AUTOPILOT DESIGN AND GUIDANCE CONTROL OF ULISAR UUV (UNMANNED UNDERWATER VEHICLE)

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

KADİR ISIYEL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2007

Approval of the thesis:

AUTOPILOT DESIGN AND GUIDANCE CONTROL OF ULISAR UUV
(UNMANNED UNDERWATER VEHICLE)

submitted by **KADİR ISIYEL** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet Erkmen _____
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. M. Kemal Leblebicioğlu _____
Supervisor, **Electrical and Electronics Engineering Department, METU**

Examining Committee Memebers:

Prof. Dr. M. Kemal Özgören _____
Mechanical Engineering Department, METU

Prof. Dr. M. Kemal Leblebicioğlu _____
Electrical and Electronics Engineering Department, METU

Assoc. Prof. Halit Oğuztüzün _____
Computer Engineering Department, METU

Assist. Prof. Afşar Saranlı _____
Electrical and Electronics Engineering Department, METU

Assist. Prof. İlkay Ulusoy _____
Electrical and Electronics Engineering Department, METU

Date: 03.09.2007

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Kadir Isıyel

Signature :

ABSTRACT

AUTOPILOT DESIGN AND GUIDANCE CONTROL OF ULISAR UUV (UNMANNED UNDERWATER VEHICLE)

Isıyel, Kadir
M.Sc., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. M. Kemal Leblebicioğlu

September 2007, 109 pages

Unmanned Underwater Vehicles (UUV) in open-seas are highly nonlinear with system motions. Because of the complex interaction of the body with environment it is difficult to control them efficiently. Linearization is applied to system in order to design controllers developed for linear systems. To overcome the effects of disturbances, a mathematical model which will compensate all disturbances and effects of linearization is required. In this study first a mathematical model is formed wherein the linear and nonlinear hydrodynamic coefficients are calculated with strip theory.

After the basic mathematical model is developed, it is simplified and decoupled into speed, steering and diving subsystems. Consequently PID (Proportional Derivative Integral), SMC (Sliding Mode Control) and LQR (Linear Quadratic Regulator)/LQG (Linear Quadratic Gaussian) control methods can be applied on each subsystem to design controllers. Some of the system parameters can be estimated from state vector data based on measurements using the methods of linear sequential estimation and genetic algorithms. As for the final part of the study, an online obstacle avoidance algorithm which avoids local optimums using Boolean operators is presented. In addition a simple guidance algorithm is suggested for waypoint navigation.

Due to the fact that ULISAR UUV is still on construction phase, we were unable to test

our algorithms. But in the near future, we plan to study all these algorithms on the UUV ULISAR.

Keywords: Mathematical Modeling, Control, Parameter Estimation, Guidance

Öz

ULİSAR (ÇOK MAKSATLI ULUSAL İNSANSIZ SU ALTI ARACI) İNSANSIZ SU ALTI ARACININ OTOPİLOT TASARIMI VE GÜDÜM KONTROLU

Isıyel, Kadir
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. M. Kemal Leblebicioğlu

Eylül 2007, 109 sayfa

Açık deniz koşullarında bağlaşımlı hareketleriyle insansız su altı araçları yüksek seviyede doğrusal olmayan özellik gösterirler. Araç gövdesinin çevresi ile karmaşık etkileşimlerde bulunması, aracın etkin olarak denetimini oldukça güçleştirir. Bu noktada doğrusal sistemler için tasarlanan denetim yöntemlerinin kullanılabilmesi için sistem doğrusallaştırılır. Denetim açısından, çeşitli bozucu etkilerin üstesinden gelebilmek ve sistem üzerinde denetim sağlayabilmek için bu etkileri karşılayabilecek bir matematik model gerekmektedir. Bu çalışmada öncelikle bir matematiksel model oluşturulmuş ve modelde yer alan doğrusal ve doğrusal olmayan hidrodinamik katsayılar şerit teoremi ile hesaplanmıştır.

Matematiksel modelin elde edilmesinden sonra sistem; sürat, dönüş ve dalma alt sistemlerine ayrıştırılmış ve sisteme sırası ile PID, SMC ve LQR/LQG denetim yöntemleri uygulanmıştır. Daha sonra katsayı kestirimi yöntemi olarak doğrusal ardışık kestirim ve genetik algoritma yöntemleri uygulanmıştır. Çalışmanın bir parçası olarak, çevrimiçi seyirde kullanılmak üzere, Bool algoritmalarını kullanarak yerel minimum noktalarından kaçınan engelden sakınma algoritmaları denenmiştir. Son olarak, aracın belirlenen noktaları takip edebilmesi için temel bir güdüm algoritması oluşturulmuştur.

ULİSAR projesi hâlihazırda üretim aşamasında olduğundan dolayı oluşturulan algoritmalar

gerçek bir sistemde denenememiştir. Ancak yakın gelecekte bu tezde oluşturulan tüm algoritmaların gerçek sistem üzerinde denenmesi planlanmaktadır.

Anahtar Kelimeler: Matematiksel Modelleme, Denetim, Katsayı Kestirimi, Güdüm

To My Mother

With Whom Life Had Meant A Lot

ACKNOWLEDGMENTS

I would like to thank Prof. Dr. M. Kemal Leblebicioğlu for his guidance, patience, encouragements and understanding in every phase of my study. He became more than a superviser in every aspect. I would like to thank to Hüseyin Yiğitler for his invaluable advices and support by any means and to Emre Ege with other members of Robotics Laboratory. With his motivation and logistic support to our laboratory I would like to thank to Dr. Afşar Saranlı.

I would like to express my respect to Lieutenant İsmail Çalışkan for leading me in every way and both for sharing all his resources and his invaluable time. I can never forget his sacrifices in many ways.

Introducing the LaTeX 2_ε to me I would like to thank Tolga İnan and with other members of Computer Vision Laboratory to Erdem Akagündüz for their helps.

Giving permission to me for getting a graduate program and letting me to have the best national education opportunities available, I want to impart all my gratitude to my sequent commanders in Turkish Naval Forces.

Finally I would like to thank my parents for their perpetual support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation and Introduction to Underwater Vehicles

In this thesis, our goal is to simulate the control and guidance procedure of ULISAR unmanned underwater vehicle.

In this chapter a brief information about underwater vehicles, their applications and importance and lastly our objectives to be achieved for this thesis are mentioned.

Underwater vehicles are classified in two main groups as manned underwater vehicles (MUVs) and unmanned underwater vehicles (UUVs). Today, because of high operational costs, operator weariness and the painful experiences in history, which gave rise to the improvements in the UUVs, employment of the MUVs are highly limited. From operational aspects, UUVs are grouped in two main categories as remotely operated vehicles (ROVS) and autonomous underwater vehicles (AUVs). While ROVs give chance of intervention to the operator in any phase of operation, with their highly operational costs and their hulk values in case of lost, in recent years they have been disfavored. Nowadays, research on fully autonomous systems increased and lessened the necessity of a human operator. In the 1990s, about 30 new AUVs are built worldwide [37]. A self-contained, intelligent and self-decisive AUV is the goal for the current underwater vehicle research .

ULISAR is a TUBITAK (Türkiye Bilimsel ve Teknolojik Araştırma Kurumu) supported project . Being a small UUV compared with their coevals, ULISAR will be a novel ROV vehicle. ROVs are small, efficient and tethered vehicles to collect underwater data and fulfill given commands. Online communication with vehicle is achieved by generally fiber optic cable because of its variable huge bandwidths. Instead of utilizing cables, a few disadvan-

tages of which can be stated as drag in water, risk of disjunction and reduction in speed, ULISAR's communication will be maintained via acoustic link which will be satisfied by acoustic modems. Project will comprise two vehicles, one on the sea surface maintaining the RF (Radio Frequency) communication with control center and the other one in the sea, which will gather underwater information and achieve main task. The surface vehicle will relay the information it takes from bottom vehicle and vice versa.

For many years, ROVs proved their efficiency in many situations like underwater pipe inspections, rescuing goods from sunkens, oceanographic data collection and different mine counter-measure operations.

Constructing an UUV is an exhaustive and time-consuming job where most of the tests are executed in laboratory environment. Testing the parts in water environment is not always applicable and logical because of the risk of losing the valuable equipments and most important of all, risk of damage to the healths of project personnel. Therefore an effective and inexpensive choice to be implemented for simulation of system for tests. Simulation with computer aid is a practical and quick method of finding failures that may be confronted at sea. A good working simulator needs an actual model of the system, which will then imitate the outputs of the real system when the same inputs are applied. The coefficients of the system have to be accurately found in order to simulate the system efficiently otherwise simulation will fail and unpredicted situations may occur at real system tests [37].

In this study, we started the simulation first by forming the mathematical model. We generated the model forming the kinematic and rigid-body dynamics. Then we found the linear and nonlinear hydrodynamic coefficients by strip theory and boundary integral method as stated in [8] and [22]. Forming a mathematical model, exploiting the fact that our vehicle is not a fast varying system, it is linearized around an equilibrium point. Linearized system should be controlled more easily where most of the robust control methods are for linear systems. First we started control procedure with designing PID controller because of its simplicity and applicability to the most of the linear systems. Then we tried SMC, trusting the compensating success of it. Lastly we designed a controller using LQR/LQG methods. In last phase the benefit of separation principle is used, where first a regulator and then an observer is designed using Kalman Filter and they are put together to form a compensator for the plant [29].

For the efficiency of the simulation, unknown parameters of the model are important therefore for a good simulation we should have to acquire correct values of these parameters. Therefore we applied two parameter estimation algorithms. We started with linear sequen-

tial estimation method and tried to estimate the coefficients then we attempted to find them by running a genetic algorithm.

Underwater vehicles in real time operations need obstacle avoidance algorithms. Considering the needs, we worked on a problem such that, generated path would avoid the vehicle to stick in a local minimum, which may be a gap between two rocks where the vehicle can not pass successfully. Using the Boolean algorithms, the local minimum point is avoided [31]. Sometimes generated new path may be far from the optimum path but this is acceptable when the safety conditions are prior to any other issue.

Last of all, our vehicle needs a guidance system where a waypoint guidance system based on line of sight is preferred. The details about the guidance will be given on Chapter 5.

1.2 Literature on Control and Guidance of UUVs

The main factors that make control process difficult can be stated as: highly nonlinear, time-varying dynamic behavior of the vehicle, uncertainties in hydrodynamic coefficients, disturbances by sea environment (especially high frequency waves near surface), unpredicted underwater currents, for our case changes in the gravity and buoyancy. Considering the difficulty to fine-tune the control gains during operations, it will be advantageous to have a control system that will tune itself if the control performance decreases [37].

Different control techniques have been applied to underwater vehicles in recent years. Jalving used classical PID control methods for Norwegian Defence Research Establishment-AUV. He decoupled the system into three lightly interacting subsystems and designed three autopilots for steering, diving and speed control. The design of the each controller was based on PID techniques [12]. Yoerger and Slotine designed a sliding mode controller for an underwater vehicle. In their study they neglected cross-coupling terms and investigated the uncertainties of the hydrodynamic coefficients [36]. Meanwhile, preferring SMC for controlling their vehicle, Healey and Lienard were the ones who decoupled the system into three subsystems first time. Each autopilot was again designed using SMC with exploiting the advantage and ease of decoupled system [10]. Nakamura and Savant urged a nonlinear tracking control of an AUV pondering kinematic motion [37], [20]. They achieved the control by thinking the nonholonomic nature of the system without considering the dynamics of the system. Cristi, Papoulias and Healey designed a robust adaptive SMC such that in the presence of dynamical uncertainties, controllers can adjust to the changing dynamics and operating conditions [23]. A hybrid adaptive controller using both continuous and discrete

operations was mentioned by Tabaii et al [28].

In guidance of UUVs not so many studies have been performed. Healey et al. worked on the waypoint guidance by line of sight principle where the guidance is accomplished by a heading command to the vehicle's steering system to approach the line of sight between the present position of the vehicle and the waypoint to be reached. In missile guidance this is related to "proportional navigation" [10]. Caccia et al. introduced a PI- type task functions which enables a Lyapunov-based guidance system to compensate the effects of both unmodeled interactions between vehicle and environment [5].

1.3 Organization

The organization of the thesis is as follows:

- Chapter 1 mentions what is planned to achieve with this thesis and some studies done by other authors.
- Chapter 2 gives some mathematical formulation and transformations forming the mathematical model.
- Chapter 3 informs about the control methods used to control our vehicle. Comparison between the methods are also mentioned.
- Chapter 4 shows the efforts in estimating the linear hydrodynamic coefficients. Linear sequential estimation method and genetic algorithm are the methods used for estimation.
- Chapter 5 acquaints about guidance system for underwater vehicle and obstacle avoidance method.
- Chapter 6 gives a summary of the obtained results in this study. Then a discussion and possible future enhancements concluded in this the chapter.

CHAPTER 2

MATHEMATICAL MODELING

2.1 Introduction

In this chapter the equations of motion for our vehicle will be generated. First information about the body-fixed reference frame, linear and angular velocities, inertial reference frame positions and Euler rates will be given. Next, the vehicle kinematics which will be the relation of body-fixed velocities with inertial frame positions will be shown. Then the rigid body dynamics which is expanded from the Newton's second law will be derived. Lastly dynamics as the study of forces and moments of the moving objects will be investigated.

ULISAR is a small and modular UUV which brings a novel approach to underwater operations. She is comprised of the equipments that will carry out basic underwater operations successfully and fulfill the requirements of an underwater inspection. ULISAR will comprise an imaging sonar, two B/W cameras, lights, an acoustic modem to communicate with surface vehicle, acoustic transducers, pressure sensor, PC-104 stack and video grabber as main equipments. All power requirement will be satisfied by Lithium-Polymer battery packs. Her average speed is predicted to be about 1,5 knots. She will have stabilizers and fins to enhance the stability. Since she has no roll and sway control directly those pars will aid in satisfying passive roll control. Also in order to have passive roll stability, center of gravity must be below the center of buoyancy which will be performed by placing the heavy parts near the bottom of the vehicle. This method proved its success in many different designs [24]. She will be capable of diving to the depths of 100 meters but for the first tries 50 meters will be a fair depth.

General parts and main components of ULISAR are shown in Figure 2.1.

