning factors are used in this model to include *Eta_FlightHeading*, $\eta$, and *Phi_FlightHeading*, $\phi$, as seen in Appendix A.

## B. REMUS CONTROL ARCHITECTURE

The key to a robust control model is the use of feedback for specific motion variables as measured by sensors to drive the vehicle's actuators (control planes, rudders or thrusters). The steering controller is the only autopilot controller necessary for modeling addressed in this thesis. It is a second order model that uses $r$ and $\psi$ for feedback, modeling side-slip velocity, $v$, as a disturbance that can be overcome by the robust SMC model. Additionally, simple line-of-sight guidance is used to maintain track path by looking ahead to planned waypoints.

## C. SLIDING MODE CONTROL

Using multivariable sliding mode control methods, an accurate steering controller can be developed. These methods are used with predominantly linear system models as opposed to the SMC methods used for nonlinear systems (Healey, 1992). Revising the EOM for a state variable $\dot{r}(t)$ to the general form

$$\dot{x} = Ax + Bu \tag{37}$$

where $x \in \mathbf{R}^{n*1}; A \in \mathbf{R}^{n*n}; B \in \mathbf{R}^{n*r}; u \in \mathbf{R}^{r*1}$, and $u$ is the rudder angle, a SMC can be designed to drive this state to stable solution, or one in which the sliding surface $\sigma = 0,\ \sigma \in \mathbf{R}^{\rho*1}$. With the sliding surface defined as:

$$\sigma = s'\tilde{x};\ \tilde{x} = x - x_{com} \tag{38}$$

where ***s'*** is a vector of directions in the state error space. The elements of $\sigma$ are the lengths of the projection of the state error vector, $x_{com}$ is a dynamic exogenous variable created as a command signal to track, and $\tilde{x}$ is the state error which is required to be driven to zero so that the command state equals actual state. The values of ***s'*** are found by the requirement that when $\sigma = 0$, the system dynamics must exhibit stable sliding on the surface. Thus, the closed loop dynamics are given by the poles of the closed loop matrix as,

$$(A - bk_2) = A_c,\ with\ k_2 = [s'B]^{-1}s'A \tag{39}$$

where $k_2$ is chosen by pole placement and $A_c s' = 0$ to achieve the condition $\sigma = 0$. The eigenvectors of the $A_c$ matrix determine the linear state feedback gains for each state used to define the sliding surface as follows:

$$\sigma(t) = s_2(r_{\_com} - r(t)) + s_3(\psi_{com} - \psi_{LOS}(t)) \tag{40}$$

The poles selected for the REMUS model SMC solution were moved farther from zero than those in the original ARIES model in order to stabilize the system dynamics. As seen in Appendix A, these poles were placed at [–1.4 -1.45 0.0]. The pole at the origin is necessary to allow for the single sliding constraint for the single input system implied by

$\sigma = 0$. The remaining poles both exhibit stable dynamics as they are in the left half plane. The gains obtained from this pole placement were [k1 k2 k3] = [0.769 –0.6 0.0] for [$v$, $r$, $\psi$] respectively. With the sliding surface defined in equation (39) and the gains determined from pole placement, the commanded rudder in the LOS controller becomes:

$$dr(t) = -k2 \cdot r(t) \cdot \eta \cdot \tanh(\sigma(t)/\phi) \tag{41}$$

where $\eta$ and $\phi$ are tuning factors equal to 0.5 and 0.1 respectively.

## D. LINE OF SIGHT GUIDANCE

This purpose of the Line of Sight (LOS) controller is to reduce the heading error to zero. The REMUS model adapts the original LOS guidance for ARIES [Marco and Healey (2001)] with a follow-the-rabbit technique similar in nature to the transducer based dead-reckoning approach with which REMUS operates. The LOS controller forces the vehicle to head in the direction of the current waypoint by defining the error in the heading, $\tilde{\psi}_{LOS}$, as the difference between the commanded line of sight and the actual heading, or:

$$\tilde{\psi}(t)_{LOS} = \psi(t)_{track} - \psi(t) \tag{42}$$

$$\text{where } \psi(t)_{track} = \arctan(\tilde{Y}(t)_{wpt(i)}, \tilde{X}(t)_{wpt(i)}) \tag{43}$$

The commanded heading is based on the angle between the current position and the next waypoint. The REMUS model simply adds an additional look-ahead point or dead-reckoning point on the track toward the next waypoint

forward of the vehicle position as seen in Figure 3 below. The distance to this point is incorporated into the heading error as follows:

$$\tilde{\psi}(t)_{LOS} = \psi(t)_{track} - \psi(t) - \arctan(cte(t)/rabbit) \qquad (44)$$

where *rabbit* is the look-ahead point and *cte* is the cross track error between the actual vehicle position and the desired track.

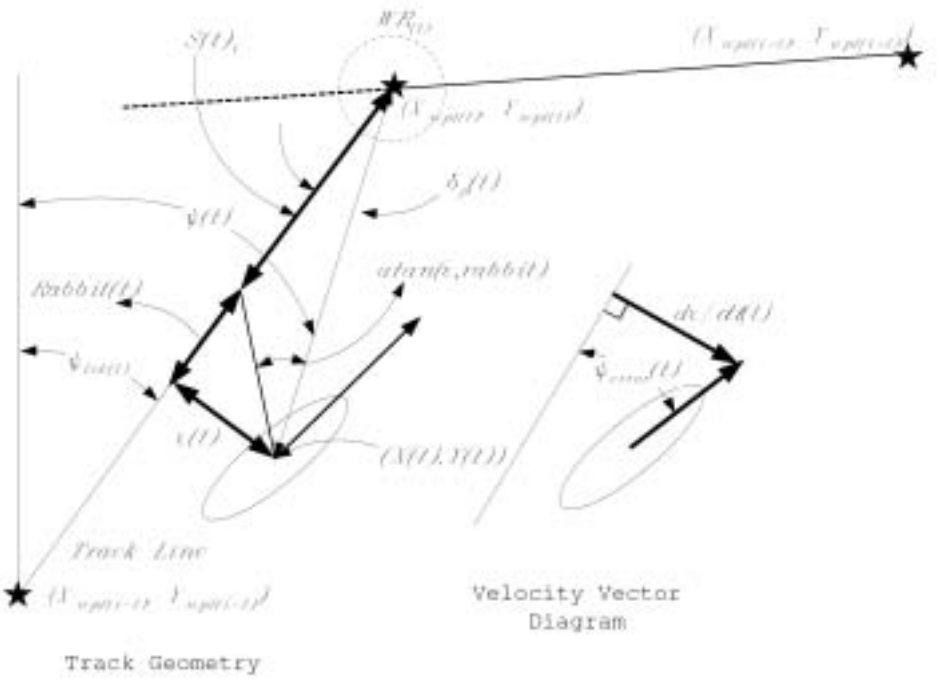



Figure 3. Track Geometry and Velocity Vector Diagram

While LOS guidance controls REMUS along the track from waypoint to waypoint, a different method is used to determine when to turn as the waypoints are approached. The following command is used to ensure that REMUS will begin tracking the next waypoint when approaching the present waypoint:

$$sqrt(X\_Way\_Error(t)^2 + Y\_Way\_Error(t)^2) <= W\_R \left| s(t) < 0.0 \right| ss(t) < rabbit \quad (45)$$

where W_R is the watch radius around the waypoint, *s* is the distance remaining on track, and *ss* is the radial distance to go to the next waypoint. Thus, REMUS will begin to track off the next waypoint if it has entered the watch radius around its present waypoint, if is has passed its present waypoint, or if the rabbit distance is greater than the radial distance to go to the waypoint.

THIS PAGE INTENTIONALLY LEFT BLANK

# IV. OBSTACLE AVOIDANCE MODEL

## A. THE REMUS SEARCH PATH

Path planning for the REMUS vehicle is based on the information to be gathered during a mission. REMUS is used in minefields to search and classify mine-like objects whose location is frequently known. However, it is also widely used to map the very shallow water zone of the littoral region where an accurate map may not exist to provide hydrographic maps with for use by fleet units. The search path used for this vehicle is commonly referred to as the lawnmower technique and is used to cover a square grid area. Depending on search area and target detection analysis performed prior to a mission, this path may vary. This thesis models a REUMUS path that uses rows approximately 200 meters in length with 15-40 meters of separation as seen in Figure 4 below.