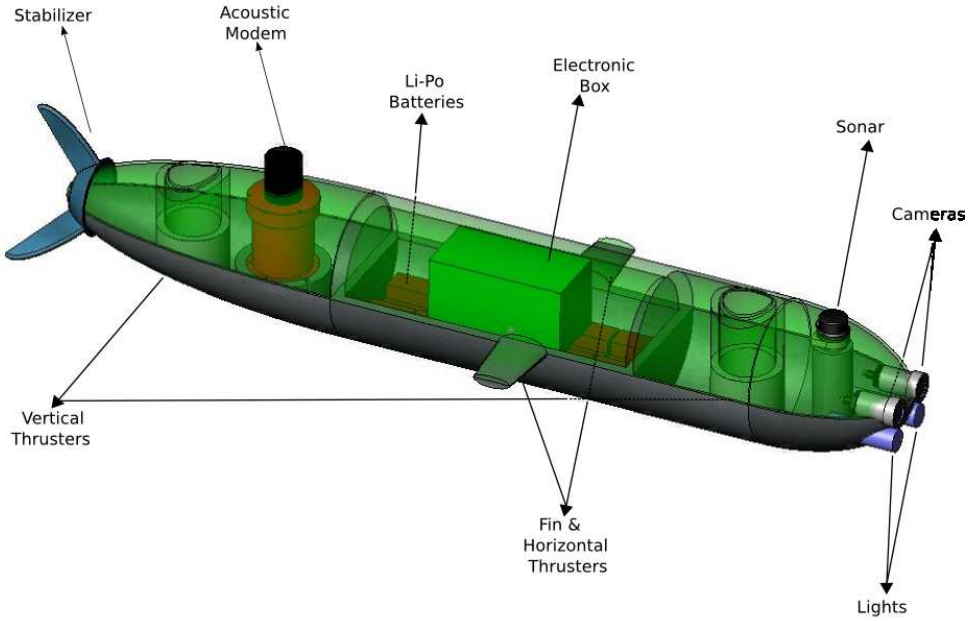

Figure 2.1: ULISAR UUV and main parts

The design of an underwater vehicle guidance and control systems requires knowledge of an extensive field of disciplines. Some of these are vectorial kinematics and dynamics, hydrodynamics, navigation systems and lastly control theory [8]. To able to design a high performance control system it is obvious that a good mathematical model of the vehicle is needed for both simulation and verification of the design.

First of all modeling of underwater vehicle is based on the study of statics and dynamics. Statics is the analysis of the forces and moments on physical systems in static equilibrium, while dynamics is concerned with the effects of forces on the motion of objects.

The motion of underwater vehicles is studied in 6 degrees of freedom (DOF) where 6 independent coordinates are necessary to determine the position and orientation of a rigid body. The first three coordinates and their time derivatives correspond to the position and translational motion along the x-, y-, and z- axes, whereas the last 3 coordinates and their time derivatives are used to describe orientation and rotational motion. For underwater vehicles these 6 degree of freedom are explained as:

-*surge* : motion in the x-direction

-*sway* : motion in the y-direction
-*heave* : motion in the z-direction
-*roll* : rotation about the x-axis
-*pitch* : rotation about the y-axis
-*yaw* : rotation about the z-axis

Table 2.1: Notation used for marine vehicles

DOF	Motions & Rotations	Forces & Moments	Linear & Angular Velocities	Positions & Euler Angles
1	Motions in the x-direction (surge)	X	u	x
2	Motions in the y-direction (sway)	Y	v	y
3	Motions in the z-direction (heave)	Z	w	z
4	Rotation in the x-axis (roll)	K	p	ϕ
5	Rotation in the y-axis (pitch)	M	q	θ
6	Rotation in the z-axis (yaw)	N	r	ψ

2.2 Kinematics

In this thesis we will use the following assumptions:

- Our vehicle is a rigid-body with a constant mass (Our vehicle's mass will change in time with proportional to the amount of water she will let in, but this amount is predicted to be small because of the slow velocity hence this mass change can be assumed to negligible.)
- Vehicle is not affected by the surface high frequency waves (operation condition is assumed to be deep waters).
- The effect of the rotating world to the accelerations of a point on the surface of the Earth is negligible (Indeed for slow vehicles this is a practical and tolerable assumption) [8].

- Hydrodynamic coefficients are not variable (Though stated in [13] nonlinear damping terms do not affect maneuverability of the underwater vehicles, changes in the speed and accelerations will differ the hydrodynamic coefficients. But since these coefficients are very small their changes will be much smaller where they can be assumed to negligible).
- We have the port-starboard (xz-plane) and bottom-top (xy-plane) symmetry.(Our heavy main parts are located on the middle of the xz-plane axis hence we have gained automatically a symmetry).

2.2.1 Coordinate Frames

Defining the motions of the underwater vehicles in 6-DOF, two coordinate reference frames are used. The moving coordinate frame $X_0Y_0Z_0$ is fixed to the vehicle and called the "Body-fixed reference frame" and other one according to the ground (earth) is called "Earth-fixed reference frame". Selecting the origin of the body-fixed coordinate frame as the *center of gravity (CG)* is a logical solution.

For underwater vehicles, body axes X_0, Y_0 and Z_0 coincide with the principal axes of inertia and are usually defined as [8]:

- X_0 - longitudinal axis (directed from aft to fore)
- Y_0 - transverse axis (directed to starboard)
- Z_0 - normal axis (directed from top to bottom)

Based on the The Society of Naval Architects and Marine Engineers (SNAME) notation, general motion of a vehicle in 6-DOF can be shown by the below vectors [8],

$$\eta = \left[\eta_1^T, \eta_2^T\right]^T; \qquad \eta_1 = [x,y,z]^T; \qquad \eta_2 = [\phi,\theta,\psi]^T \tag{2.1}$$

$$\nu = \left[\nu_1^T, \nu_2^T\right]^T; \qquad \nu_1 = [u,v,\omega]^T; \qquad \nu_2 = [p,q,r]^T \tag{2.2}$$

$$\tau = \left[\tau_1^T, \tau_2^T\right]^T; \qquad \tau_1 = [X,Y,Z]^T; \qquad \tau_2 = [K,M,N]^T \tag{2.3}$$

Above η denotes the position and orientation vector with coordinates in the earth-fixed frame, ν denotes the linear and angular velocity vector with coordinates in the body-fixed

coordinate frame and τ describes the forces and moments acting on the vehicle in the body-fixed frame. In a guidance and control system, orientation is usually represented by means of Euler angles or quaternions. Generally Euler angles are preferred for their simplicity but because tangent 90° is not defined for pitch angle, quaternions are used. In our case 90° pitch angle is an extreme case hence using Euler angles brings no disadvantage to us.

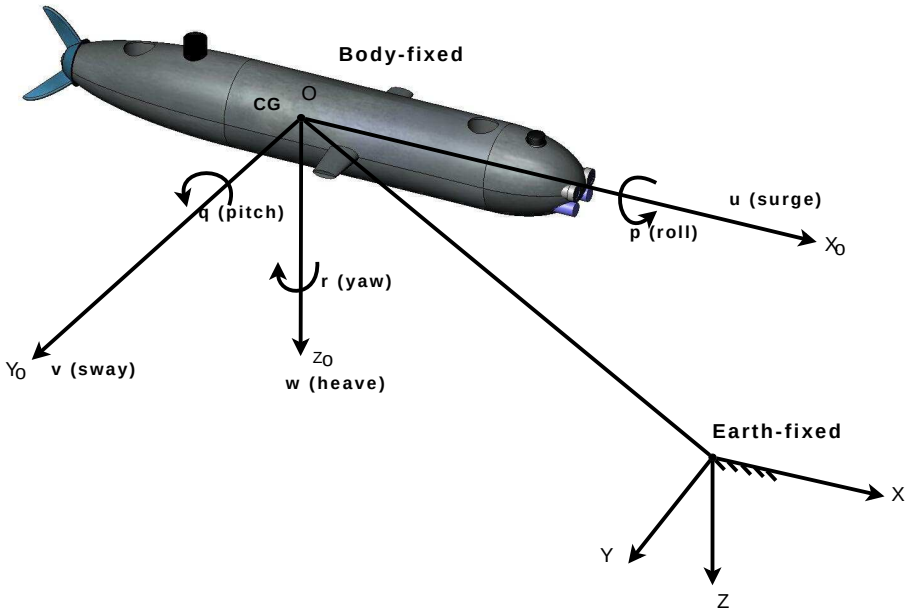

Figure 2.2: Earth-fixed and Body-fixed reference frames

All the motions of our vehicle in the body-fixed frame have to be represented relative to an inertial reference frame. For underwater vehicles we can assume that the effect of the rotating world to the accelerations of a point on the surface of the Earth is negligible. Therefore we do not need a star-fixed reference frame and we can select earth-fixed reference frame *XYZ* as inertial. In all our calculations, the position and the orientation of our vehicle should be explained according to the inertial reference frame where the linear and angular velocities should be expressed in the body-fixed reference frame.

2.2.2 Euler Angles

As mentioned above for transformation from body-fixed frame to earth-fixed frame and vice verse, Euler angles are used. In all our transformations *xyz- convention* will be used. First transforming translational motion, we will utilize the following equation:

$$\dot{\eta}_1 = T_1(\eta_2)\,\nu_1 \tag{2.4}$$

Writing above equation according to (2.1) and (2.2) we get

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = T_1(\eta_2) \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.5}$$

where T_1 in (2.5) is defined as [8]

$$T_1(\eta_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{2.6}$$

Rotational transformations are achieved by the body-fixed angular velocity vector $\nu_2 = [p, q, r]^T$ and Euler rate vector $\dot{\eta}_2 = \left[\dot{\phi}, \dot{\theta}, \dot{\psi}\right]^T$ related formula as

$$\dot{\eta}_2 = T_2(\eta_2)\,\nu_2 \tag{2.7}$$

Mentioning the vectors in open form we get

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = T_2(\eta_2) \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.8}$$

Angular velocity transformation matrix in (2.8) is defined as

$$T_2(\eta_2) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{2.9}$$

2.3 Rigid-Body Dynamics

In a general form the nonlinear dynamic equations of motion in 6 DOF can be written as:

$$M\dot{\nu} + C(\nu)\,\nu + D(\nu)\,\nu + \mathfrak{g}(\eta) = \tau \tag{2.10}$$

Using the Euler's first and second axioms which were built on Newton's second law we can write the 6 DOF Rigid body equations of motion as:

$$\begin{aligned}
X &= m\left[\dot{u} - vr + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(pr + \dot{q})\right] \\
Y &= m\left[\dot{v} - wp + ur - y_G(r^2 + p^2) + z_G(qr - \dot{p}) + x_G(qp + \dot{r})\right] \\
Z &= m\left[\dot{w} - uq + vp - z_G(p^2 + q^2) + x_G(rp - \dot{q}) + y_G(rq + \dot{p})\right] \\
K &= I_x\dot{p} + (I_z - I_y)\,qr - (\dot{r} + pq)I_{xz} + (r^2 - q2)I_{yz} + (pr - \dot{q})I_{xy} \\
&\quad + m\left[y_G(\dot{w} - uq + vp) - z_G(\dot{v} - wp + ur)\right] \\
M &= I_y\dot{q} + (I_x - I_z)\,pr - (\dot{p} + qr)I_{xy} + (p^2 - r2)I_{zx} + (qp - \dot{r})I_{yz} \\
&\quad + m\left[z_G(\dot{u} - vr + wq) - x_G(\dot{w} - uq + vp)\right] \\
N &= I_z\dot{r} + (I_y - I_x)\,pq - (\dot{q} + rp)I_{yz} + (q^2 - p2)I_{xy} + (rq - \dot{p})I_{zx} \\
&\quad + m\left[x_G(\dot{v} - wp + ur) - y_G(\dot{u} - vr + wq)\right]
\end{aligned} \tag{2.11}$$

In mathematical formulation for the ease of calculations (2.11) can be represented by vectorial form as

$$M_{RB}\dot{\nu} + C_{RB}(\nu)\nu = \tau_{RB} \tag{2.12}$$

Here the elements of the equations of motion relating to $\dot{\nu}$ can be written in M_{RB} as

$$M_{RB} = \begin{bmatrix}
m & 0 & 0 & 0 & mz_G & -my_G \\
0 & m & 0 & -mz_G & 0 & mx_G \\
0 & 0 & m & my_G & -mx_G & 0 \\
0 & -mz_G & my_G & I_x & -I_{xy} & -I_{zx} \\
mz_G & 0 & -mx_G & -I_{yx} & I_y & -I_{yz} \\
-my_G & mx_G & 0 & -I_{zx} & -I_{zy} & I_z
\end{bmatrix} \tag{2.13}$$

and the remainder elements the Coriolis term $\omega \times v$ and centripetal term $\omega \times (\omega \times r_G)$ can be written in C_{RB}. The Coriolis effect can be defined as the apparent deflection of objects from a straight path if the objects are viewed from a rotating frame of reference. The centripetal force is the external force required to make a body follow a circular path at constant speed. The force is directed inward, toward the center of the circle. Hence we can write our matrix as

$$C_{RB} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -m(y_G q + z_G r) & m(y_G p + w) & m(z_G p - v) \\ m(x_G q - w) & -m(z_G r + x_G p) & m(z_G q + u) \\ m(x_G r + v) & m(y_G r - u) & -m(x_G p + y_G q) \end{bmatrix}$$

$$\begin{bmatrix} m(y_G q + z_G r) & -m(x_G q - w) & -m(x_G r + v) \\ -m(y_G p + w) & m(z_G r + x_G p) & -m(y_G r - u) \\ -m(z_G p - v) & -m(z_G q + u) & m(x_G p + y_G q) \\ 0 & -I_{yz} q - I_{xz} p + I_z r & I_{yz} r + I_{xy} p - I_y q \\ I_{yz} q + I_{xz} p - I_z r & 0 & -I_{xz} r - I_{xy} q + I_x p \\ -I_{yz} r - I_{xy} p + I_y q & I_{xz} r + I_{xy} q - I_x p & 0 \end{bmatrix} \tag{2.14}$$

2.4 Added Mass

Added mass is the inertia added to system because of the accelerating body which will move some liquid surrounding its body. But the vehicle will force the surrounding fluid with proportional to forced harmonic motion due to acceleration of body where the particles which are far from the body will be induced less. In order to generate the added mass forces and moments Kirchhoff's equations related to the fluid kinetic energy will be used.

Kinetic energy of the ideal fluid can be written as

$$T_A = \frac{1}{2} \nu^T M_A \nu \tag{2.15}$$

where M_A is the 6×6 added mass inertia matrix which is comprised of 36 distinct added mass coefficients.

Hence we can write the added mass inertia matrix as

$$M_A \cong - \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \tag{2.16}$$

In the added mass inertia matrix the elements of the matrix are the derivatives of the forces and moments in the stated axis with respect to the accelerations. In other words,

$$M_{\dot{w}} = \frac{\partial M}{\partial \dot{w}} \tag{2.17}$$

Expressing the body-fixed velocity vectors as $\nu_1 = [u, v, \omega]^T$ and $\nu_2 = [p, q, r]^T$, relation of the force τ_1 and moment τ_2 is achieved with Kirchhoff's equations in vector form.

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \nu_1}\right) + \nu_2 \times \frac{\partial T}{\partial \nu_1} = \tau_1 \tag{2.18}$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \nu_2}\right) + \nu_2 \times \frac{\partial T}{\partial \nu_2} + \nu_1 \times \frac{\partial T}{\partial \nu_1} = \tau_2 \tag{2.19}$$

Hence for the totally submerged vehicle we will find the added mass terms by using Kirchhoff's equations. Here we will utilize the fluid kinetic energy principle and take into consideration that by the motion of the vehicle in any direction, it will bring forth a kinetic energy for surrounding fluid [8]. Expanding the equations (2.18) and (2.19) yields

$$\frac{d}{dt}\begin{bmatrix} \frac{\partial T_A}{\partial u} \\ \frac{\partial T_A}{\partial v} \\ \frac{\partial T_A}{\partial \omega} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} \frac{\partial T_A}{\partial u} \\ \frac{\partial T_A}{\partial v} \\ \frac{\partial T_A}{\partial \omega} \end{bmatrix} = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} \quad \Rightarrow \quad \frac{d}{dt}\begin{bmatrix} \frac{\partial T_A}{\partial u} \\ \frac{\partial T_A}{\partial v} \\ \frac{\partial T_A}{\partial \omega} \end{bmatrix} + \begin{bmatrix} q\frac{\partial T_A}{\partial \omega} - r\frac{\partial T_A}{\partial v} \\ r\frac{\partial T_A}{\partial u} - p\frac{\partial T_A}{\partial \omega} \\ p\frac{\partial T_A}{\partial v} - q\frac{\partial T_A}{\partial u} \end{bmatrix} = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} \tag{2.20}$$

For the moments from the added mass,

$$\frac{d}{dt}\begin{bmatrix} \frac{\partial T_A}{\partial p} \\ \frac{\partial T_A}{\partial q} \\ \frac{\partial T_A}{\partial r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} \frac{\partial T_A}{\partial p} \\ \frac{\partial T_A}{\partial q} \\ \frac{\partial T_A}{\partial r} \end{bmatrix} + \begin{bmatrix} u \\ v \\ \omega \end{bmatrix} \times \begin{bmatrix} \frac{\partial T_A}{\partial u} \\ \frac{\partial T_A}{\partial v} \\ \frac{\partial T_A}{\partial \omega} \end{bmatrix} = \begin{bmatrix} K_A \\ M_A \\ N_A \end{bmatrix} \tag{2.21}$$

Applying vectorial products in (2.21) gives

$$\frac{d}{dt}\begin{bmatrix} q\frac{\partial T_A}{\partial r} - r\frac{\partial T_A}{\partial q} \\ r\frac{\partial T_A}{\partial p} - p\frac{\partial T_A}{\partial r} \\ p\frac{\partial T_A}{\partial q} - q\frac{\partial T_A}{\partial p} \end{bmatrix} + \begin{bmatrix} v\frac{\partial T_A}{\partial \omega} - \omega\frac{\partial T_A}{\partial v} \\ \omega\frac{\partial T_A}{\partial u} - u\frac{\partial T_A}{\partial \omega} \\ u\frac{\partial T_A}{\partial v} - v\frac{\partial T_A}{\partial u} \end{bmatrix} = \begin{bmatrix} K_A \\ M_A \\ N_A \end{bmatrix} \tag{2.22}$$

When we take the partial derivatives of T_A with respect to our linear and angular velocity

vectors and substitute them in (2.20) and (2.22) yields us the main added mass terms [8].

$$
\begin{aligned}
X_A &= X_{\dot{u}}\dot{u} + X_{\dot{\omega}}(\dot{\omega} + uq) + X_{\dot{q}}\dot{q} + Z_{\dot{\omega}}\omega q + Z_{\dot{q}}q^2 \\
&\quad + X_{\dot{v}}v + X_{\dot{p}}\dot{p} + X_{\dot{r}}\dot{r} - Y_{\dot{v}}vr - Y_{\dot{p}}pr - Y_{\dot{r}}r^2 \\
&\quad - X_{\dot{v}}ur - Y_{\dot{\omega}}\omega r \\
&\quad + Y_{\dot{\omega}}vq + Z_{\dot{p}}pq - (Y_{\dot{q}} - Z_{\dot{r}})qr \\
Y_A &= X_{\dot{v}}\dot{u} + Y_{\dot{\omega}}\dot{\omega} + Y_{\dot{q}}\dot{q} \\
&\quad + Y_{\dot{v}}\dot{v} + Y_{\dot{p}}\dot{p} + Y_{\dot{r}}\dot{r} + X_{\dot{v}}rv - Y_{\dot{\omega}}vp + X_{\dot{r}}r^2 + (X_{\dot{p}} - Z_{\dot{r}})rp - Z_{\dot{p}}p^2 \\
&\quad - X_{\dot{\omega}}(up - \omega r) + X_{\dot{u}}ur - Z_{\dot{\omega}}\omega p \\
&\quad - Z_{\dot{q}}pq + X_{\dot{q}}qr \\
Z_A &= X_{\dot{\omega}}(\dot{u} - \omega q) + Z_{\dot{\omega}}\dot{\omega} + Z_{\dot{q}}\dot{q} - X_{\dot{u}}uq - X_{\dot{q}}q^2 \\
&\quad + Y_{\dot{\omega}}\dot{v} + Z_{\dot{p}}\dot{p} + Z_{\dot{r}}\dot{r} + Y_{\dot{v}}vp + Y_{\dot{r}}rp + Y_{\dot{p}}p^2 \\
&\quad + X_{\dot{v}}up + Y_{\dot{\omega}}\omega p \\
&\quad - X_{\dot{v}}vq - (X_{\dot{p}} - Y_{\dot{q}})pq - X_{\dot{r}}qr \\
K_A &= X_{\dot{p}}\dot{u} + Z_{\dot{p}}\dot{\omega} + K_{\dot{q}}\dot{q} - X_{\dot{v}}\omega u + X_{\dot{r}}uq - Y_{\dot{\omega}}\omega^2 - (Y_{\dot{q}} - Z_{\dot{r}})\omega q + M_{\dot{r}}q^2 \\
&\quad + Y_{\dot{p}}\dot{v} + K_{\dot{p}}\dot{p} + K_{\dot{r}}\dot{r} + Y_{\dot{\omega}}v^2 - (Y_{\dot{q}} - Z_{\dot{r}})vr + Z_{\dot{p}}vp - M_{\dot{r}}r^2 - K_{\dot{q}}rp \\
&\quad + X_{\dot{\omega}}uv - (Y_{\dot{v}} - Z_{\dot{\omega}})v\omega - (Y_{\dot{r}} + Z_{\dot{q}})\omega r - Y_{\dot{p}}\omega p - X_{\dot{q}}ur \\
&\quad + (Y_{\dot{r}} + Z_{\dot{q}})vq + K_{\dot{r}}pq - (M_{\dot{q}} - N_{\dot{r}})qr \\
M_A &= X_{\dot{q}}(\dot{u} + \omega q) + Z_{\dot{q}}(\dot{\omega} - uq) + M_{\dot{q}}\dot{q} - X_{\dot{\omega}}\left(u^2 - \omega^2\right) - (Z_{\dot{\omega}} - X_{\dot{u}})\omega u \\
&\quad + Y_{\dot{q}}\dot{v} + K_{\dot{q}}\dot{p} + M_{\dot{r}}\dot{r} + Y_{\dot{p}}vr - Y_{\dot{r}}vp - K_{\dot{r}}\left(p^2 - r^2\right) + (K_{\dot{p}} - N_{\dot{r}})rp \\
&\quad - Y_{\dot{\omega}}uv + X_{\dot{v}}v\omega - (X_{\dot{r}} + Z_{\dot{p}})(up - \omega r) + (X_{\dot{p}} - Z_{\dot{r}})(\omega p + ur) \\
&\quad - M_{\dot{r}}pq + K_{\dot{q}}qr \\
N_A &= X_{\dot{r}}\dot{u} + Z_{\dot{r}}\dot{\omega} + M_{\dot{r}}\dot{q} + X_{\dot{v}}u^2 + Y_{\dot{\omega}}\omega u - (X_{\dot{p}} - Y_{\dot{q}})uq - Z_{\dot{p}}\omega q - K_{\dot{q}}q^2 \\
&\quad + Y_{\dot{r}}\dot{v} + K_{\dot{r}}\dot{p} + N_{\dot{r}}\dot{r} - X_{\dot{v}}v^2 - X_{\dot{r}}vr - (X_{\dot{p}} - Y_{\dot{q}})vp + M_{\dot{r}}rp + K_{\dot{q}}p^2 \\
&\quad - (X_{\dot{u}} - Y_{\dot{v}})uv - X_{\dot{\omega}}v\omega + (X_{\dot{q}} + Y_{\dot{p}})up + Y_{\dot{r}}ur + Z_{\dot{q}}\omega p \\
&\quad - (X_{\dot{q}} + Y_{\dot{p}})vq - (K_{\dot{p}} - M_{\dot{q}})pq - K_{\dot{r}}qr
\end{aligned}
\tag{2.23}
$$

2.5 Damping

Hydrodynamic damping for the underwater vehicles occurs because of the following effects.

- Potential Damping: Damping caused by the surface waves. These waves are generally

high frequency waves with small wave lengths. By our assumption at the beginning of the mathematical model stating that our vehicle works near the sea bottom gives us the right to neglect this effect.

- Skin Friction: This is the damping occurring because of the flow of water around the boundary of vehicle. While vehicle advancing with a constant speed, water near the bow achieves laminar flow (streamline flow) with no disruption to the surface. Going forward on its flow after passing the bow, skin friction decelerates the liquid that is why turbulent flow starts at this point. This process of passing from laminar flow to turbulent flow is known as boundary layer transition. Skin friction is represented with linear skin friction because of laminar boundary layer and quadratic skin friction due to turbulent boundary layers. Non-dimensional Reynolds number assigns the type of the flow.

- Wave Damping: This is the damping due to the waves while vehicles try to advance on the surface of the water. Again with our assumption that the operations will be near the sea bottom, we can neglect this damping.

- Damping of Vortex Shedding: Vortex shedding occurs because of the pressure differences on the flow path of water. Liquid after passing the first meet surface of the object creates the low pressure vortices, which ends with a turbulent and unsteady flow. The size of the vortices and the effect of damping due to vortex shedding is directly proportional to front (projected) sectional area of the vehicle and with square of velocity. Trying to increase the operational speed of underwater vehicle brings damping disadvantage with it. At this point, the outer body design and the vehicle's production material take an important role.

From the aspect of losses, effect of the damping will mostly occur due to skin friction and vortex shedding. Skin friction is an important effect on the damping of the vehicle but the details of it is far beyond the scope of this thesis. More details about damping can be found in [15] and [22]. The design of the body to decrease the damping is an important problem for the construction of vehicle. Here eccentricity plays an important role, which shows how much the head shape of vehicle deviates from circular mode. Also the ratio of the total length of the vehicle to the diameter directly affects the speed of the vehicle where the torpedoes are good examples for this kind of design. Now let us formulate the components of the damping which are important for us.

The damping force due to the vortex shedding can be modeled as

$$D_f(U) = -\frac{1}{2}\rho\, c_d\,(Rn)\, A_{cs} \mid U \mid U \tag{2.24}$$

where c_d is the non-dimensional drag coefficient directly related with Reynolds number. A_{cs} is the projected cross sectional area of the vehicle facing with water which is $\pi d^2/4$ for a sphere (d is diameter). U stands for the velocity of the vehicle and ρ is the density of the water. Reynolds number is a function of velocity (U), physical length (l) and kinematic viscosity (ν). Reynolds number can be calculated by the following formula [22]. In our calculations we assumed $c_d = 0.20$.

$$R_n = \frac{\rho\, U\, l}{\mu} = \frac{U\, l}{\nu} \tag{2.25}$$

Here μ stands for the fluid viscosity and ρ for the density of water ($\nu = 1.05 \times 10^{-6}$ for sea water with 20°C and salinity of 3.5 %). Appendix B shows the drag coefficient of a sphere for different Reynolds numbers [22].

Damping due to the skin friction will be modeled as linear and quadratic damping. Hence our damping will be as

$$D(\nu)\,\nu + |\nu| D(\nu)\,\nu \tag{2.26}$$

Though some approximations and simplifications will be achieved on the damping matrix in following steps, now it can be written as

$$D_M(\nu) = \begin{bmatrix} X_u + X_{|u|u}|u| & X_v + X_{|v|v}|v| & X_w + X_{|w|w}|w| \\ Y_u + Y_{|u|u}|u| & Y_v + Y_{|v|v}|v| & Y_w + Y_{|w|w}|w| \\ Z_u + Z_{|u|u}|u| & Z_v + Z_{|v|v}|v| & Z_w + Z_{|w|w}|w| \\ K_u + K_{|u|u}|u| & K_v + K_{|v|v}|v| & K_w + K_{|w|w}|w| \\ M_u + M_{|u|u}|u| & M_v + M_{|v|v}|v| & M_w + M_{|w|w}|w| \\ N_u + N_{|u|u}|u| & N_v + N_{|v|v}|v| & N_w + N_{|w|w}|w| \end{bmatrix}$$

$$\begin{bmatrix} X_p + X_{|p|p}|p| & X_q + X_{|q|q}|q| & X_r + X_{|r|r}|r| \\ Y_p + Y_{|p|p}|p| & Y_q + Y_{|q|q}|q| & Y_r + Y_{|r|r}|r| \\ Z_p + Z_{|p|p}|p| & Z_q + Z_{|q|q}|q| & Z_r + Z_{|r|r}|r| \\ K_p + K_{|p|p}|p| & K_q + K_{|q|q}|q| & K_r + K_{|r|r}|r| \\ M_p + M_{|p|p}|p| & M_q + M_{|q|q}|q| & M_r + M_{|r|r}|r| \\ N_p + N_{|p|p}|p| & N_q + N_{|q|q}|q| & N_r + N_{|r|r}|r| \end{bmatrix} \tag{2.27}$$

2.6 Gravitational and Buoyant Forces

In our vehicle center of gravity will be defined with $r_G = [x_G, y_G, z_G]^T$ and the center of buoyancy will be expressed with $r_B = [x_B, y_B, z_B]^T$. The gravitational force f_G will act on center of gravity and buoyant force will act on center of buoyancy f_B where both forces act in inertial frame but they are defined in body-fixed frame.

The mass of the vehicle is defined as m, V as the volume of fluid displaced, g as the acceleration of gravity downwards, ρ density of the fluid. Weight of vehicle and buoyancy force can be written as $W = mg$, $B = \rho g V$.