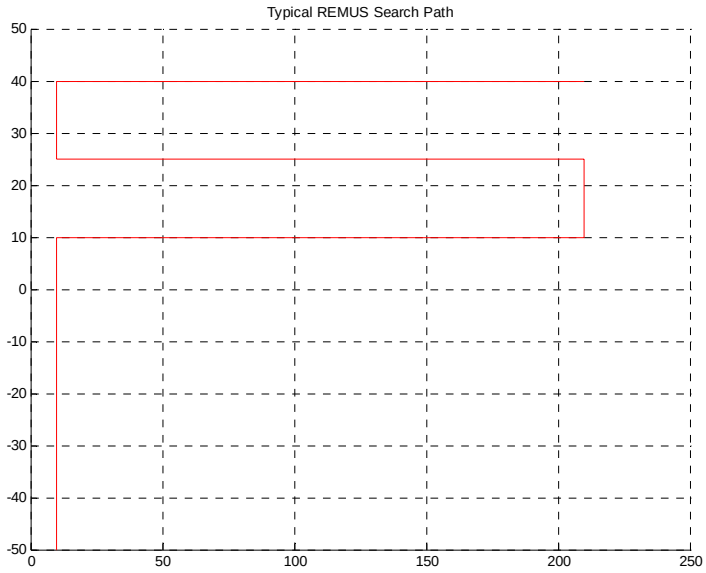



Figure 4. Typical REMUS Search Path

## B. SONAR MODEL

This model uses a two-dimensional forward-looking sonar with a 120° horizontal scan and a 110-meter radial range as seen in Figure 5. This is an estimated range based on a viable 400KHz sonar frequency. As Lane contends, obstacle avoidance for underwater vehicles necessitates high resolution, reliable, multi-beam sonars of this type (Lane, 2001). The probability of detection is based on a cookie-cutter approach in which the probability of detection is unity within the scan area and zero anywhere else. Bearing is measured to the nearest degree and range is measured every meter.

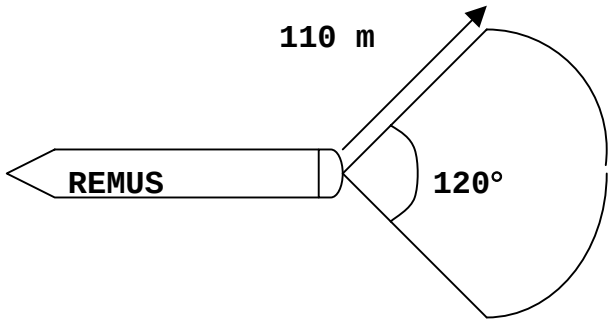



Figure 5. Forward-look Sonar Model

The advantage of using a forward-looking sonar over side-scan sonars in object avoidance is twofold. One, it allows for scanning ahead of the vehicle which facilitates reaction to detected obstacles, and two, it allows for possible overlap of acoustic imagery ahead of the vehicle providing more accurate detection information. REMUS is currently configured with two side-scan sonars. Based on the swath width of the sonar, REMUS must make narrow passes

over a given area at 15-40 meter increments for adequate coverage of the sea floor. A forward look sonar, while more difficult to mount, would prove more capable in preventing collision and would allow for mapping a more efficient path in cluttered environments.

## C. HEURISTICS

There are several methods used for obstacle avoidance in robot vehicles today. (Several are outlined in Chapter 1.) The obstacle avoidance model developed in this thesis is based on the product of bearing and range weighting functions that form the gain factor for a dynamic obstacle avoidance behavior. The basis for the weighting functions lies in a fuzzy logic methodology. The weighting functions are MATLAB membership functions from the fuzzy logic toolbox with the parameters selected to maximize obstacle avoidance behavior. The membership function for bearing is a Gaussian curve function of the form:

$$w1 = 10^{\frac{-(x-c)^2}{(2\sigma^2)}} \tag{46}$$

where the parameters x, c, and $\sigma$ are position (or angular position in degrees for the purpose of this model), center, and shape respectively. Shape defines the steepness of the Gaussian curve. The values selected for these parameters to provided sufficient tuning in this membership function were -90:90, 0, 20 respectively. The bearing weighting function can be seen in Figure 6 below.

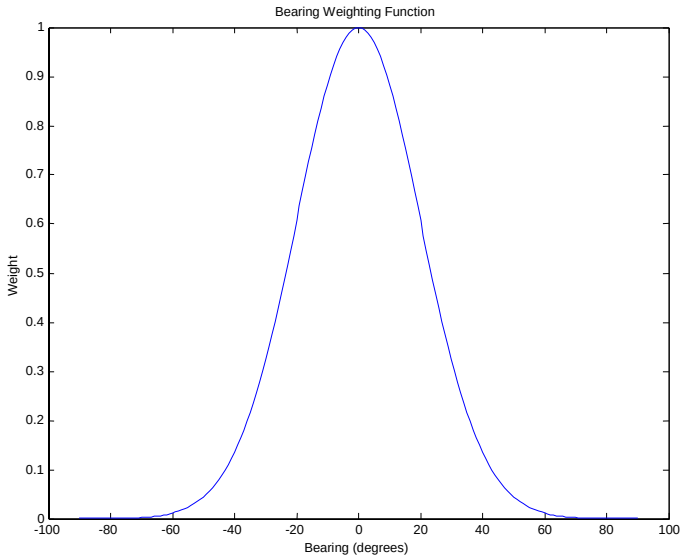



Figure 6. Bearing Weighting Function

It is evident that the weight given to an object dead ahead of the vehicle is closer to unity than one that is over 30° to port or starboard.

The membership function for range is an asymmetrical polynomial spline-based curve called zmf for its z shape and is of the form

$$w2 = zmf(x,[a\ b]) \tag{47}$$

where *a* and *b* are parameters that locate the extremes of the sloped portions of the curve. These parameters are called breakpoints and define where the curve changes concavity. In order to maximize obstacle avoidance behavior, these values were tuned to be (sonrange-99) and (sonrange-90). With this selection, the range weight is approximately unity for anything closer than 20 meters and zero for anything farther than 40 meters from REMUS. The range weighting function can be seen in Figure 7 below.

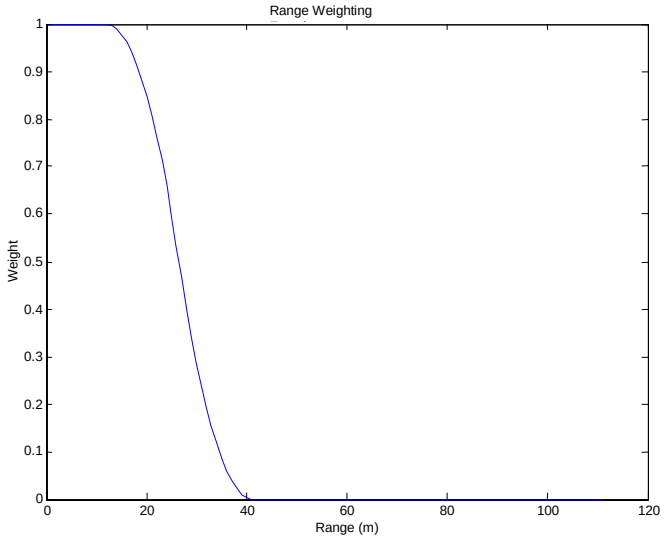



Figure 7. Range Weighting Function

A final weight based on both bearing and range is calculated from the product of *w1* and *w2*. This weight becomes the gain coefficient that is applied to a maximum avoidance heading for each individual object. The maximum heading is $\pi/4$ as seen below:

$$\psi_{oa}(t,c) = w1w2(\pi/4) \tag{48}$$

where *t* is the time step and *c* is the obstacle being evaluated. The avoidance heading for all obstacles over a single time step (or one look) is then

$$\psi_{oalook}(t) = \sum_{1}^{c} \psi_{oa}(t,c) \tag{49}$$

Following an evaluation of each obstacle at every time step, a final obstacle avoidance heading term is determined from the sum of the obstacle avoidance heading of each individual object within a specified bearing and range from the vehicle or

$$\psi_{oatot}(t) = \frac{\psi_{oalook}(t)}{cc} \tag{50}$$

where *cc* is the counter used to determine how many obstacles fall into this window. The counter is used to normalize this overall obstacle avoidance term to an average for all of the obstacles within the range above. This bearing and range of the window is determined through a rough evaluation of the weighting functions. In order to fall into the window, the gain factor must be equal to or exceed a value of *w1w2*=0.15.

The obstacle avoidance term $\psi_{oatot}(t)$ is then incorporated into vehicle heading error (discussed in Chapter 3, equation (43)) as:

$$\tilde{\psi}_{LOS}(t) = \psi_{track}(t) - \psi_{cont}(t) - \arctan(cte(t)/rabbit) + \psi_{oatot}(t) \tag{51}$$

This heading error drives the rudder commands to maneuver around detected objects in the track path. The overall object avoidance system dynamics can be seen in the diagram below:

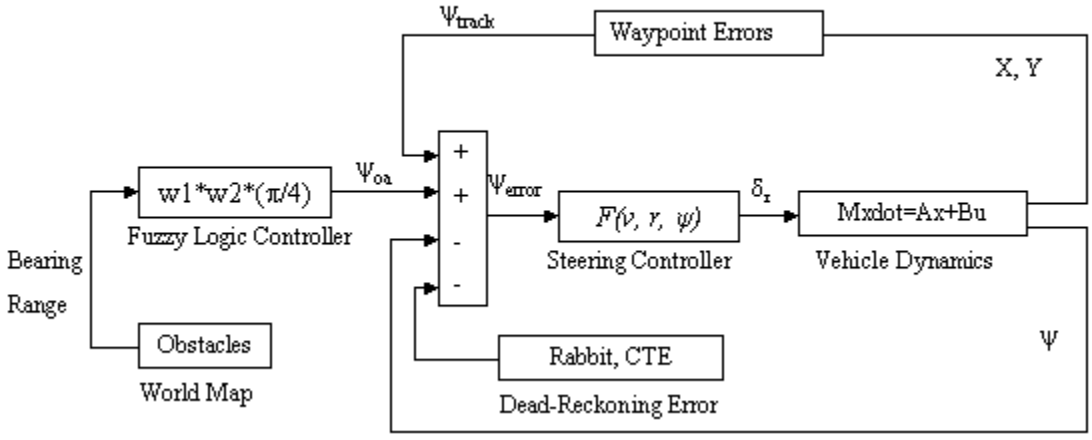



Figure 8. Block Diagram System Dynamics

# V. VEHICLE SIMULATION

## A. BASIC SINGLE POINT OBSTACLE AVOIDANCE

The initial test performed on the two-dimensional sonar model was navigation around a single point obstacle. This is the simplest obstacle avoidance test for the 2-D model. Three variations of this test were run for the basic single point obstacle avoidance. The first was for a single point on the path. The second was for a single point to the right or left of the path. Finally, a run was performed to test the accuracy of the steering and obstacle avoidance model for each of the four quadrants. This was achieved by running the REMUS through a figure-eight path that had a single point obstacle at the midpoint of each leg. Results for single point obstacle runs can be seen in the figures below. The first two tests were repeated for a cluster of points designed to mimic an obstacle with length and width both on the path and just off the path and will be addressed in the next section.

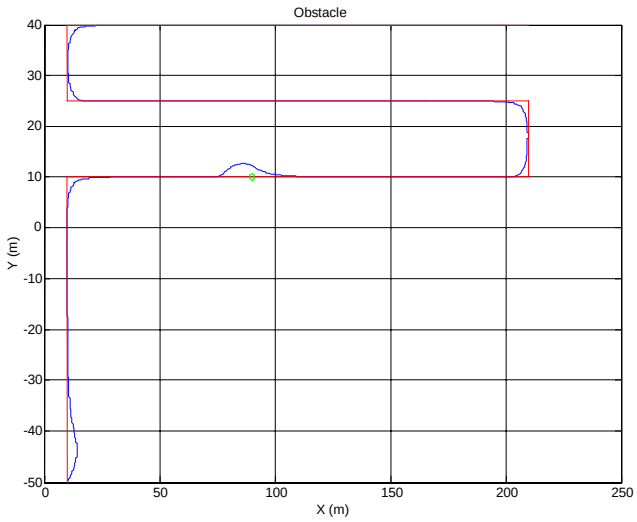



Figure 9. Single Point Obstacle Run (On Path)

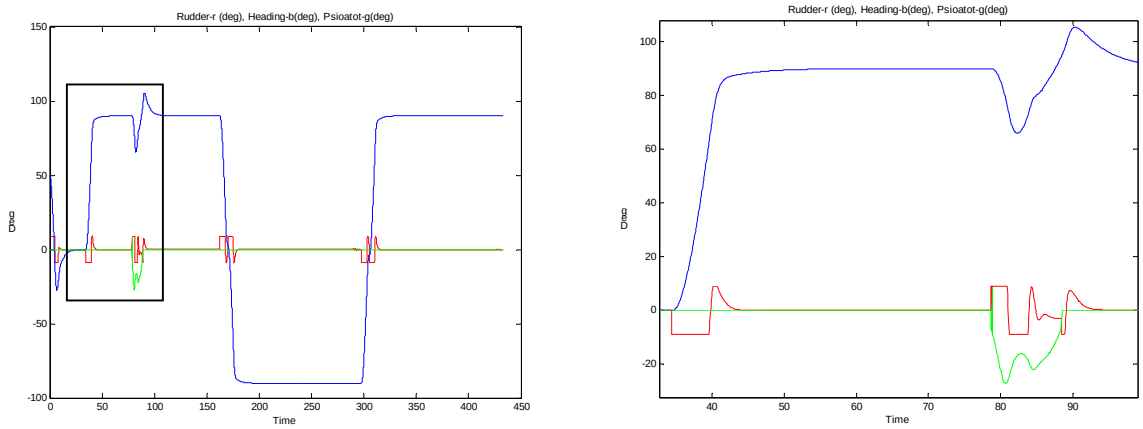



Figure 10. Single Point Obstacle Run: Rudder/Heading/$\psi_{oa}$

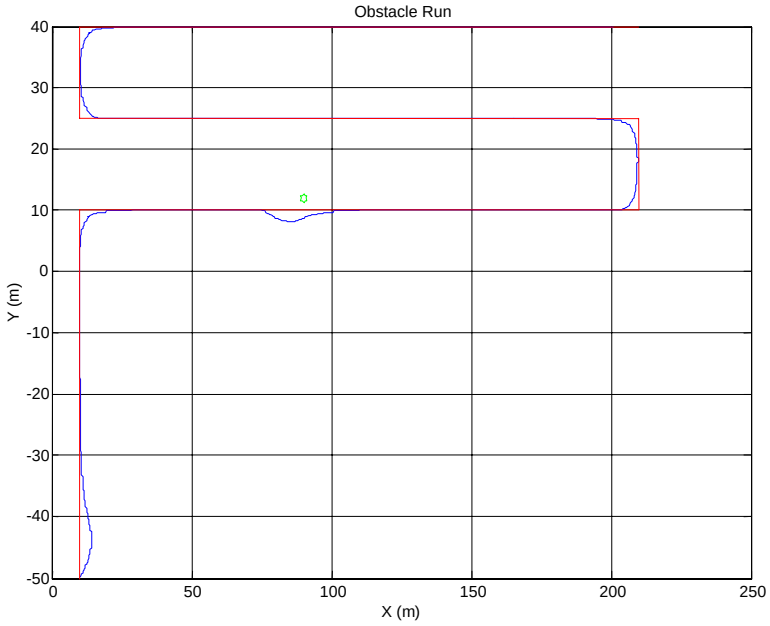



Figure 11. Single Point Obstacle Run (Off Path)

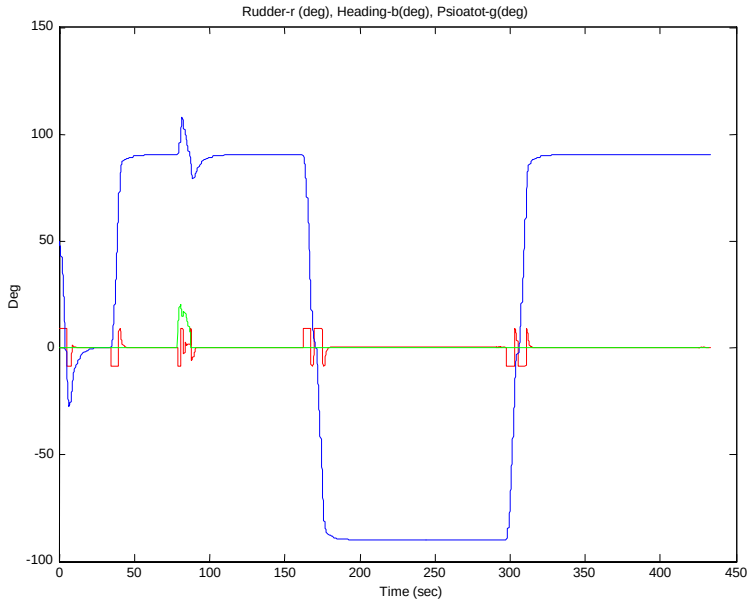



Figure 12. Rudder/Heading/$\psi_{oa}$ (Off Path)

Figures 10, 12, and 14 show the rudder dynamics, vehicle heading, and obstacle avoidance heading term for the duration of each vehicle run. The rudder action has a direct correlation with the obstacle avoidance heading and overall vehicle heading. The large angle motions of the heading are the ninety-degree turns made to track the ordered vehicle path. There is an associated rudder action with each of these turns as seen by the corresponding rudder curve. These rudder curves show that the maximum programmable rudder is 9°. For all dynamic behaviors, whether associated with a turn or obstacle avoidance maneuver, the rudder initiates the turn with this maximum value. In order to regain track, the rudder action may vary. The major heading changes to track the path require a full rudder for a longer period of time than do the obstacle avoidance heading changes. This is evident in the constant horizontal value on the rudder curve. The

difference in Figure 9 and Figure 11 is in the direction of turn to maneuver around the obstacle. When the obstacle is on the path, the vehicle maneuvers to the left. When it is off the path, the vehicle maneuvers to the opposite side of the obstacle.