Then the gravitational and buoyant forces in body-fixed frame can be defined by using Euler transformations

$$f_G = T_1^{-1}(\eta_2) \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix} \tag{2.28}$$

and

$$f_B = -T_1^{-1}(\eta_2) \begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix} \tag{2.29}$$

Finally gravitational and buoyant forces and moments can be written as [8]

$$\mathfrak{g} = -\begin{bmatrix} f_G(\eta) + f_B(\eta) \\ r_G \times f_G(\eta) + r_B \times f_B(\eta) \end{bmatrix} \tag{2.30}$$

Substituting the center of gravity and buoyancy with forces in (2.30) we get

$$\mathfrak{g} = \begin{bmatrix} (W-B)s\theta \\ -(W-B)c\theta s\phi \\ -(W-B)c\theta c\phi \\ -(y_G W - y_B B)c\theta c\phi + (z_G W - z_B B)c\theta s\phi \\ (z_G W - z_B B)s\theta + (x_G W - x_B B)c\theta c\phi \\ -(x_G W - x_B B)c\theta s\phi - (y_G W - y_B B)s\theta \end{bmatrix} \tag{2.31}$$

To that point we showed the path to generate the mathematical model of an underwater vehicle. But because of the nonlinear and coupled attitude of the model there will be too many unknowns with different weighted values which is an undesired point. Now we will

use some simplifications and assumptions to reduce the complicated model which proved success in many designed underwater vehicles.

Because of the position of the center of gravity and center of buoyancy, x_G and y_G will be equal to zero. On the other hand, since $I_{xy} = 0$ (top/bottom symmetry) and $I_{xz} = 0$ (port/starboard symmetry) our new inertia matrix becomes

$$I_0 \cong \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & I_{yz} \\ 0 & I_{zy} & I_z \end{bmatrix} \tag{2.32}$$

Also because of the symmetry properties, our mass inertia matrix can be simplified. The simplification procedure applied to mass matrix can also be applied to damping matrix [8]. In Section 2.2 we assumed that we have xz- and xy- plane symmetries (port/starboard and bottom/top symmetries) by which we acquire the following simplified mass matrix

$$M = \begin{bmatrix} M_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & M_{22} & 0 & 0 & 0 & M_{26} \\ 0 & 0 & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & M_{53} & 0 & M_{55} & 0 \\ 0 & M_{62} & 0 & 0 & 0 & M_{66} \end{bmatrix} \tag{2.33}$$

The same simplification method can be applied to the damping matrix and the same coefficients on the stated positions will be left but in most of the underwater applications a rough approximation is done where the damping matrix with its linear and quadratic terms is assumed to form a diagonal matrix.

2.7 Hydrodynamic Coefficients

In this section we will find the numerical values of hydrodynamical coefficients. We will derive the added mass coefficients and damping coefficients which are the representations of the derivation of forces and moments with respect to the linear and angular velocities and accelerations. As we evinced in Section 2.5 axial drag can be calculated by

$$X_{|u|u} = -\frac{1}{2}\rho\, c_d\,(Rn)\, A_{cs}\,|U|U \tag{2.34}$$

The remaining crossflow drag will be found by strip theory according to [8]. After simplification in the damping matrix the following coefficients remain, which are found by the

following formula: $Y_{|v|v}$, $Z_{|w|w}$, $K_{|p|p}$, $M_{|q|q}$, $N_{|r|r}$, $Y_{|r|r}$, $Z_{|q|q}$, $M_{|w|w}$, $N_{|v|v}$.

$$Y_{|v|v} = Z_{|w|w} = -\frac{1}{2}\rho\, c_{dc} \int_{-L/2}^{L/2} 2\,b(x)\,dx \tag{2.35}$$

$$M_{|w|w} = -N_{|v|v} = \frac{1}{2}\rho\, c_{dc} \int_{-L/2}^{L/2} 2\,b(x)\,x\,dx \tag{2.36}$$

$$Y_{|r|r} = -Z_{|q|q} = -\frac{1}{2}\rho\, c_{dc} \int_{-L/2}^{L/2} 2\,b(x)\,x\,|x|\,dx \tag{2.37}$$

$$M_{|q|q} = -N_{|r|r} = -\frac{1}{2}\rho\, c_{dc} \int_{-L/2}^{L/2} 2\,b(x)\,x^3\,dx \tag{2.38}$$

$$K_{|p|p} = 0 \tag{2.39}$$

Here ρ is the water density, c_{dc} crossflow drag coefficient, $b(x)$ is the half-width of the vehicle with respect to the total length, l is the length of the vehicle.

After simplifications because of symmetry our added mass matrix (2.16) will transform into

$$M = \begin{bmatrix} X_{\dot{u}} & 0 & 0 & & 0 & 0 \\ 0 & Y_{\dot{v}} & 0 & 0 & 0 & N_{\dot{v}} \\ 0 & 0 & Z_{\dot{w}} & 0 & M_{\dot{w}} & 0 \\ 0 & 0 & 0 & K_{\dot{p}} & 0 & 0 \\ 0 & 0 & Z_{\dot{q}} & 0 & M_{\dot{q}} & 0 \\ 0 & Y_{\dot{r}} & 0 & 0 & 0 & N_{\dot{r}} \end{bmatrix} \tag{2.40}$$

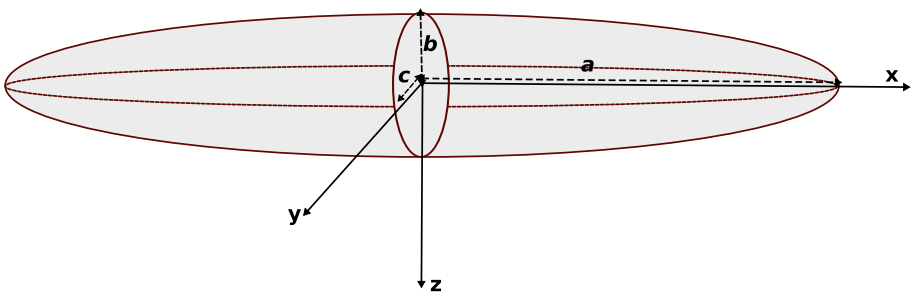

Figure 2.3: Prolate Ellipsoid and Dimensions

Hence we have to derive the remainders form the added mass matrix terms which are; $X_{\dot{u}}$, $Y_{\dot{v}}$, $Z_{\dot{w}}$, $K_{\dot{p}}$, $M_{\dot{q}}$, $N_{\dot{r}}$, $Y_{\dot{r}}$, $Z_{\dot{q}}$, $M_{\dot{w}}$, $N_{\dot{v}}$. For slender bodies these coefficients can be derived by strip theory. These coefficients in three-dimensions are found by integrating the two-dimensional coefficients along the vehicle length. In our case our vehicle shows similarities with a prolate ellipsoid as shown in Figure (2.3).

Therefore using the strip theory added mass coefficients will be found according to [8] as

$$-X_{\dot{u}} = \int_{-L/2}^{L/2} A_{11}(y,z)dx \quad \simeq \quad 0.10m \tag{2.41}$$

$$-Y_{\dot{v}} = \int_{-L/2}^{L/2} A_{22}(y,z)dx \tag{2.42}$$

$$-Z_{\dot{w}} = \int_{-L/2}^{L/2} A_{33}(y,z)dx \tag{2.43}$$

$$-K_{\dot{p}} = \int_{-L/2}^{L/2} A_{44}(y,z)dx \quad \triangleq \quad \int_{-B/2}^{B/2} y^2 A_{33}(x,z)\,dy + \int_{-H/2}^{H/2} z^2 A_{22}(x,y)\,dz \tag{2.44}$$

$$-M_{\dot{q}} = \int_{-L/2}^{L/2} A_{55}(y,z)dx \quad \triangleq \quad \int_{-L/2}^{L/2} x^2 A_{33}(x,z)\,dx + \int_{-H/2}^{H/2} z^2 A_{11}(x,y)\,dz \tag{2.45}$$

$$-N_{\dot{p}} = \int_{-L/2}^{L/2} A_{66}(y,z)dx \quad \triangleq \quad \int_{-B/2}^{B/2} y^2 A_{11}(x,z)\,dy + \int_{-L/2}^{L/2} x^2 A_{22}(y,z)\,dx \tag{2.46}$$

where L,B and H are the dimensions of the vehicle. For a prolate ellipsoid, 2-dimensional coefficients in the above equations are given in Figure (2.4).

On the other hand those added mass coefficients can be found by theoretical formulations stated by Lamb in [15].

$$X_{\dot{u}} = -\frac{\alpha_0}{2-\alpha_0}\,m \tag{2.47}$$

$$Y_{\dot{v}} = \quad Z_{\dot{w}} \quad = -\frac{\beta_0}{2-\beta_0}\,m \tag{2.48}$$

$$K_{\dot{p}} = \quad 0 \tag{2.49}$$

$$N_{\dot{r}} = \quad M_{\dot{q}} \quad = -\frac{1}{5}\,\frac{(b^2-a^2)^2(\alpha_0-\beta_0)}{2\,(b^2-a^2)+(b^2+a^2)(\beta_0-\alpha_0)}\,m \tag{2.50}$$

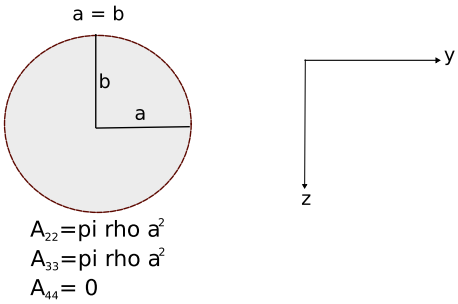

Figure 2.4: Two-dimensional Added Mass Coefficients

Here m is the mass of the vehicle which can be found by

$$m = \frac{4}{3}\pi\rho\, a\, b^2 \tag{2.51}$$

and α_0 and β_0 are defined as

$$\alpha_0 = \frac{2(1-e^2)}{e^3}\left(\frac{1}{2}ln\frac{1+e}{1-e} - e\right) \tag{2.52}$$

$$\beta_0 = \frac{1}{e^2} - \frac{1-e^2}{2e^3}ln\frac{1+e}{1-e} \tag{2.53}$$

Above in the equations e stands for the eccentricity defined as

$$e = 1-(b/a)^2 \tag{2.54}$$

Also there exists another alternative method of equations which are related with the added mass terms. We used this last method to check the coefficients found by the first explained method above, both gave the same results after calculations. This last method is similar with the one method mentioned above. In this method, Lamb expresses first k-terms as

$$k_1 = \frac{\alpha_0}{2-\alpha_0} \tag{2.55}$$

$$k_2 = \frac{\beta_0}{2-\beta_0} \tag{2.56}$$

$$k' = \frac{e^4(\beta_0-\alpha_0)}{(2-e^2)\,[2e^2-(2-e^2)(\beta_0-\alpha_0)]} \tag{2.57}$$

Then he mentioned the added mass coefficients as

$$X_{\dot{u}} = -k_1\, m \tag{2.58}$$

$$Y_{\dot{v}} = -Z_{\dot{w}} \ = -k_1\, m \tag{2.59}$$

$$N_{\dot{r}} = -M_{\dot{q}} \ = -k'\, I_y \tag{2.60}$$

where the moment of inertia in y- axis, I_y can be found by

$$I_y = I_z = \frac{4}{15}\pi\,\rho\, a\, b^2\,(a^2+b^2) \tag{2.61}$$

For a prolate ellipsoid calculation of the quadratic damping coefficients can be achieved by the equations (2.35)-(2.39) but since shape of our vehicle is a little different from an ellipsoid we have to separate it in order to find stated coefficients.

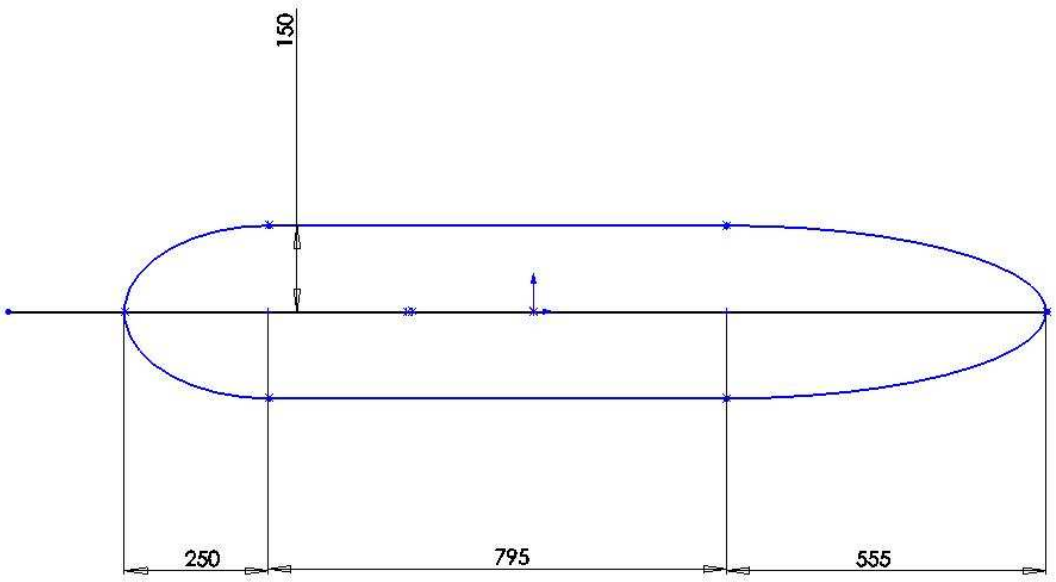

Figure 2.5: Cross-sectional view of our Vehicle

It is clear in Figure (2.5) that we have to separate the ellipsoid into three parts. We will find the equations of each part. Our vehicle's length is 1.60 meters and the distance of the

separation points from nose to aft are .25 m. and 1.045 m. The width of hull with respect to total length of vehicle is required in the quadratic damping equations where we formulated the each section of the vehicle as

$$y = \sqrt{0.15^2 - \left(\frac{0.15}{0.25}\right)^2 (x-0.25)^2} \qquad 0 \le x \le 0.25 \qquad (L_1) \qquad (2.62)$$

$$y = 0.15 \qquad 0.25 < x < 1.045 \qquad (L_2) \qquad (2.63)$$

$$y = \sqrt{0.15^2 - \left(\frac{0.15}{0.555}\right)^2 (x-1.045)^2} \qquad 1.045 \le x \le 1.60 \qquad (L_3) \qquad (2.64)$$

here y shows the half width of the ellipsoid with respect to the length which is defined as $b(x)$ in equations (2.35)-(2.39).

Hence our quadratic damping equations will be as

$$Y_{|v|v} = Z_{|w|w} = -\frac{1}{2}\rho\, c_{dc} \int_0^{P1} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.25}\right)^2 (x-0.25)^2}\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P1}^{P2} 0.30\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P2}^{P3} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.555}\right)^2 (x-1.045)^2}\, dx \qquad (2.65)$$

$$M_{|w|w} = -N_{|v|v} = -\frac{1}{2}\rho\, c_{dc} \int_0^{P1} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.25}\right)^2 (x-0.25)^2}\, x\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P1}^{P2} 0.30\, x\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P2}^{P3} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.555}\right)^2 (x-1.045)^2}\, x\, dx \qquad (2.66)$$

$$Y_{|r|r} = -Z_{|q|q} = -\frac{1}{2}\rho\, c_{dc} \int_0^{P1} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.25}\right)^2 (x-0.25)^2}\, x|x|\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P1}^{P2} 0.30\, x|x|\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P2}^{P3} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.555}\right)^2 (x-1.045)^2}\, x|x|\, dx \qquad (2.67)$$

$$M_{|q|q} = -N_{|r|r} = -\frac{1}{2}\rho\, c_{dc} \int_0^{P1} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.25}\right)^2 (x-0.25)^2}\, x^3\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P1}^{P2} 0.30\, x^3\, dx - \frac{1}{2}\rho\, c_{dc} \int_{P2}^{P3} 2\sqrt{0.15^2 - \left(\frac{0.15}{0.555}\right)^2 (x-1.045)^2}\, x^3\, dx \qquad (2.68)$$

$$K_{|p|p} = 0 \tag{2.69}$$

2.8 Retrieval of Hydrodynamic Coefficients

In this section, we will show the results of our calculations finding the main hydrodynamic coefficients by using formula in Section 2.7. Through the study it is assumed that the hydrodynamic coefficients are time-invariant. Although it is not the case in real time such an assumption will not effect the system so much because the coefficients are very small compared to mass and states, slight changes in them will be compensated by the system.

In the calculations of the coefficients the following values are taken.

Table 2.2: Vehicle Related Values Used in Coefficient Retrieval

Mass (m)	64 kg	Total Length (L)	1.60 m
Half Length (a)	0.80 m	Half Width (b,c)	0.15 m
Vehicle Avg. Density (ρ_v)	996 kg/m^3	Water Density (ρ_v)	1023 kg/m^3
Axial Drag Coef.	0.1412	Crossflow Drag Coef.	2.1
Center of Gravity, x- (x_G)	0	Center of Gravity, y-(y_G)	0
Center of Gravity, z- (z_G)	0.08 m	Center of Buoyancy, x- (y_B)	0
Center of Buoyancy,y- (y_B)	0	Center of Buoyancy, z- (z_B)	0
Axial Projected Area (A_f)	0.0707 m^2	Eccentricity	0.9648
Added Mass Coef. 1 (α_0)	0.1610	Added Mass Coefficient 1 (β_0)	0.9494
Lamb's Coef. 1 (k_1)	0.0876	Lamb's Coef. 2 (k_2)	0.9037
Lamb's Coef. 1 (k')	0.6271	Total Volume (V)	0.0629 m^3
First Section Area (A_1)	0.0294 m^2	Second Section Area (A_2)	0.1192 m^2
Third Section Area (A_3)	0.0654 m^2	Total Cross-Sec. Area (A_1)	0.4284 m^2

Added mass hydrodynamic coefficients found by the strip theory are given in Table (2.8). Those are the non-dimensionalized values. Since the units of the axial, crossflow and rolling added mass coefficients are different, non-dimensionalization is achieved in different ways.

The units with kg are divided by $\frac{\rho}{2}L^3$, $kg.m$ with $\frac{\rho}{2}L^4$ and others with similar methods. After non-dimensionalization we founded the added mass coefficients as

Table 2.3: Added Mass Coefficients

$X_{\dot{u}}$	-3.2×10^{-3}	$K_{\dot{p}}$	0
$Y_{\dot{v}}$	-3.33×10^{-2}	$M_{\dot{q}}$	-4.6×10^{-3}
$Z_{\dot{w}}$	-3.33×10^{-2}	$N_{\dot{r}}$	-1.2×10^{-3}
$Y_{\dot{r}}$	1.38×10^{-3}	$N_{\dot{v}}$	1.38×10^{-3}
$Z_{\dot{q}}$	-1.38×10^{-2}	$M_{\dot{w}}$	-1.38×10^{-3}

Weight of our vehicle is found by the formula

$$W = m\mathfrak{g} \tag{2.70}$$

and buoyancy with

$$B = \rho V\mathfrak{g} \tag{2.71}$$

where m denotes the mass, $\mathfrak{g}$ gravity, V volume of our vehicle and ρ the density of water. Hence after calculations, W is found $627kgm/s$ and buoyancy is found as $631kgm/s$, denoting that our vehicle is slightly positive buoyant which is a desired conclusion

Quadratic damping coefficients are calculated by the piecewise integrals. Piecewise integrals for strip theory are used because of the different shape of our vehicle shown in Figure (2.5). Strip theory is derived for ellipsoid shaped bodies therefore to apply the theory to our vehicle we inspected the our shape in three sections. We calculated the cross-sectional area for each section and added them to find the total area. Again we used the same procedure that we used in non-dimensionalization of the added mass coefficients here.

Calculated coefficients by Equations (2.65)-(2.69) are shown in Table (2.4).

2.9 Equations of Motion

In this section we will show the vectorial representation of the body-fixed equations of motion for our vehicle. In Section 2.3 we have expressed the nonlinear equations of motion in

body-fixed frame as

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + \mathfrak{g}(\eta) = \tau \tag{2.72}$$

$$\dot{\eta} = T(\eta)\nu \tag{2.73}$$

where

$$M = M_{RB} + M_A \qquad C(\nu) = C_{RB}(\nu) + C_A(\nu) \tag{2.74}$$

$$D(\nu) = D_{skin}(\nu) + D_{vortex}(\nu) \tag{2.75}$$

After our simplifications we obtain the component matrices of the equations of motion as

$$M = M_{RB} + M_A = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 & 0 & 0 & m - N_{\dot{v}} \\ 0 & 0 & m - Z_{\dot{w}} & 0 & m - M_{\dot{w}} & 0 \\ 0 & 0 & 0 & m - K_{\dot{p}} & 0 & 0 \\ 0 & 0 & m - Z_{\dot{q}} & 0 & m - M_{\dot{q}} & 0 \\ 0 & m - Y_{\dot{r}} & 0 & 0 & 0 & m - N_{\dot{r}} \end{bmatrix} \tag{2.76}$$

Table 2.4: Linear Quadratic Damping Coefficients

$X_{\|u\|u}$	-3.9×10^{-3}	$Y_{\|r\|r}$	-2.99×10^{-5}
$Y_{\|v\|v}$	-9.77×10^{-2}	$Z_{\|q\|q}$	2.99×10^{-5}
$Z_{\|w\|w}$	-9.77×10^{-2}	$N_{\|r\|r}$	7.51×10^{-2}
$K_{\|p\|p}$	0	$M_{\|w\|w}$	1.694×10^{-1}
$M_{\|q\|q}$	-7.51×10^{-2}	$N_{\|v\|v}$	-1.694×10^{-1}

$$C_A = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix} \tag{2.77}$$

$$C_{RB} = \begin{bmatrix} 0 & 0 & 0 & mz_G r & mw & -mv \\ 0 & 0 & 0 & -mw & mz_G r & mu \\ 0 & 0 & 0 & -m(z_G p + v) & -m(z_G q + u) & 0 \\ -mz_G r & mw & m(z_G p + v) & 0 & I_{yz}q + I_z r & I_{yz}r - I_y q \\ -mw & -mz_G r & m(z_G q + u) & I_{yz}q + I_z r & 0 & I_x p \\ mv & -mu & 0 & -I_{yz}r + I_y q & -I_x p & 0 \end{bmatrix} \tag{2.78}$$

$$D_v(v) = -$$

$$\begin{bmatrix} X_u + X_{|u|u}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v + Y_{|v|v}|v| & 0 & 0 & 0 & Y_r + Y_{|r|r}|r| \\ 0 & 0 & Z_w + Z_{|w|w}|w| & 0 & Z_q + Z_{|q|q}|q| & 0 \\ 0 & 0 & 0 & K_p + K_{|p|p}|p| & 0 & 0 \\ 0 & 0 & M_w + M_{|w|w}|w| & 0 & M_q + M_{|q|q}|q| & 0 \\ 0 & N_v + N_{|v|v}|v| & 0 & 0 & 0 & N_r + N_{|r|r}|r| \end{bmatrix} \tag{2.79}$$

Last of all our gravitational and buoyant forces matrix will be as

$$\mathfrak{g} = \begin{bmatrix} (W-B)s\theta \\ -(W-B)c\theta s\phi \\ -(W-B)c\theta c\phi \\ -(y_G W - y_B B)c\theta c\phi + (z_G W - z_B B)c\theta s\phi \\ (z_G W - z_B B)s\theta + (x_G W - x_B B)c\theta c\phi \\ -(x_G W - x_B B)c\theta s\phi - (y_G W - y_B B)s\theta \end{bmatrix} \tag{2.80}$$

2.10 Summary

In this chapter the mathematical model of our vehicle is formed by adding damping equations, gravitational and buoyant forces to rigid body dynamics. Rigid body dynamics are generated according to [8] using Newton's second law. In order to evaluate the motion of the underwater vehicle in earth-fixed coordinate system, kinematic transformations are needed, therefore the relation of motion between earth-fixed and body-fixed coordinate system is built. Then we found some of our damping coefficients via strip theory. Considering the 2D cross sectional structure of our vehicle, its body is partitioned into three sections. A formula for each section is derived in order to evaluate the strip theory and the coefficients are derived. At the end of the chapter, simplified equations of motions are stated for our vehicle. In simplifications, some assumptions like symmetry and other similarities are used.

Since some of the coefficients can not be derived directly by analytic formula they are estimated by relating them with other coefficients. The values of the hydrodynamic coefficients are very small compared to the mass of the vehicle therefore small errors in coefficients will not severely effect control of our system. Furthermore our controllers are designed to be robust enough to compensate the changes in coefficients, also the errors accumulated from wrong calculations. A requirement to possess the optimum values of hydrodynamic coefficients led us to solve a parameter estimation problem in Chapter 4.

CHAPTER 3

CONTROL PROCEDURES

3.1 Introduction

In this chapter, general control methods for our system are explained. We started control procedure by first linearizing our system around an equilibrium point and found a linear model in order to apply linear control methods. Then we decoupled the system into three linear subsystems of speed, steering and depth control. Controlling the speed of an underwater vehicle is an indispensable process before starting to control other components, therefore we designed a speed controller first. Then we designed steering and depth controllers for our vehicle. In the design process we analyzed PID, SMC and LQR/LQG methods. Since not all the components of our states for each designer are observable, an estimator is needed where we used the continuous Kalman Filter as the optimal estimator when noise is assumed as Gaussian.

Section 3.2 describes the linearization of our vehicle and decoupling of the system. Section 3.3 is comprised of speed control achieved via PID method. Section 3.4 contains the information about steering control procedure using optimal control, SMC and PID. Also SMC and optimal control (LQR) methods are described in this section. Depth control attained with same methods are described in Section 3.5. In Section 3.6 information about LQG design is given and also its difference from LQR is explained. Because of the necessity of the Kalman filter in designing LQG controller, a brief information about it is given in Section 3.7. Lastly a summary of this chapter in given in Section 3.8.

3.2 Linearized Equations of Motion

Underwater vehicles operating in complex environments with coupled maneuvers are known to be highly nonlinear, nevertheless to exploit the advantage of enhanced control methods we prefer to linearize our model around an equilibrium point, which is a constant speed for our case. Hence linearizing our model and achieving some simplifications a sort of well known control methods should be applied easily. Since linearization is the approximation of the nonlinear system near the linearization point, we will gain the information about our system in general.

In our configuration we have 4 thrusters, 2 of them are placed vertically and other 2 horizontally. Horizontal thrusters are used both for speed and steering (yaw) and vertical thrusters are used for depth control. In order to achieve robust control, we have chosen the Decoupled Control Method hence divided the 6 DOF (Degree of Freedom) motion into three main subsystems and designed different control methods for each subsystem. We achieved separation as

1. Speed System
2. Steering System
3. Diving System

Design and analysis phase of all work is done by using Control Toolbox in *MATLAB* and *Simulink*.

We know that the nonlinear equations of motion for an underwater vehicle can be written as

$$M(\dot{(}\nu)) + C(\nu)\nu + D(\nu)\nu + \mathfrak{g}(\eta) = \tau \tag{3.1}$$

$$\dot{\eta} = J(\eta)\nu \tag{3.2}$$

Nonlinear equations can be linearized around an equilibrium point, but a linearization point should be defined first, for our case which can be defined as

$$\nu_0(t) = [u_0(t), v_0(t), w_0(t), p_0(t), q_0(t), r_0(t)] \tag{3.3}$$

$$\eta_0(t) = [x_0(t), y_0(t), z_0(t), \phi_0(t), \theta_0(t), \psi_0(t)] \tag{3.4}$$

Since linearization is based on the perturbations from equilibrium point, perturbations can be defined as

$$\Delta\nu(t) = \nu(t) - \nu_0(t), \quad \Delta\eta(t) = \eta(t) - \eta_0(t), \quad \Delta\tau(t) = \tau(t) - \tau_0(t) \tag{3.5}$$

then our equations of motion can be written as [8]

$$M\Delta\dot{\nu} + \left.\frac{\partial C(\nu)\nu}{\partial\nu}\right|_{\nu_0}\Delta\nu + \left.\frac{\partial D(\nu)\nu}{\partial\nu}\right|_{\nu_0}\Delta\nu + \left.\frac{\partial \mathfrak{g}(\eta)}{\partial\eta}\right|_{\eta_0} = \Delta\tau \tag{3.6}$$

The kinematic transformation equation becomes

$$\dot{\eta}_0 + \Delta\dot{\eta} = J(\eta_0 + \Delta\eta)(\nu_0 + \Delta\nu) \tag{3.7}$$

Substituting initial condition for η, $\dot{\eta}_0 = J(\eta_0)\nu_0$ in Equation (3.7) yields

$$J(\eta_0)\nu_0 + \Delta\dot{\eta} = J(\eta_0 + \Delta\eta)\nu_0 + J(\eta_0 + \Delta\eta)\Delta\nu \tag{3.8}$$

which can be written as

$$\Delta\dot{\eta} = J(\eta_0 + \Delta\eta - J(\eta_0))\nu_0 + J(\eta_0 + \Delta\eta)\Delta\nu \tag{3.9}$$

Linear time invariant equations of motion can be derived by the assumption that the vehicle is moving in the longitudinal plane with non-zero velocities of u_0 and w_0. Also adding the assumption that other initial velocities are zero, $v_0 = p_0 = q_0 = r_0 = 0$ and equilibrium point is defined by zero roll and pitch angles, $\phi_0 = \theta_0 = 0$, linear time-invariant matrices are obtained as

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}-M^{-1}(C+D) & -M^{-1}G\\ J & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}M^{-1}\\ 0\end{bmatrix}u \tag{3.10}$$

where $x_1 = \Delta\nu$, $x_2 = \Delta\eta$ and $u = \tau$. The matrices except J in (3.10) are defined at the end of Chapter 2. The J matrix is defined as [8]

$$J = \begin{bmatrix}J_1 & 0\\ 0 & I\end{bmatrix} \qquad J_1 = \begin{bmatrix}cos\,\psi_0 & -sin\,\psi_0 & 0\\ sin\,\psi_0 & cos\,\psi_0 & 0\\ 0 & 0 & 1\end{bmatrix} \tag{3.11}$$

In the process of designing controllers, the thruster model defined in [1] is used. In that model, thrust and torque for an underwater propeller can be stated as

$$T = \rho D^4 C_T |n| n \tag{3.12}$$

$$Q = \rho D^5 C_Q |n| n \tag{3.13}$$

where T defines thrust, Q torque, ρ density of fluid, D propeller diameter, n propeller revolution, C_T thrust coefficient and C_Q torque coefficient. Thrust coefficient is directly related with advance coefficient J, which can be defined as $J = U_a/nD$. Relation between advance speed of vehicle and control voltage applied to thrusters is shown in Figure (3.1).