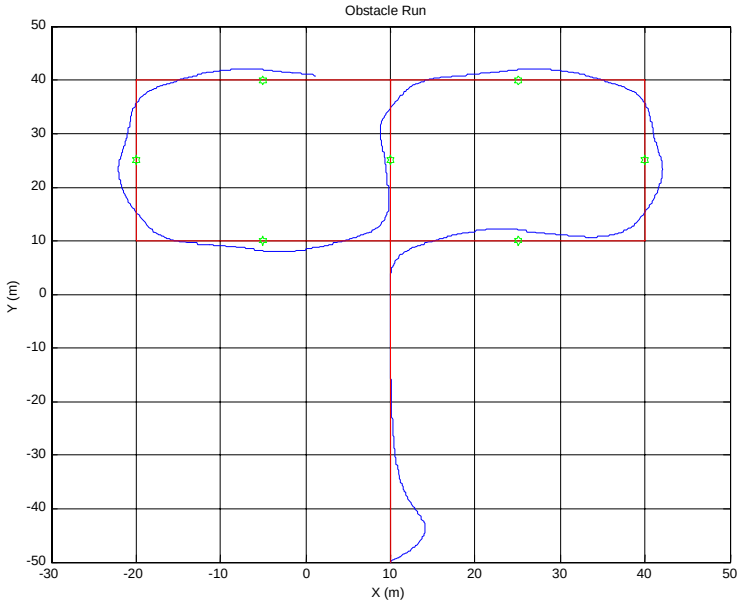



Figure 13. Figure-Eight Obstacle Run

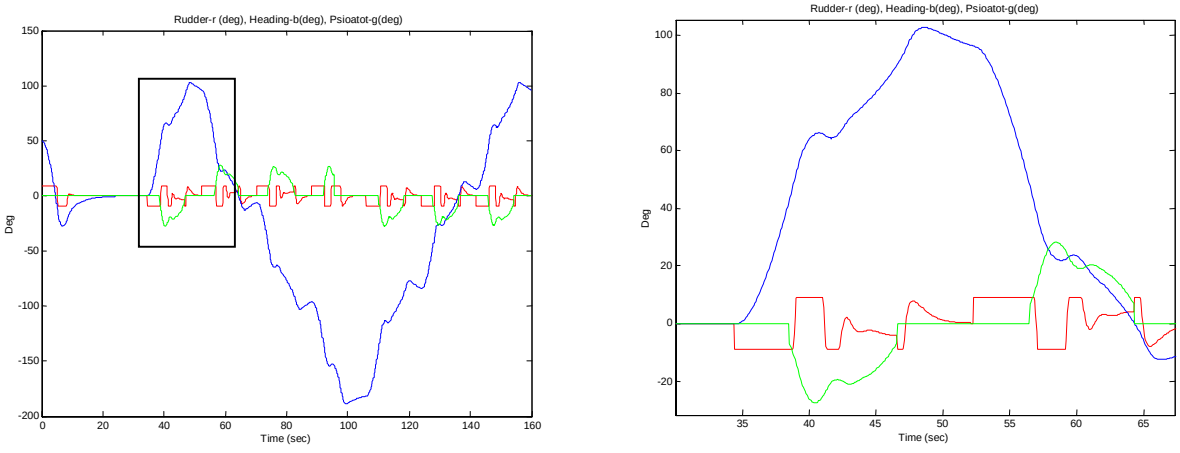



Figure 14. Rudder/Heading/$\psi_{oa}$ Figure-Eight

Figures 13 and 14 show the results for the vehicle run through the figure-eight path. Although the vehicle does not maintain the track as accurately as it does the previous runs, it completes the run with proper dynamics for each quadrant. The obstacle avoidance heading is not equal for each of the obstacle avoidance behaviors due to the fact that the vehicle is not weighing the same number of obstacles along each leg of the figure-eight. It only weights the obstacles that fall within the scan with the proper proximity as described in Chapter IV.

## B. MULTIPLE POINT OBSTACLE AVOIDANCE

A single point obstacle avoidance model is far simpler than a multiple point obstacle avoidance model not only in the maneuvering of the vehicle, but also in maintaining the obstacle picture. For multiple point obstacle avoidance, it is necessary to have a model that reacts to obstacles in a certain proximity to its path rather than all possible obstacles seen by the sonar scan. Weighting functions allow for an accurate compilation of this obstacle picture. The REMUS model builds an obstacle counter for obstacles having a weighting function product greater than 0.15 as discussed in the previous chapter. This value allows for a maximum rudder and bearing weight of approximately 0.386, the square root of 0.15. Referring to the membership functions in Figure 6 and Figure 7, a value of 0.386 correlates to a bearing and range of approximately +/-30° and 30 meters respectively.

As seen in the following figures, REMUS successfully avoids multiple points and multiple point clusters in the same fashion it avoided a single points. The rudder

dynamics are minimal during all avoidance maneuvers for an efficient model. All of the obstacle runs, for single point or multiple point obstacle avoidance, show REMUS responding to obstacles in advance of the actual obstacle position. While this model has not been optimized with refined techniques, the early response time would allow sufficient processing time in an actual sonar return for real-world environments. The dynamics of REMUS are very reactive such that REMUS regains the track path directly after the passing an obstacle. Though this behavior is not ideal due to the proximity at which REMUS passes the obstacle, through optimization, it could be improved.