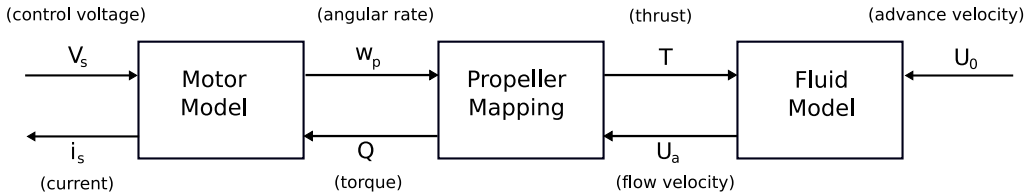

Figure 3.1: Relation between Thruster Components

3.3 Speed Control

Today robust and reliable control stands as a first priority towards the development of efficient underwater vehicles most of which operate in strict and tough conditions. On nonlinear control systems, modeling inaccuracies may cause to undesired effects, hence to deal with model uncertainties robust control methods are needed. On the other hand when the simple systems or the simplified models of complex systems are acquired, basic control methods like PID (Proportional Integral Differential) is preferred.

In the speed control due to simplicity of our model where the effects from sway, heave, roll, pitch and yaw are neglected, PID Control method is preferred and the control models are designed by using Simulink. Neglecting other effects, a SISO model with one state and one input is obtained as in [8].

$$(m - X_{\dot{u}})\dot{u} = X_{|u|u}|u|u + \tau + X_{ext} \tag{3.14}$$

In equation acquisition, some effects like Coriolis and centripetal forces are omitted nevertheless quadratic damping is taken as the main disturbing effect. Here τ stands for the horizontal thruster force. For linear case it is known that 1^{st} order approximation of the thrust force τ is equal to,

$$\tau = \rho D^4 C_T(J_0)|n|n \tag{3.15}$$

where ρ states the density of sea water, D propeller diameter, C_T advance thruster coefficient which is a function of advance number ($J_0 = U_a/nD$) , U_a water speed passing through the propeller and lastly n for the revolution of propeller.

Hence our equation can be written as,

$$(m - X_{\dot{u}})\dot{u} = X_{|u|u}|u|u + \tau + X_{ext} \tag{3.16}$$

τ stands for the thruster force, which is found by (3.15).

PID control is applied to one DOF model with positive coefficients of K_p, K_d and K_i selected with respect to the response of the system. It is assumed that the state and output is directly measurable and the model parameters are obtain according to the formulas that stated on the previous chapter. A white Gaussian noise is added to the system as an external disturbance in order to raise the reality of model compared with the actual one.

$$\tau = K_p(x(t) - x_d(t)) + K_d(\dot{x}(t) - \dot{x}_d(t)) + K_i \int_0^t (x(\tau) - x_d(\tau))d\tau \tag{3.17}$$

Most underwater vehicle controllers prefer PI- control law instead of PID in order to get rid of necessity for Kalman filter, which should be designed for estimating the derivative of surge and angular propeller accelerations. Since PID parameters are found by response optimization in Simulink, we preferred using PID triple parameters instead of PI- control law. PID parameters for first block are found as: $K_p = 16.1$, $K_i = 1.1$ and $K_d = 0.01$ and for the second block, parameters are found as: $K_p = 6.3$, $K_i = 0.001$ and $K_d = 0.01$. Simulink diagram of the Speed Controller is shown in Appendix C.

It can be seen that the effect of the disturbance in Figure (3.3) where PID controller can not quickly compensate the error.

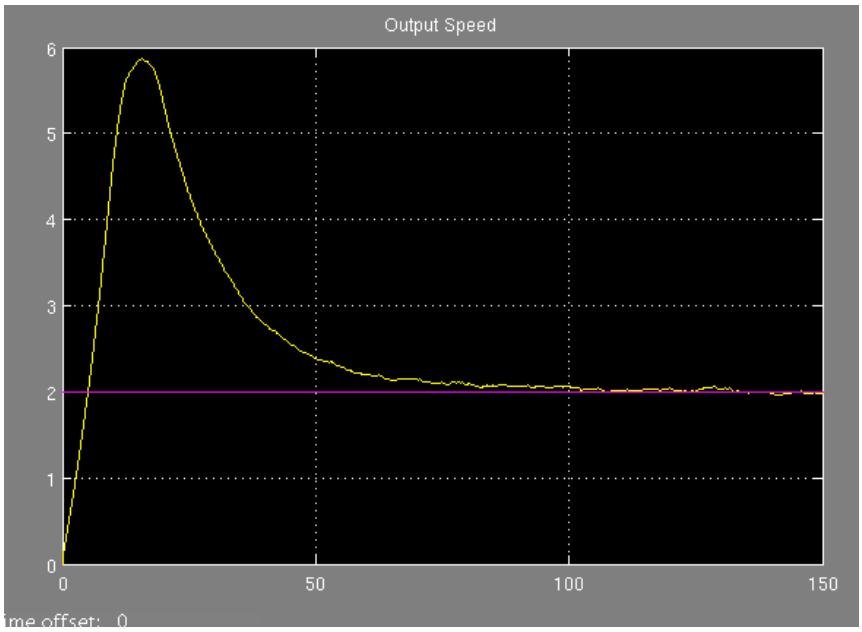

Figure 3.2: Commanded and Real Output Velocities (m/s) for PID

3.4 Steer Control

Before starting steering control, information about SMC and Optimal Control methods will be given.

3.4.1 Sliding Mode Control

As being one of the most robust control methods, Sliding Mode Control (SMC) is based on the philosophy that it is easier to control 1^{st} order systems compared with high order systems ($n > 1$). Therefore any modeling and parameter inaccuracies can be compensated with this method though in a wearisome manner. In addition to stated assets, SMC brings us the advantage to face with strong perturbations like currents, waves and other unpredicted effects in complex sea environment. General application of SMC to underwater vehicles consists of designing a controller for the linearized part of the system and considering the nonlinearities as the parametric uncertainties. In design step we face with two different sliding surface selections. In the first method we select a scalar function of form $s = \dot{e} + \lambda e$, which is the sum of the position error and the velocity vector. For $s = 0$ this functions defines a sliding

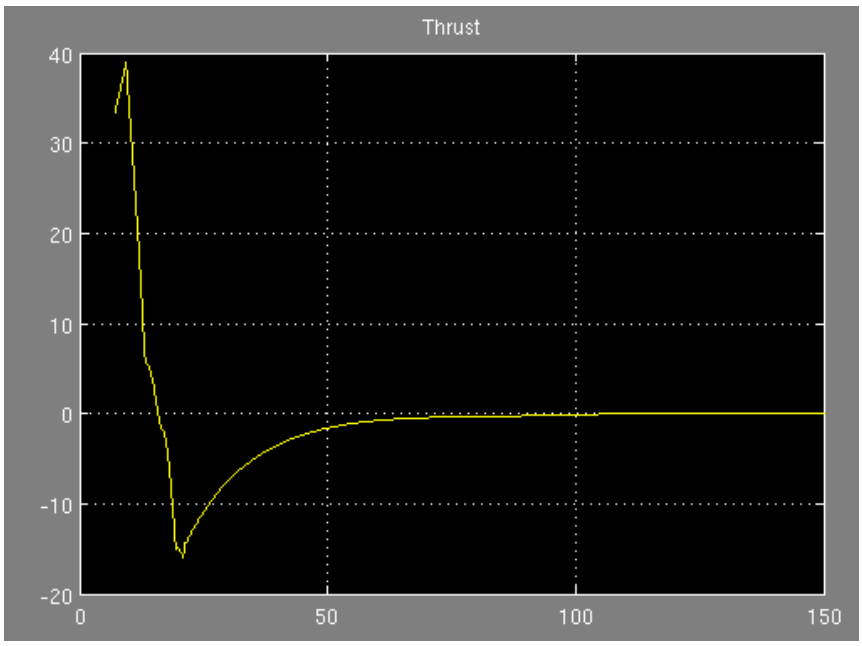

Figure 3.3: Thrust Output in Speed Control

surface ensuring that the tracking error e converges to zero. In the second method sliding surface is based on the state variable errors depicted as: $\sigma(e) = s^T e$.

In the first method we start design by defining the tracking error vector with $e = x - x_d$ where x stands for the state vector, x_d for the desired state vector. Then we define a scalar time-varying surface $s(t)$ in $\Re^n$ by the scalar equation $s(x,t) = 0$, where

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1} e \tag{3.18}$$

with λ being a strictly positive constant. In a general manner, choosing $n = 2$ we get a weighted sum of the position error and the velocity vector. For $s = 0$, our surface defines a sliding surface with dynamics:

$$e(t) = e^{-\lambda(t-t_0)} e(t_0) \tag{3.19}$$

guaranteeing that tracking error $e(t)$ will converge to zero exponentially in finite time whatever the initial condition is.

As a second method when the coupled movements considered, using the sliding surface based on the state variable errors instead of the output errors seems to be more logical and

more useful especially in underwater environment. In that manner sliding surface is defined as $\sigma(e) = s^T e$ where $e = x - x_d$ is the state tracking error, $s \in \Re^n$ is an arbitrary vector to be evaluated in the end. It will be a sufficient condition to lead the sliding surface to zero $(\sigma(e) \to 0)$ for the convergence of the state tracking error to zero $(e \to 0)$.

Assuming our model as:

$$\dot{x} = Ax + Bu + f(x) \tag{3.20}$$

where for our case $x \in \Re^n$, $u \in \Re^m$, $A \in \Re^{nxn}$, $B \in \Re^{mxn}$, $f(x)$ acting as the deviation from linearity because of modeling errors and environmental disturbances. Feedback control input can be taken as:

$$u = \bar{u} + \hat{u} \tag{3.21}$$

where $\hat{u}$ is the linear feedback part of model and $\bar{u}$ is the nonlinear feedback control that has a compensating effect.

Nominal part of control is chosen as:

$$\hat{u} = -k^T x \tag{3.22}$$

where k stating the feedback gain vector. Applying this input into our linear model we obtain the closed loop dynamics:

$$\dot{x} = (A - Bk^T)x + B\bar{u} + f(x) = A_c x + B\bar{u} + f(x) \tag{3.23}$$

Here feedback gain vector can be determined by pole placement or optimal control methods. To find the compensating part of the control input, we have to keep (3.23) satisfying that $\sigma(e) \to 0$, which requires $\dot{\sigma}(e) < 0$. From the definition of sliding surface we know that $\dot{\sigma}(e) = s^T(\dot{x} - \dot{x}_d)$ hence multiplying (3.23) with s^T from left and subtracting $s^T\dot{x}_d$ from both side yields:

$$\dot{\sigma}(e) = s^T A_c x + s^T B\bar{u} + s^T f(x) - s^T \dot{x}_d \tag{3.24}$$

Assuming that $s^T B \neq 0$, we choose compensating part of control, $\bar{u}$ as:

$$\bar{u} = (s^T B)^{-1} \left[s^T \dot{x}_d - s^T \hat{f}(x) - \eta\, sgn(\sigma) \right] \tag{3.25}$$

and applying to the equation yields:

$$\dot{\sigma}(e) = s^T A_c\, x - \eta\, sgn(\sigma(e)) + s^T \Delta f(x) \tag{3.26}$$

Now we work on s. We know that if λ stating the eigenvalue of an arbitrary matrix M, following equation satisfies with a nonzero vector $\bar{v}$,

$$M\bar{v} = \lambda\bar{v} \tag{3.27}$$

Then assigning one of the eigenvalues of A_c as zero, the term $s^T A_c x$ in (3.26) can be made zero by taking vector s as the right eigenvector of A_c^T corresponding to the eigenvalue with zero value.

Eliminating the term $s^T A_c x$ in (3.26) yields:

$$\dot{\sigma}(e) = -\eta\, sgn(\sigma(e)) + s^T \Delta f(x) \tag{3.28}$$

This term is global asymptotically stable in case of,

$$\eta > ||s||.||\Delta f(x)|| \tag{3.29}$$

which can be shown by the *Barbălat*'s lemma first by selecting a candidate Lyapunov function as:

$$V(\sigma) = \frac{1}{2}\sigma^2 \tag{3.30}$$

which ensures that $V(\sigma) > 0$ then differentiating V, we get:

$$\dot{V}(\sigma) = \sigma\dot{\sigma} = -\eta\sigma sgn(\sigma) + \sigma s^T \Delta f(x) = -\eta|\sigma| + \sigma s^T \Delta f(x) \tag{3.31}$$

from that equation it is clear that selecting η as stated in (3.29), $\dot{V}(\sigma)$ becomes negative semi definite ($\dot{V}(\sigma) \leq 0$). Lastly taking second derivative of V yields:

$$\ddot{V}(\sigma) = \eta^2 sgn^2(\sigma) - \eta s^T \Delta f(x) sgn(\sigma) - \eta sgn(\sigma) s^T \Delta f(x) + (s^T \Delta f(x))^2 + \sigma s^T \Delta \dot{f}(x) \tag{3.32}$$

It can be easily seen that $\ddot{V}(\sigma)$ is bounded. Hence (3.30), (3.31) and (3.32) satisfies *Barbălat*'s lemma asserting that:
if,

1. $V(\sigma,t)$ is lower bounded. ($V(\sigma) \geq 0$)
2. $\dot{V}(\sigma,t)$ is negative semi definite. ($\dot{V}(\sigma) \leq 0$)
3. $\dot{V}(\sigma,t)$ is uniformly continuous in time. ($\ddot{V}(\sigma)$ is bounded $\forall t \geq t_0$)

then $V(\sigma,t) \to 0$ as $t \to \infty$.

That fact in accordance with (3.29) brings the consequence of σ to converge to zero in finite time.

$$V(\sigma,t) \to 0 \quad \Longrightarrow \quad -\eta|\sigma| + \sigma s^T \Delta f(x) \to 0 \quad \Longrightarrow \quad \sigma \to 0 \tag{3.33}$$

To conclude our calculations, combining two parts of control effort as stated in (3.21), our control law becomes,

$$u = -k^T x + (s^T B)^{-1}[s^T \dot{x}_d - s^T \hat{f}(x) - \eta\, sgn(\sigma)] \tag{3.34}$$

In practice to reduce the chattering effect instead of $sgn(\sigma)$ function, $sat(\sigma/\phi)$ or $tanh(\sigma/\phi)$ functions are used with ϕ appears as the sliding surface boundary layer thickness.

Finally we obtain the modified control law as:

$$u = -k^T x + (s^T B)^{-1}[s^T \dot{x}_d - s^T \hat{f}(x) - \eta\, tanh(\sigma/\phi)] \tag{3.35}$$

3.4.2 Steering Control with SMC

Steering control with SMC is found by first linearizing the system around an equilibrium point and substituting the coefficients in the linearized matrix. Decoupling the system for an efficient control, the following matrix is obtained.

Here υ stands for sway, r for yaw and ψ for the Euler Angle for steering.

$$\begin{bmatrix} m - Y_{\dot{\upsilon}} & -Y_{\dot{r}} & 0 \\ -N_{\dot{\upsilon}} & (I_{xz} - N_{\dot{r}}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\upsilon} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} Y_{\upsilon} & mu - Y_r & 0 \\ N_{\upsilon} & N_r & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \upsilon \\ r \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ \tau_N \\ 0 \end{bmatrix} \tag{3.36}$$

The sliding surface is taken as

$$\sigma = s_1\, \upsilon + s_2\, r + s_3\, (\psi - \psi_d) \tag{3.37}$$

and our model is transformed into state space form as

$$\dot{x} = Ax + Bu \tag{3.38}$$

Hence using the simulation values, the following equation is obtained

$$\dot{x} = \begin{bmatrix} -0.0943 & -0.0322 & 0 \\ -0.9537 & -1.1733 & 0 \\ 0 & 1 & 0 \end{bmatrix} x + \begin{bmatrix} -1.6627 \\ 4.5960 \\ 0 \end{bmatrix} u \tag{3.39}$$

Then A_c matrix is found

$$A_c = (A - Bk^T) = \begin{bmatrix} -0.3653 & -0.2994 & 0 \\ -0.2047 & -0.4347 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.40}$$

where the vector k is determined by the pole-placement method. Choosing the poles as

$$p = [-0.15; -0.65; 0]^T \tag{3.41}$$

the k vector is found as

$$k = [-0.1630; -0.1607; 0]^T \tag{3.42}$$

Then the right eigenvector is found as

$$s = [-0.4762; 0.8496; 0.2268]^T \Longrightarrow A_c^T s = 0 \tag{3.43}$$

hence our sliding surface σ, takes the form

$$\sigma = -0.4762\upsilon + 0.8496r + 0.2268(\psi - \psi_d) \tag{3.44}$$

Lastly after finding all the coefficients, the control law turns into

$$\begin{aligned} u &= 0.1630\upsilon + 0.1607r \\ &+ \frac{1}{4.6965}(-0.23\, tanh(-0.4762\upsilon + 0.8496r + 0.2268(\psi - \psi_d)/0.05)) \end{aligned} \tag{3.45}$$

Here the sliding surface boundary layer thickness is selected as 0.05 which seems to be a proper value and η is selected as 0.23 in accordance with the balance between robustness and performance.

The model of the SMC Steering control is shown in Figure (3.4),

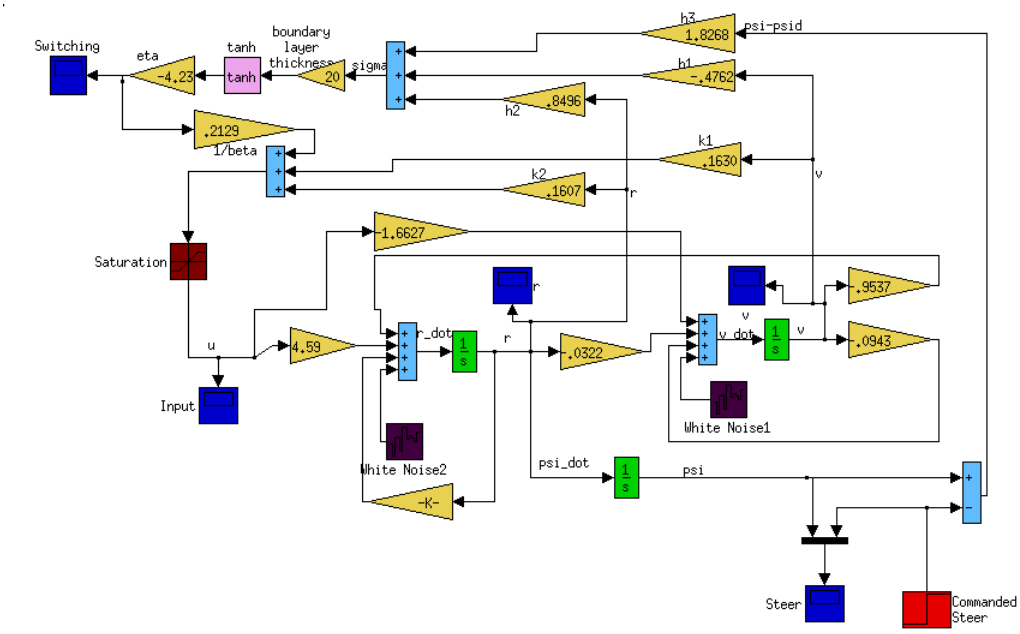

Figure 3.4: Model of Sliding Mode Controller for Steering

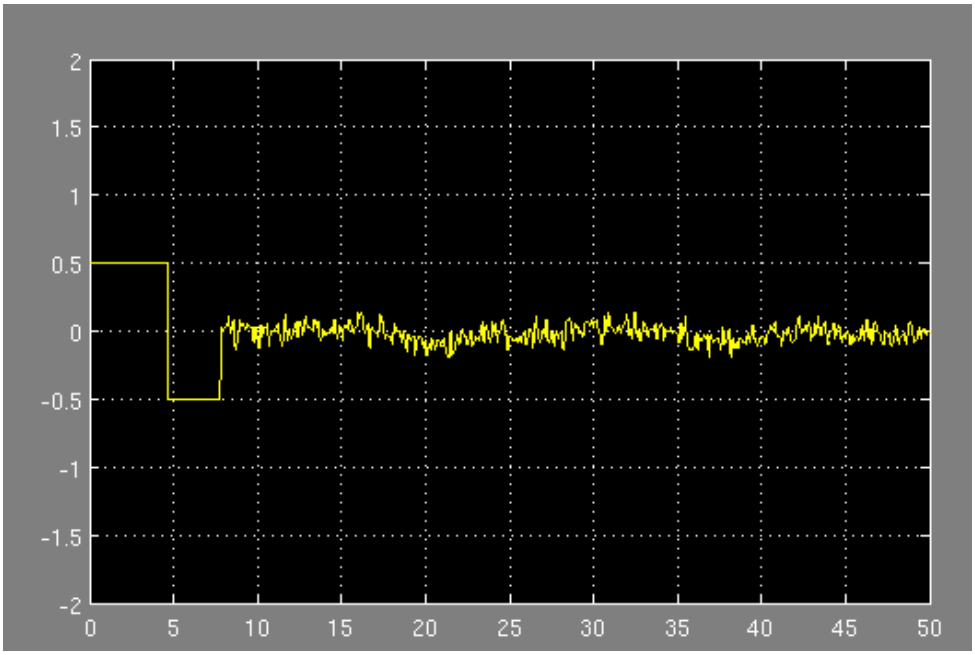

Figure 3.5: Input for Sliding Mode Controller for Steering

3.4.3 Optimal Control

Like the behavior of nature which is designed in a manner of always selecting the minimum effort with maximum performance, every system can be controlled with a minimum effort. Therefore in order to find the minimum effort we use Optimal Control.

Before starting work for optimal control, we know that our system is linear hence selecting a quadratic performance function we can apply the Linear Quadratic Regulator (LQR) rules, which requires a linear system and quadratic performance index to minimize in order to get minimum control effort. Here our aim is to find a control input which will guide the system to follow a desired state variable and meanwhile minimize a performance index which is chosen for minimum energy control system in conformity with our situation.

Considering a linear time-invariant system stated as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{3.46}$$

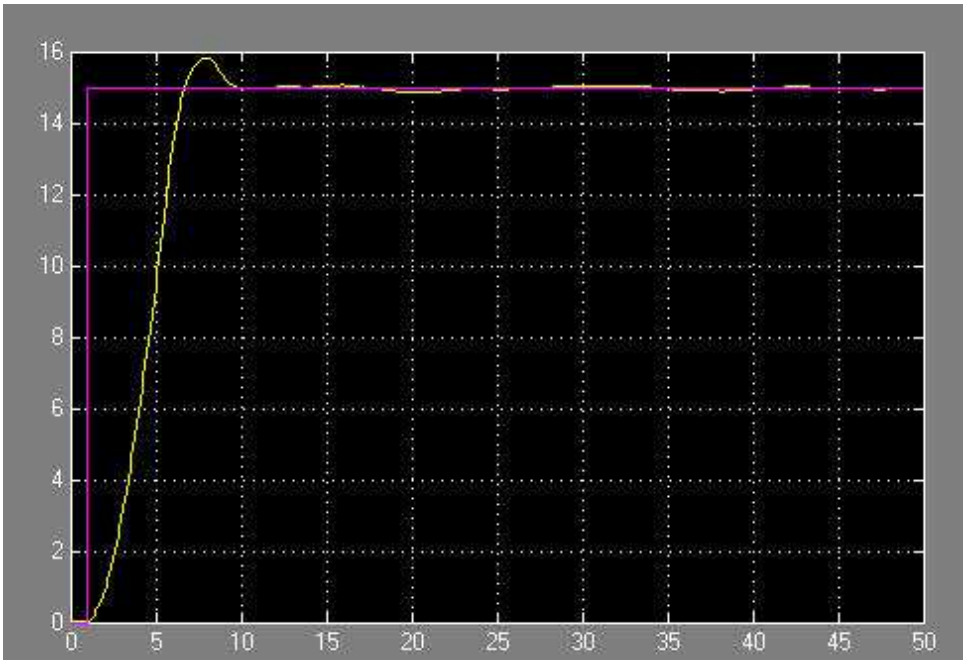

Figure 3.6: Steering for Sliding Mode Controller for Steering

with a cost functional

$$\begin{aligned} J(u(t)) &= J(x(t_0), u(t), t_0) \\ &= \frac{1}{2}\left[x_d(t_f) - x(t_f)\right]^T F \left[x_d(t_f) - x(t_f)\right] \\ &+ \frac{1}{2}\int_{t_0}^{t_f} \left[\left[x_d(t) - x(t)\right]^T Q \left[x_d(t) - x(t)\right] + u^T(t) R\, u(t)\right] dt \\ &= \frac{1}{2} e^T(t_f) F\, e(t_f) \\ &+ \frac{1}{2}\int_{t_0}^{t_f} \left[e^T(t)\; u^T(t)\right] \begin{bmatrix} Q(t) & 0 \\ 0 & R(t) \end{bmatrix} \begin{bmatrix} e(t) \\ u(t) \end{bmatrix} dt \end{aligned} \tag{3.47}$$

where error $e(t)$, states the difference between the reference (desired state) x_d and state x as $e(t) = x_d - x$. For infinite time interval case *final cost function* in the cost functional $J(u)$ does not have any meaning for the system and the final time t_f, is taken as infinite($t_f \to \infty$). Hence our cost functional can be written in a more simplified way as

$$J(u) = +\frac{1}{2}\int_{t_0}^{\infty} \left[x^T(t) Q\, x(t) + u^T(t) R\, u(t)\right] dt \tag{3.48}$$

We follow the standart procedure, first building with the *Hamiltonian* as

$$H(x(t),u(t),\lambda(t)) = \frac{1}{2}x^T(t)Q\,x(t) + \frac{1}{2}u^T(t)R\,u(t) + \lambda^T(t)\left[A\,x(t) + B\,u(t)\right] \tag{3.49}$$

where $\lambda(t)$ is the costate vector.

Then we obtain the optimal control $u(t)$ as

$$\frac{\partial H}{\partial u} = 0 \Longrightarrow R\,u(t) + B^T\,\lambda(t) = 0 \tag{3.50}$$

from which we find the control input as

$$u(t) = -R^{-1}\,B^T\,\lambda(t) \tag{3.51}$$

after that we obtain the state and costate equations as

$$\dot{x}(t) = +\left(\frac{\partial H}{\partial \lambda}\right) \Longrightarrow \dot{x}(t) = A\,x(t) + B\,u(t) \tag{3.52}$$

$$\dot{\lambda}(t) = -\left(\frac{\partial H}{\partial x}\right) \Longrightarrow \dot{\lambda}(t) = -Q\,x(t) - A^T\lambda(t) \tag{3.53}$$

with using (3.51) in (3.52) we get

$$\dot{x}(t) = A\,x(t) - B\,R^{-1}\,B^T\,\lambda(t) \tag{3.54}$$

In the infinite final-time interval optimal control, we have to satisfy that the system is completely controllable, which requires that the controllability matrix,

$$C = \left[B\ AB\ \ldots A^{n-1}B\right] \tag{3.55}$$

must be of *full rank* (*n linearly independent columns*). Having a controllable system guarantees that the optimal cost is *finite*.