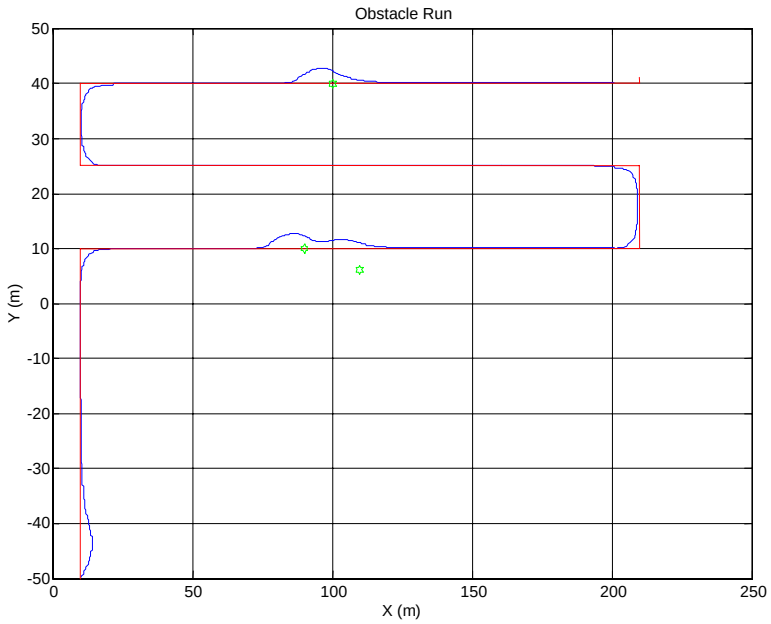



Figure 15. Multiple Single Point Obstacle Run

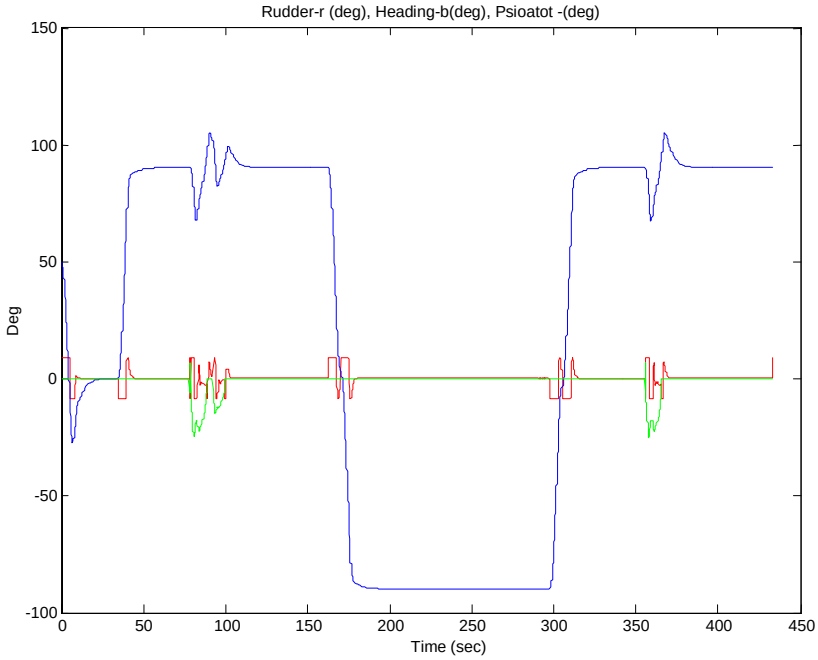



Figure 16. Rudder/Heading/$\psi_{oa}$

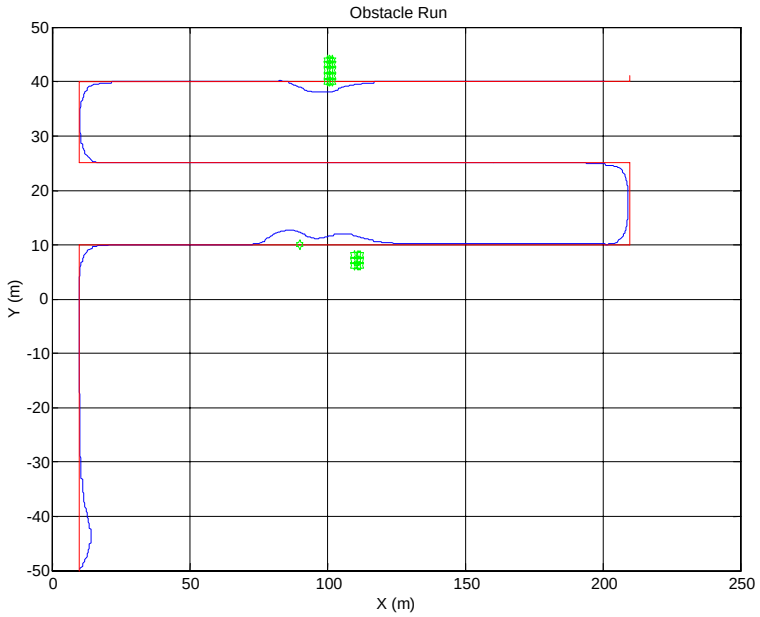



Figure 17. Multiple Point Obstacle Run

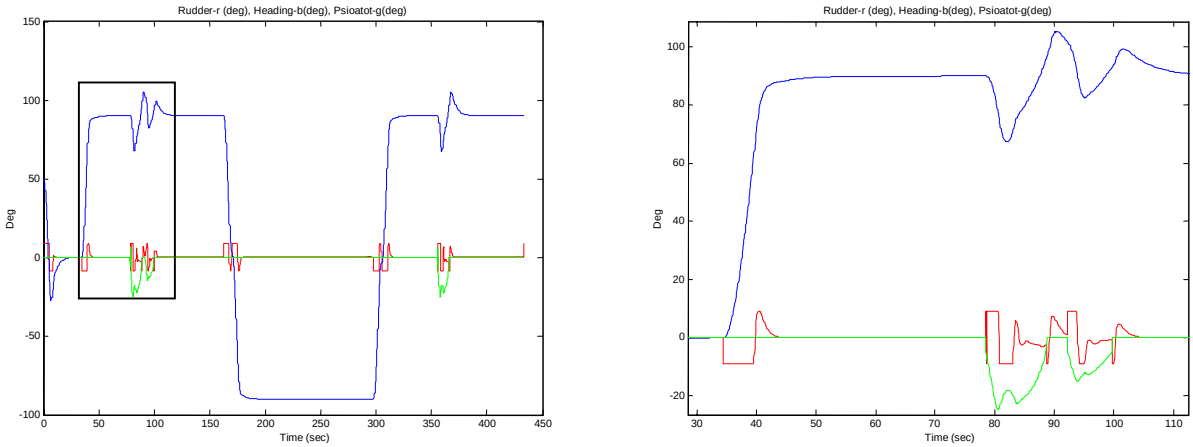



Figure 18. Multiple Point Obstacle Run: Rudder/Heading/$\psi_{oa}$

The vehicle heading in Figure 18 (bottom right) can be offset by 90° in or to compare the vehicle dynamics with the obstacle avoidance heading. As seen in Figure 19 below, an obstacle appearing in the vehicle path causes the vehicle heading to deviate from its track path heading of 90° approximately the same amount as the obstacle avoidance heading. These two headings do not exactly match because the total heading incorporates additional factors as in equation (51).

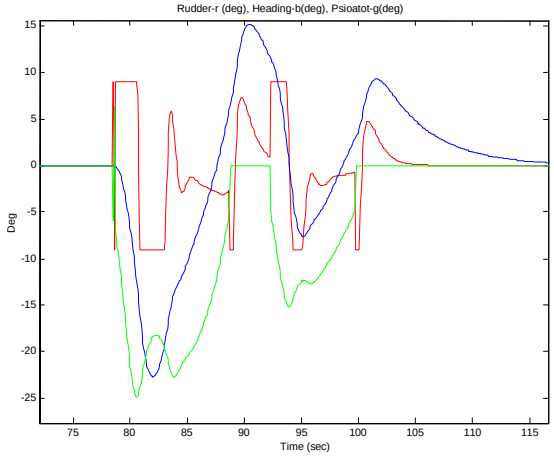



Figure 19. Vehicle Heading Comparison with 90° Offset

The above figures present the obstacle avoidance for the weighting functions described in the previous chapter. A comparison can be made for different values of the weighting functions to show the utility of the selected functions. A range weighting function that uses breakpoints defined at (sonrange-95) and (sonrange-70) changes the vehicle dynamics around the obstacles. Figure 20 shows the curve for this alternate weighting function.

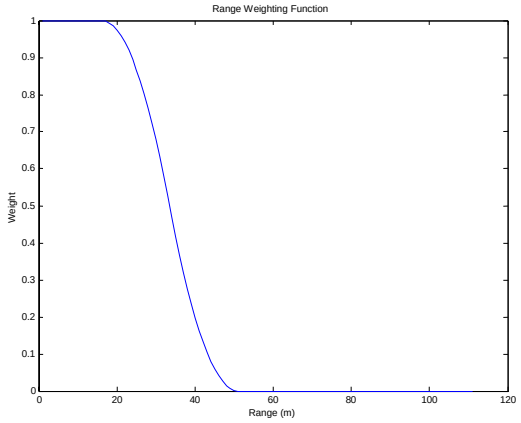
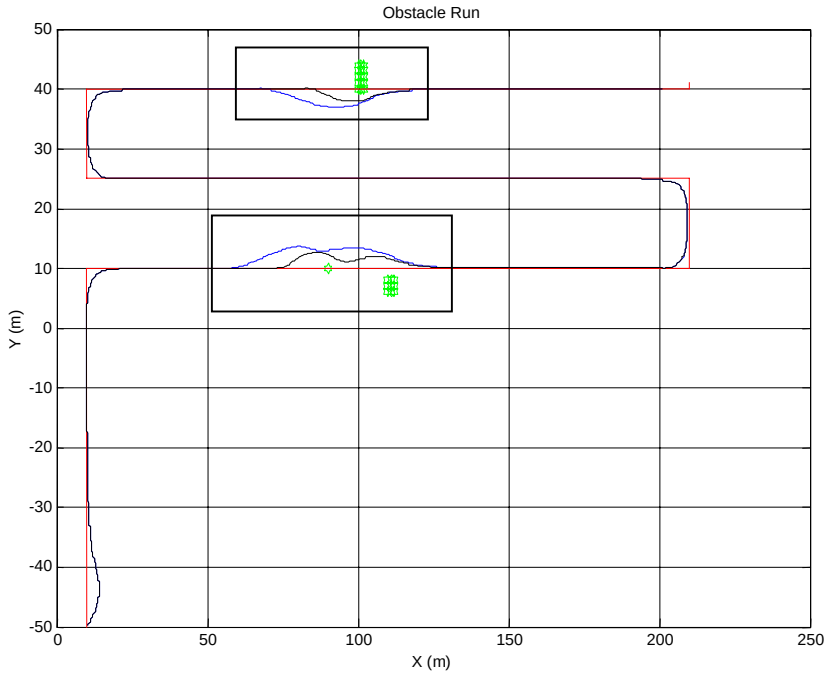
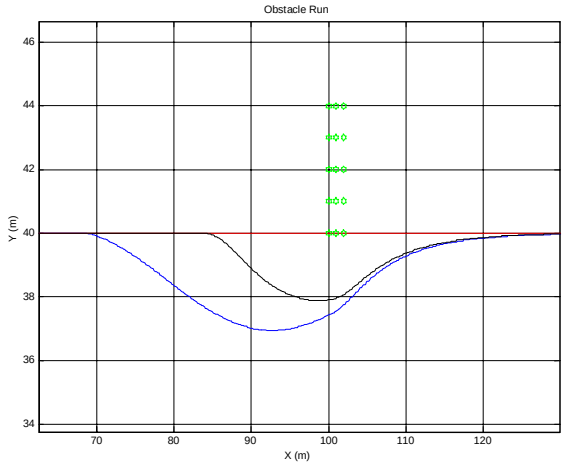



Figure 20. Alternate Range Weighting Function

The vehicle response is too early with the alternate breakpoints, although the off track distance increases by approximately a half meter. For mine countermeasures operations, a higher off track distance increases vehicle safety. However, the sonar configuration of REMUS supports side scan imaging as well as possible forward looking. Thus, minimizing off track distance is more ideal for obtaining accurate side scan data. Figure 21 shows the dynamic behavior comparison of the two weighting functions.



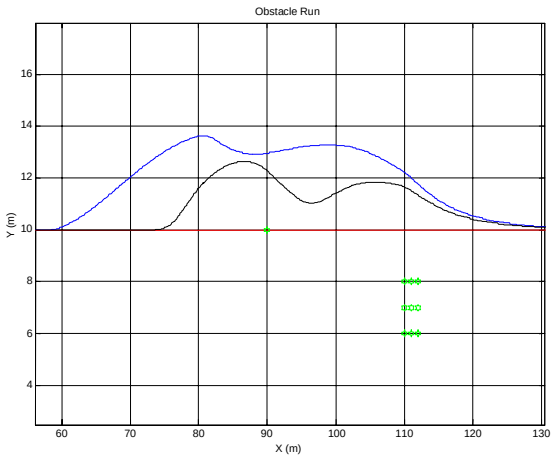




Figure 21. Range Weighting Function Dynamics Comparison

# VI. CONCLUSIONS AND RECOMMENDATIONS

## A. CONCLUSIONS

Obstacle avoidance for autonomous vehicles is widely studied for a variety of applications. This thesis focuses on a particular application for the REMUS AUV. One of the most critical factors in obstacle avoidance behavior is the ability to discern how a vehicle will react to its environment. It is necessary to model realistic sensors that gather sufficient environmental data for safe vehicle navigation. The sensor modeled in this thesis will be used by the Center for Autonomous Underwater Vehicle Research in future operations and requires an accurate model prior to implementation. The model shows that with appropriate onboard processors, the REMUS vehicle could, if necessary, execute a local reflexive maneuver. REMUS has the ability to use range and bearing data from a sonar return to determine if that return constitutes a threat along its proposed path and further navigate around the threat before regaining its original path. Through weighting functions, nonlinear path deviations can be achieved to avoid these threats, while still scanning the underwater environment for possible mines and other environmental data.

There remains a need for a fast and effective means of interpreting the sonar data. Visual analyses of sonar returns are made daily in naval applications. This ability has to be effectively implemented in an underwater vehicle for obstacle avoidance to be successful. One method would be through analysis of shadow areas in sonar returns.

Often, a sonar scan does not pick up the same obstacle each time it passes over a given area. However, multiple scans with positive detection over a decreasing range will allow for the processing system to correlate a positive detect on a specific bearing and range to an obstacle. Thus, the model developed in this thesis accurately represents a sonar in that on each time step, the vehicle sees every object within the bearing and range of the scan.

A last point to be made for this model is a concern for the overuse of actuators for dynamic movements. In a multiple point obstacle field with several dynamic movements, the vehicle has a significant number of rudder "bangs" or direction changes in a very short period of time. This dynamic rudder action will dissipate power and will quickly wear out the servomechanisms. Thus, a more robust design, or one that eliminates response to non-hazardous obstacles, might be necessary in a high clutter environments.

## B. RECOMMENDATIONS

There are many areas in which this thesis work can be improved upon to build a more complete and robust obstacle avoidance model for the REMUS vehicle. The most obvious but most complicated of these is the development of a three dimensional model. This would require adding a depth to the sonar scan such that the scan would cover somewhere from ten degrees above the horizontal to thirty degrees below the horizontal. In order to implement such a model, the vehicle EOM would have to be modified to include diving and climbing maneuvers for obstacle avoidance. To produce

a more exact model, it would be necessary to conduct an open water test with the REMUS vehicle to determine hydrodynamic coefficients for diving as well as steering. Additionally, the incorporation of a CTE controller to the steering model once experimental data is obtained would make it more robust. A CTE controller is not functional in the model at present due to the lack of experimental values for coefficients in the CTE equations.

A second addition to the proposed model that would increase its utility would be through speed control. A model with acceleration and deceleration capability would allow for more dynamic obstacle avoidance. For example, if REMUS turned to an area of increased obstacles due to an obstacle avoidance command from some other object in its path, a speed reduction could follow to permit data processing prior to driving a new path.

A speed controller would be useful for a model that incorporates moving obstacles as well as stationary. The proposed model uses only stationary obstacles in the vehicle path. By incorporating a range rate variable into the avoidance control, the vehicle could compare it’s own speed with the relative speed at which it closes the obstacle and thus determine if the detected obstacle is moving. Use of range rate data would allow REMUS to better determine the safest path around obstacles.

The Fuzzy Logic methodology used to develop weighting functions for obstacle avoidance behavior may not be the most accurate method available. However, simple additions to this model could make it more accurate, such as using range rate as a weighting factor. Additionally, an

optimization could be performed on the implementation of the weighting function gain factor so that REMUS clears each obstacle by a specified distance, does not begin avoidance behavior too early, and does not return to track at such sharp angles.

Finally, errors in vehicle position and sensory information must be taken into account for dynamic behaviors to be accurate. Currently, REMUS navigates a track through transponder cross-fix data that has about a 2-3 meter positional error associated with it. While the steering model runs under the assumption that REMUS no longer uses these transponders, GPS position errors may still be a factor. If future REMUS vehicles can operate using autopilots in the steering model, there will be only slight errors in sensory information as the position of the obstacles are in a local frame of reference with respect to the vehicle. Additionally, the REMUS obstacle avoidance model uses these relative positions to plan reflexive maneuvers. Through Concurrent Mapping and Localization (CML) techniques, REMUS could store obstacles it has passed in a database for use in planning a return path to its original position or for a possible rendezvous for data transfer. Ruiz (2001) gives a thorough overview of CML techniques.

# APPENDIX A

```
% This mfile uses corrected hydrodynamic coeff from MIT to
develop
% a steering model.  It models REMUS running through a field of
% multiple obstacles, both single points and those with lenght
% and width.
clear
clf
clc

% REMUS Characteristic Specifications:

L = 1.33;          % Length in m
W = 2.99e02;        % Weigth in N
g = 9.81;            % Acceleration of gravity in m/s^2
m = W/g;            % Mass in kg
V = 1.543;          % Max Speed in m/s
rho = 1.03e03;      % Density of Salt H20 in kg/m^3
D = .191;           % Max diameter in m

%State Model PArameters
U = 1.543; % m/s
Boy = 2.99e02;
xg  = 0; yg = 0; zg = 1.96e-02; % in m

Iy = 3.45; %kg/m^3 (from MIT thesis)
Iz=Iy;

% MIT REMUS Coeff (Dimensionalized)

disp('MIT REMUS Coefficients');

Nvdot = 1.93;
Nrdot = -4.88;
Yvdot = -3.55e01;
Yrdot = 1.93;
%Nv = -4.47; should be same as Mw which is stated as +30.7
% should be -9.3 but going by Hoerner eqn, we get about 4.47
Nv = -4.47;
Nr = -6.87; %Same as Mq;
Yv = -6.66e01; %Same as Zw; Note should be -6.66e1 from MIT
thesis not 2.86e01
Yr = 2.2 ; %Same as Zq = 2.2; MIT has miscalculation
Nd = -3.46e01/3.5; % Nd and Yd scaled by 3.5 to align w/exp data
Yd = 5.06e01/3.5;

% The Steering Equations for the REMUS are the following.
% These equations assume the primarily horizontal motions ...

MM=[(m-Yvdot) -Yrdot 0;-Nvdot (Iz-Nrdot) 0;0 0 1];
AA=[Yv (Yr-m*V) 0;Nv Nr 0; 0 1 0];
BB=[Yd;Nd;0];
```

```
A=inv(MM)*AA; B=inv(MM)*BB; C=[0,0,1]; D=0;

A2=[A(1:2,1),A(1:2,2)];B2=[B(1);B(2)];
xss=inv(A2)*B2;
poles = eig(A2);
RadGy = sqrt(Iz/(W/g)); % in meters
RadCurv = U/(xss(1)); % in meteres
SideSlip = atan2(xss(1),U)*180/pi; % in deg/s

[num,den]=ss2tf(A,B,C,D); z=roots(num); p=roots(den);

% Desired closed loop poles for sliding:

 k=place(A,B,[-1.4,-1.45,0.0]);

% Closed loop dynamics matrix

Ac=A-B*k;
[m,n]=eig(Ac');
S=m(:,3);

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

TRUE  = 1;
FALSE = 0;

DegRad = pi/180;
RadDeg = 180/pi;

% Define Obstacles:(put them in near track for trial runs)

Xo(1) = 10; % First obstacle x-dist ref global origin in m
owidth(1) = 1; % First obstacel width in m
Yo(1) = 90; % First obstacle y-dist ref global orinin in m
olgth(1) = 1; % First object length in m

Xo(2) = 40; % Second obstacle x-dist ref global origin in m
owidth(2) = 5; % Second obstacle width in m
Yo(2) = 100; % Second obstacle y-dist ref global origin in m
olgth(2) = 3; % Second obstacle length in m

Xo(3) = 6; % Third obstacle x-dist ref global origin in m
owidth(3) = 3; % Third obstacle width in m
Yo(3) = 110; % Third obstacle y-dist ref global origin in m
olgth(3) = 3; % Third obstacle length in m

numobs = 3;
numpts = 0;
for p=1:numobs
    numpts=numpts + owidth(p)*olgth(p);
end

AreaObs = [];

psioa=zeros(8000,numobs);
```

```
%  Define Sonar Grid Parameters:

sonrange = 110; % radial range in m based on 400 KHz frequency
theta = 2*pi/3; % angular arc in rad

% Builds obstacles in Xo and Yo matrices:
Xobs=[]; Yobs=[];
for p = 1:numobs % model each point as an obstacle so the sonar
can see them individually
    for pp=1:olgth(p)
        if owidth(p)>1
            for q = 1:(owidth(p))
                Xobs=[Xobs,(Xo(p)+(q-1))];
                Yobs=[Yobs, (Yo(p)+(pp-1))];
            end
        elseif owidth(p)==1
            Xobs=[Xobs, Xo(p)];
            Yobs=[Yobs, Yo(p)];
        else
            Xobs = Xobs;
            Yobs = Yobs;
        end
    end
end

% Set time of run

dt = 0.125/2;
t = [0:dt:1800]';
size(t);

% Set initial conditions
start=10;
v(1)   = 0.0;
r(1)   = 0.0;
rRM(1) = r(1);

% This is the Initial Heading of the Vehicle
psi(1) = 50.0*DegRad;

% This is the Initial Position of the Vehicle
X(1) = -50.0; % Meters
Y(1) = 10;

% This data from track.out file
No_tracks=7;
Track=[10.0 10.0   2.75 2.75  0  1.25  1.00 0 25.00 8.00 40.00
        10.0 210.0  2.75 2.75  0  1.25  1.00 0 25.00 8.00 200.00
        25.0 210.0  2.75 2.75  0  1.25  1.00 0 25.00 2.00 15.00
        25.0 10.0   2.75 2.75  0  1.25  1.00 0 25.00 2.00 200.00
        40.0 10.0   2.75 2.75  0  1.25  1.00 0 25.00 2.00 15.00
        40.0 210.0  2.75 2.75  0  1.25  1.00 0 25.00 2.00 200.00
        41.0 210.0  2.75 2.75  0  1.25  1.00 0 25.00 2.00 1.0];
track=Track(:,1:2);
SurfaceTime = Track(:,9);
```

```
SurfPhase   = Track(:,8);

% Read in wayopoints from track data assumes track is loaded
for j=1:No_tracks,
   X_Way_c(j)     = track(j,1);
   Y_Way_c(j)     = track(j,2);
end;

PrevX_Way_c(1) = X(1);
PrevY_Way_c(1) =  Y(1);

r_com = 0.0;

% Set Rudder angle saturation:

sat = 9; % Degrees

% Set Watch Radius:

W_R = 2.0;

% Set dead-reckoning/look-ahead distance:

rabbit = 9;


x(:,1) = [v(1);r(1);psi(1)];

Eta_FlightHeading = 0.5;    % Lowered this from 1.0 on AERIES
model
Phi_FlightHeading = 0.1;    % Lowered this from 0.5 on AERIES
model

% Below for tanh
Eta_CTE = 0.05; % (NA given that no CTE controller is used)
Eta_CTE_Min = 1.0;
Phi_CTE = 0.2;  % (NA given that no CTE controller is used)

   Uc = [];
   Vc = [];

SegLen(1) = sqrt((X_Way_c(1)-PrevX_Way_c(1))^2+(Y_Way_c(1)-
PrevY_Way_c(1))^2);
psi_track(1) = atan2(Y_Way_c(1)-PrevY_Way_c(1),X_Way_c(1)-
PrevX_Way_c(1));

for j=2:No_tracks,
   SegLen(j) = sqrt((X_Way_c(j)-X_Way_c(j-1))^2+(Y_Way_c(j)-
Y_Way_c(j-1))^2);
   psi_track(j) = atan2(Y_Way_c(j)-Y_Way_c(j-1),X_Way_c(j)-
X_Way_c(j-1));
end;

  j=1;
  Sigma = [];
  Depth_com = [];
```

```
  dr=[];
  drl = [];
  drl(1) = 0.0;

  Depth_com(1) = 5.0;
  WayPointVertDist_com = [5.0 5.0 5.0 5.0 5.0 5.0 5.0];

for i=1:length(t)-1
   Depth_com(i) = WayPointVertDist_com(j);

   X_Way_Error(i) = X_Way_c(j) - X(i);
   Y_Way_Error(i) = Y_Way_c(j) - Y(i);

   % DeWrap psi to within +/- 2.0*pi;
   psi_cont(i) = psi(i);

   while(abs(psi_cont(i)) > 2.0*pi)
      psi_cont(i) = psi_cont(i) - sign(psi_cont(i))*2.0*pi;
   end;

   psi_errorCTE(i) = psi_cont(i) - psi_track(j);

   % DeWrap psi_error to within +/- pi;
   while(abs(psi_errorCTE(i)) > pi)
      psi_errorCTE(i) = psi_errorCTE(i) -
sign(psi_errorCTE(i))*2.0*pi;
   end;


% **  Always Calculate this
     % Beta = v(i)/U;
      Beta = 0.0;
      cpsi_e = cos(psi_errorCTE(i)+Beta);
      spsi_e = sin(psi_errorCTE(i)+Beta);

      s(i) = [X_Way_Error(i),Y_Way_Error(i)]*...
             [(X_Way_c(j)-PrevX_Way_c(j)),(Y_Way_c(j)-
PrevY_Way_c(j))]';

      % s is distance to go projected to track line(goes from 0-
100%L)
      s(i) = s(i)/SegLen(j);

      Ratio=(1.0-s(i)/SegLen(j))*100.0;

      % ss is the radial distance to go to next WP
      ss(i) = sqrt(X_Way_Error(i)^2 + Y_Way_Error(i)^2);

      % dp is the angle between line of sight and current track
        line
      dp(i) = atan2( (Y_Way_c(j)-PrevY_Way_c(j)),(X_Way_c(j)-
            PrevX_Way_c(j)) )-atan2(Y_Way_Error(i),X_Way_Error(i)
            );

      if(dp(i) > pi),
         dp(i) = dp(i) - 2.0*pi;
```

```
end;

cte(i) = s(i)*sin(dp(i));

if( abs(psi_errorCTE(i)) >= 00.0*pi/180.0) %| s(i) < 0.0 ),
      %used to read 40.0*pi not 00.0*pi for CTE controller
   % Use LOS Control
   LOS(i) = 1;

   psi_comLOS(i) = atan2(Y_Way_Error(i),X_Way_Error(i));
   %psi_comLOS = pi/2; % Test for heading controller
                         stability

   % Construct Bearing/Range to each obstacle(point):
   cc=0;
   psioalook(i)=0;
   increaseweight=FALSE;
   for c=1:numpts
      Bearing(i,c) = atan2((Yobs(c)-Y(i)),(Xobs(c)-X(i)))-
                     psi(i);
      Range(i,c) = sqrt((Yobs(c)-Y(i))^2+(Xobs(c)-X(i))^2);
      if Range(i,c)<=sonrange & (-
                    theta/2<=Bearing(i,c)<=theta/2)
     % Use Fuzzy logic
          if c>1
              sepang=abs(Bearing(i,c)-Bearing(i,(c-1)));
              if ((Range(i,c))^2 + (Range(i,(c-1)))^2 –
                  2*Range(i,c)*Range(i,(c-1))*...
                      cos(sepang))<2*D
                  increaseweight=TRUE;
              end
          end
          % Develop weighting factor based on Range (w1)
          w = 0:1:sonrange;
          w1(i)=zmf(Range(i,c), [(sonrange-99) (sonrange-
               90)]);
          % [] above are breakpoints in the curve

          % Develop weighting factor based on Bearing (w2)
          Posit = (-90:1:90)'; % in degrees
          Center = 0;
          Shape = 20;
          w2(i)=gaussmf(Bearing(i,c)*RadDeg, [Shape,
                Center]);
          % A MATLAB membership function: EXP(-(Posit –
            Center).^2/(2*Shape^2))

          if increaseweight
              w1(i)=2*w1(i);
              w2(i)=2*w2(i);
          end

         % Only want to weight the obstacle c once in each
           time step
            if (w1(i)*w2(i))>0.15
               cc=cc+1; % counter for obstacles in being
```

```
                            avoided at time t
                    % Object Bears to Left
                     if Bearing(i,c)>0
                         psioa(i,c)=-w1(i)*w2(i)*(pi/4);

                    % Object Bears to Right
                     elseif Bearing(i,c)<=0
                         psioa(i,c)=+w1(i)*w2(i)*(pi/4);
                     end
                     psioalook(i)=(psioalook(i)+psioa(i,c));
                end
            end
        end
        if cc>0
            psioatot(i)=psioalook(i)/cc;
        else
            psioatot(i)=psioalook(i);
        end

        psi_errorLOS(i) = psi_track(j) - psi_cont(i)-
                  atan2(cte(i),rabbit) + psioatot(i);

        if(abs(psi_errorLOS(i)) > pi),
           psi_errorLOS(i) = ...
           psi_errorLOS(i) –
    2.0*pi*psi_errorLOS(i)/abs(psi_errorLOS(i));
        end;

        Sigma_FlightHeading(i) = (-S(1,1)*v(i))*0.0+S(2,1)*(r_com
                                  – r(i)) + S(3,1)*psi_errorLOS(i);
        % Have taken out v influence in Sigma_FlightHeading above

      dr(i) = (-k(1,1)*v(i))*0.0-k(1,2)*r(i)-
      Eta_FlightHeading*tanh(Sigma_FlightHeading(i)/Phi_FlightHea
      ding);
         %dr(i) = -k(1)*v(i)-k(2)*r(i)+k(3)*(psi_errorLOS(i));
         %if ((i>1200)&(i<1400)); dr(i)=-4*DegRad; end; % turn
test

   else

      % Use CTE Controller
      LOS(i) = 0;
      if(cpsi_e ~= 0.0), % Trap Div. by Zero !

      % SMC Soln

      Sigma(i) = U*rRM(i)*cpsi_e + Lam1*U*spsi_e + Lam2*cte(i);

      dr(i) = (1.0/(U*a*cpsi_e))*(-U*b*rRM(i)*cpsi_e +
               U*rRM(i)^2*spsi_e-Lam1*U*rRM(i)*cpsi_e –
               Lam2*U*spsi_e –2.0*Eta_CTE*(Sigma(i)/Phi_CTE));

      else
       dr(i) = dr(i-1);
      end;
```

```
   end; % End of CTE Controller

   if(abs(dr(i)) > sat*pi/180) % change from 0.4 radians on ARIES
      dr(i) = sat*pi/180*sign(dr(i));
   end;

   % State Variable Formulation:

   x_dot(:,i+1) = [ A(1,1)*v(i) + A(1,2)*r(i) + B(1)*dr(i);
                    A(2,1)*v(i) + A(2,2)*r(i) + B(2)*dr(i);
                    r(i)];
   x(:,i+1) = x(:,i)+dt*x_dot(:,i);
   v(i+1)   = x(1,i+1);
   r(i+1)   = x(2,i+1);
   psi(i+1) = x(3,i+1);
   rRM(i+1) = r(i+1);

    % Wave Motions:

    Uc = 0.0;
    Vc = 0.0;

    %Kinematics

   X(i+1) = X(i) + (Uc + (U)*cos(psi(i)) - v(i)*sin(psi(i)))*dt;
   Y(i+1) = Y(i) + (Vc + (U)*sin(psi(i)) + v(i)*cos(psi(i)) )*dt;

%  Check to See if we are Within the Watch_Radius or if we passed
   the WP
%  Change to next WP if radial distance to go is less than rabit
   distance or if we passed the WP or if we are within the WR

   if(sqrt(X_Way_Error(i)^2.0 + Y_Way_Error(i)^2.0) <= W_R | s(i)
      < 0.0 | ss(i)<rabbit),
   disp(sprintf('WayPoint %d Reached',j));
      if(j==No_tracks),
         break;
      end;
      PrevX_Way_c(j+1) = X_Way_c(j);
      PrevY_Way_c(j+1) = Y_Way_c(j);
      j=j+1;
   end;

end;

dr(i+1) = dr(i);
cte(i+1) = cte(i);
s(i+1) = s(i);
ss(i+1) = ss(i);

% Plot this obstacle run:
figure(1); grid;
plot(t([1:i+1]),dr*180/pi,'r'); hold;
plot(t([1:i+1]),psi*180/pi,'b');
```

```
plot(t([1:i]),psioatot*180/pi,'g'); ylabel('Deg');
xlabel('Time');
title('Rudder-r (deg), Heading-b(deg), Psioatot-g(deg)');

figure(2)
axis equal;
plot(Y,X); title('Obstacle Run'); xlabel('X (m)'), ylabel('Y
(m)')
grid; hold
plot([Y_Way_c(1) PrevY_Way_c(1)],[X_Way_c(1)
PrevX_Way_c(1)],'r');
   for ii=2:No_tracks,
      plot([Y_Way_c(ii) Y_Way_c(ii-1)],[X_Way_c(ii) X_Way_c(ii-
          1)],'r');
   end;
% Plot Obstacles
plot(Yobs, Xobs,'gh');
hold;

figure(3); grid;
plot(t([1:i+1]),dr*180/pi,'r'); hold;
plot(t([1:i+1]),(psi-pi/2)*180/pi,'b');
plot(t([1:i]),psioatot*180/pi,'g'); ylabel('Deg');
xlabel('Time');
title('Rudder-r (deg), Heading-b(deg), Psioatot-g(deg)');
```

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF REFERENCES


Blidberg, Richard D., “The Development of Autonomous Underwater Vehicles (AUVs); A Brief Summary”, Autonomous Undersea Systems Institute, ICRA, Seoul, Korea, May 2001.

Borenstein, J. and Koren, Y., “The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots”, *IEEE Journal of Robotics and Automation* Vol 7, No 3, Jun 1991, pp. 278-288.

Concept of Operations, Organic Offboard Mine Reconnaissance, Director, Naval Expeditionary Warfare Division (N85) and Submarine Warfare Division (N87).

Gray, Keith W., “Obstacle Detection and Avoidance for an Autonomous Farm Tractor,” M.S. Utah State University, Logan, UT, 2000.

Gill, Mark A. C., “Obstacle Avoidance in Multi-Robot Systems”, World Scientific Publishing Company Pte. Ltd., 1998.

Healey, A.J., *Dynamics of Marine Vehicles (ME-4823)*, Class Notes, Naval Postgraduate School, Monterey, CA, 1995.

Healey, A. J., “Command and Control Demonstrations with Cooperating Vehicles,” *ONR Research Proposal in response to ONR BAA 01-012 “Demonstration of Undersea Autonomous Operation Capabilities and Related Technology Development”*, August 2001.

Healey, A. J., Kim, J., “Control And Random Searching With Multiple Robots”, *Proceedings IEEE CDC Conference 2000, Sydney Australia*, Nov. 2000 paper No INV-2303.

Healey, A. J., Marco, D. B., “Slow Speed Flight Control of Autonomous Underwater Vehicles: Experimental Results with NPS AUV II” *Proceedings of the 2nd International Offshore and Polar Engineering Conference, San Francisco*, July 14-19 1992.

Healey, A.J. and David Lienard, “Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles,” *IEEE Journal of Oceanic Engineering*, Vol. 18, pp 327-339, Jul 1993.

Hoerner, Sighard. *Fluid Dynamic Drag.* Published by author, 1965.

Johnson, Jay, “Parameter Identification of the ARIES AUV,” M.S. Thesis Naval Postgraduate School, Monterey, CA, June 2001.

Kamon, I. And Rivlin, E., “Sensory-based motion planning with global proofs”, *IEEE Transaction on Robotics and Automation*, Vol 13, no. 6, 1997.

Krogh, B. H. and Thorpe, C. E., “Integrated Path Planning and Dynamic Steering Control for Autonomous Vehicles”, *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, Apr, 1986, pp. 1664-1669.

Lane, David M., Petillot, Yvan, and Ruiz, Ioseba Tena, “Underwater Vehicle Obstacle Avoidance and Path Planning Using a Multi-Beam Forward Looking Sonar,” *IEEE Journal of Oceanic Engineering*, Vol. 26, Apr 2001.

Latrobe, J. C., “Robot Motion Planning”, Kluwer Academic Publishers, Norwell, MA, 1991.

Marco, D.B and A.J. Healey, “Command, Control and Navigation Experimental Results With the NPS ARIES AUV,” *IEEE Journal of Oceanic Engineering – Special Issue*, 2001.

Moitie, R. and Seube, N., “Guidance Algorithms for UUVs Obstacle Avoidance Systems”, OCEANS 2000, Brest, France.

Prestero, Timothy, “Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle,” M.S. Thesis Massachusetts Institute of Technology, Sep 2001.

Ruiz, Ioseba Joaquin Tena, “Enhanced Concurrent Mapping and Localization Using Forward-looking Sonar,” PHD Thesis Heriot-Watt University, Sep 2001.

Stentz, Anthony, “Optimal and Efficient Path Planning for Partially-Known Environments”, *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '94)*, Vol. 4, May, 1994, pp. 3310 - 3317.
Stokey, Roger and Von Alt, Chris, “REMUS Remote Environmental Measuring UnitS Operations and Maintenance Manual”, Woods Hole Oceanographic Institution, Apr 2001.

Thorpe, C. F., “Path Relaxation: Path Planning for a Mobile Robot”, *Carnegie-Mellon University Robotics Institute, Mobile Robots Laboratory, Autonomous Mobile Robots, Annual Report*, Sep 1986, pp. 12-19.

Warren, C.W. “A Technique for Autonomous Underwater Vehicle Route Planning”, IEEE Journal Of Ocean Engineering, Vol. 15, no. 3, pp. 199-204, 1990.

THIS PAGE INTENTIONALLY LEFT BLANK

# INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Mechanical Engineering Department Chairman, Code ME
   Naval Postgraduate School
   Monterey, CA

4. Naval/Mechanical Engineering Curriculum Code 34
   Naval Postgraduate School
   Monterey, CA

5. Professor Anthony J. Healey, Code ME/HY
   Department of Mechanical Engineering
   Naval Postgraduate School
   Monterey, CA

6. Dr. Donald Brutzman, Code UW/Br
   Undersea Warfare Group
   Naval Postgraduate School
   Monterey, CA

7. Dr. T. Swean, Code 32OE
   Office of Naval Research
   Arlington, VA

8. Doug Horner
   Naval Post Graduate School
   Monterey, CA

9. L. C. Van Alt
   WhoodsHole Oceanographic Institute
   Boston, MA

10. LT Lynn Fodrea
    Naval Surface Warfare Center, Dahlgren Division
    Dahlgren, VA