Assuming a transformation

$$\lambda(t) = P\,x(t) \tag{3.56}$$

where P is not known yet, our new optimal control becomes

$$u(t) = -R^{-1}\,B^T\,\hat{P}\,x(t) \tag{3.57}$$

now which stands as a negative feedback of our error vector $x(t)$. Here,

$$\hat{P} = \lim_{t_f \to \infty} \{P\} \tag{3.58}$$

Continuing our replacements with differentiating (3.56) w.r.t. time yields,

$$\dot{\lambda}(t) = \dot{\hat{P}}\, x(t) + \hat{P}\, \dot{x}(t) \tag{3.59}$$

and using it in the state and costate equations we get,

$$\dot{x}(t) = A\, x(t) - B\, R^{-1}\, B^T\, \hat{P}\, x(t) \tag{3.60}$$

$$\dot{\lambda}(t) = -Q\, x(t) - A^T \hat{P}\, x(t) \tag{3.61}$$

Lastly, substituting our new equations (3.60) and (3.61) in (3.59) results,

$$\begin{aligned} -Qx(t) - A^T \hat{P} x(t) &= \dot{\hat{P}}\, x(t) + \hat{P}\, \left[A\, x(t) - B\, R^{-1}\, B^T\, \hat{P}\, x(t)\right] \\ 0 &= \left[\dot{\hat{P}} + \hat{P}\, A\, + A^T \hat{P}\, + Q - \hat{P}\, B\, R^{-1}\, B^T\, \hat{P}\, \right] x(t) \end{aligned} \tag{3.62}$$

$$\tag{3.63}$$

The equation stated above is called *differential Riccati equation (DRE)*, where the matrix $\hat{P}$ is often called *Riccati matrix*. $\hat{P}$ is an *n x n* symmetric, positive definite matrix found by the solution of the *DRE*, satisfying the final condition

$$\hat{P}(t_f \longrightarrow \infty) = 0 \tag{3.64}$$

Our optimal state is the found as

$$\dot{x}(t) = \left[A - B\, R^{-1}\, B^T\, \hat{P}\, \right]\, x \tag{3.65}$$

and the optimal cost is found by

$$J = \frac{1}{2} x^T(t) \hat{P}\, x(t) \tag{3.66}$$

3.4.4 Steering Control with Optimal Control

Optimal Control is applied to our linearized steering matrix, which is stated on the previous section. Hence substituting the coefficients in the matrices, following equation is obtained

$$\dot{x} = \begin{bmatrix} -0.0943 & -0.0322 & 0 \\ -0.9537 & -1.1733 & 0 \\ 0 & 1 & 0 \end{bmatrix} x + \begin{bmatrix} -1.6627 \\ 4.5960 \\ 0 \end{bmatrix} u \tag{3.67}$$

Using the optimal control equations, error weighting matrices Q and R are applied as

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.68}$$

$$R = 1/4$$

And also the steady state Riccati matrix $\hat{\mathrm{P}}$ is calculated using MATLAB *lqr* command

$$[K, P, Eig] = lqr\,(A, B, Q, R) \tag{3.69}$$

The solutions for control problem by using optimal control can be stated as

$$K = [-0.5949\ 1.9505\ 2.0000] \quad Eig = \begin{bmatrix} -9.7858 \\ -1.0377 \\ -0.3977 \end{bmatrix}$$

$$P = \begin{bmatrix} 1.1077 & 0.3684 & 0.4409 \\ 0.3684 & 0.2394 & 0.2683 \\ 0.4409 & 0.2683 & 1.3042 \end{bmatrix}$$

3.4.5 Steering Control with PID

Lastly for steering, PID method is applied for basic control and the efficiency of this method is compared with the other methods. A simple Simulink diagram is designed, figure of which is shown in Figure (**??**). Then the control law and the output of the system for PID is found.

Though it is expected that PID method will not succeed in compensating the noise when compared with SMC, after optimizing the parameters with response optimization method in

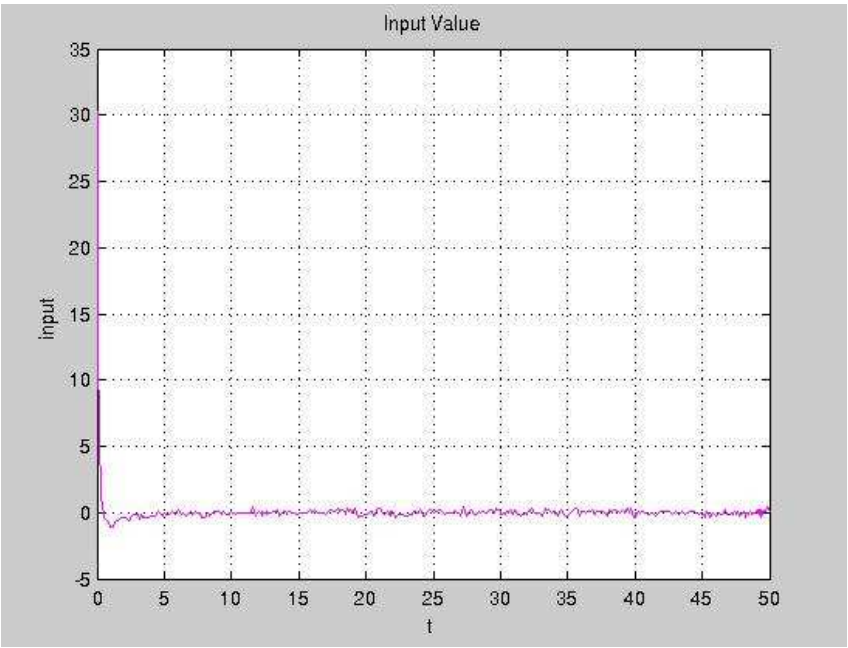

Figure 3.7: Control Input for Steering by Optimal Control

Simulink, PID method yielded similar outputs as SMC. Applying response optimization to our model, PID parameters are found as: $K_p = 0.2798$, $K_i = 0.00074$ and $K_d = 0.1175$.

Here again, the same amount of white Gaussian noise is injected to the system as in SMC method. Simulink diagram of the PID steering controller is shown in Appendix C.

3.5 Depth Control

On the depth control, the same procedure is followed as in the steering control. First solution with PID method is shown, then solution is given for SMC and lastly for optimal control. Like in the steering design, control methods responded in the same manner. SMC confirmed its robustness and stability in compensating noise more than other methods. Optimal control followed SMC, but evinced less efficiency. Lastly PID came into scene when the goal is robustness, but whatever the outputs are, we realized that also when PID coefficients are adjusted optimally desired outputs are not afar. Depth is not taken a high value due to the fact that the system will response in the same manner, no matter how deep the vehicle dives.

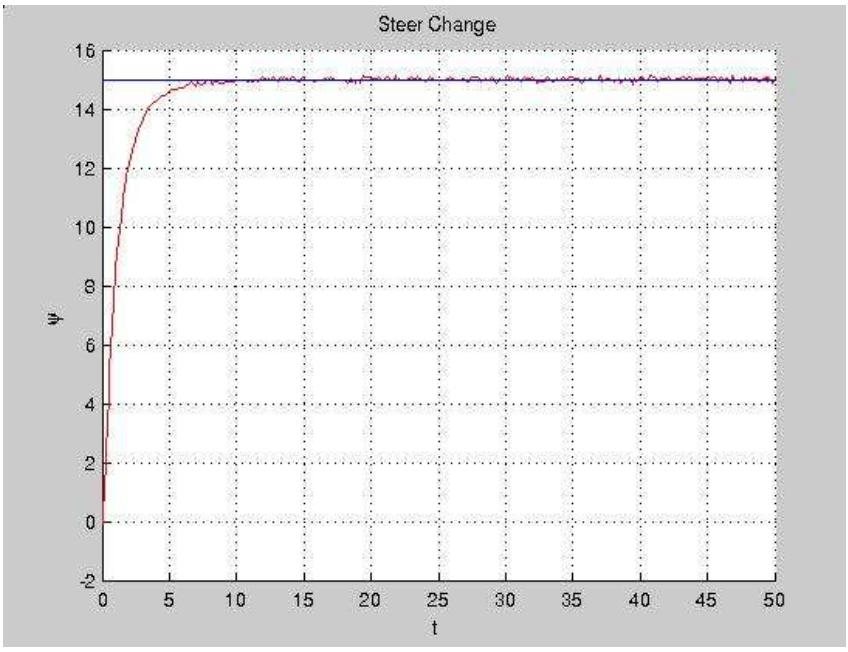

Figure 3.8: Steering Angle found by Optimal Control

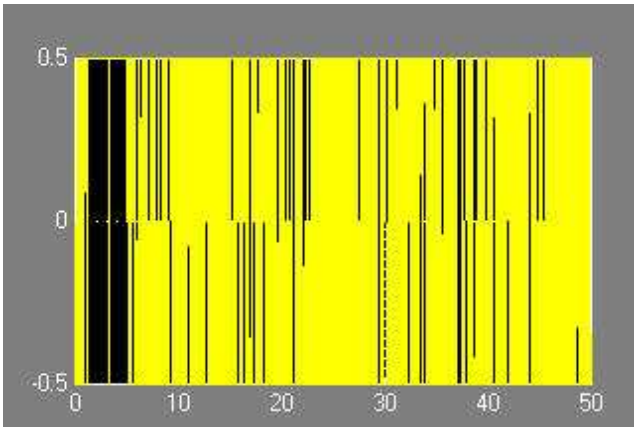

Figure 3.9: Input for PID Steering Control

3.5.1 Depth Control with PID

This time, depth control procedure is first initialized with PID method because of its direct approach. Coefficients are found by optimization using the response optimization block

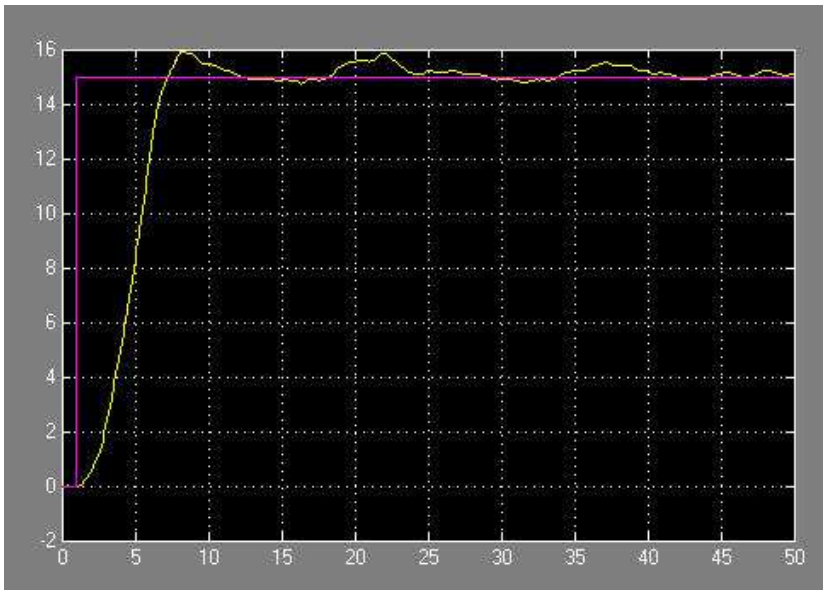

Figure 3.10: Steering Angle for PID Control

in Simulink and the solution is found at the 28th step as: $K_p = 1.2279$, $K_i = 0.0094$ and $K_d = 15.0032$. Model of the depth controller is shown in Appendix C.

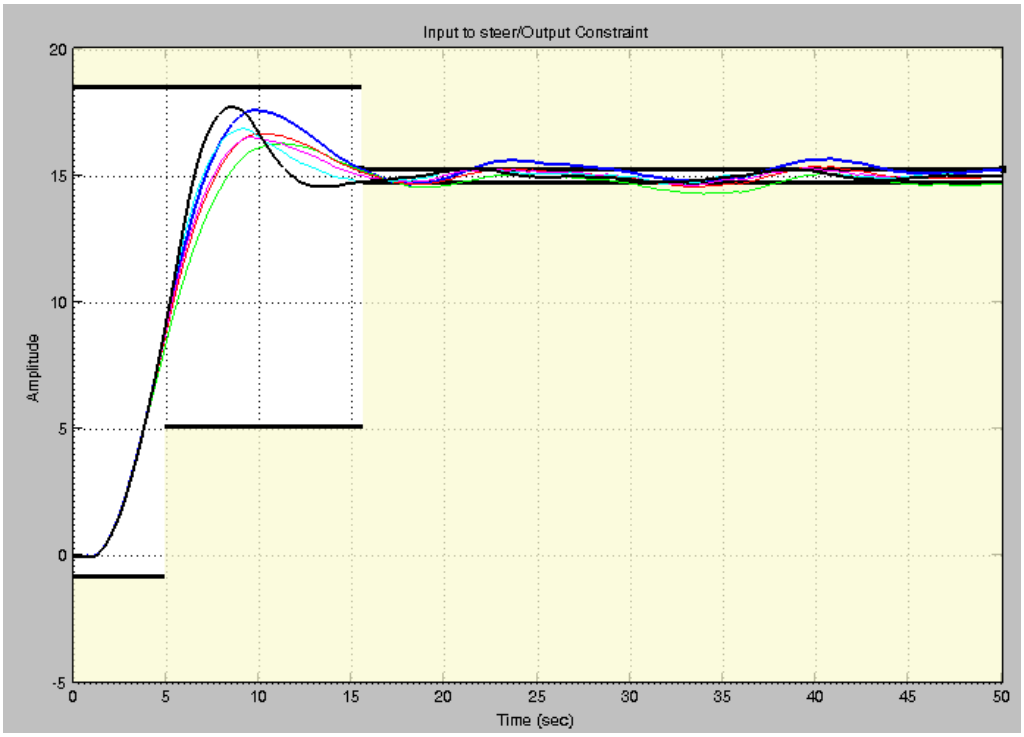

Figure 3.11: Optimization of PID Response for Steering

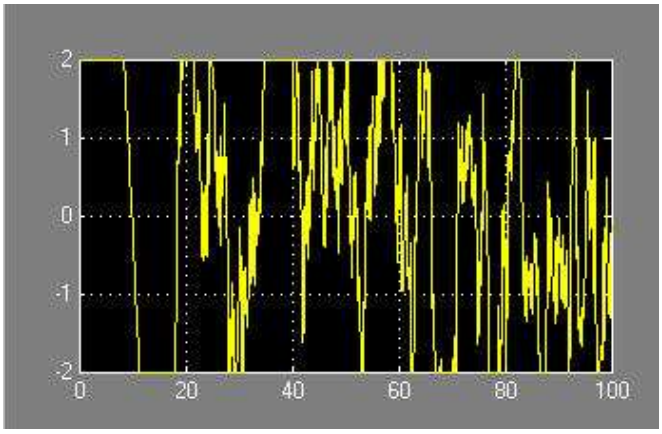

Figure 3.12: Input Value of Simulink PID Control for Depth

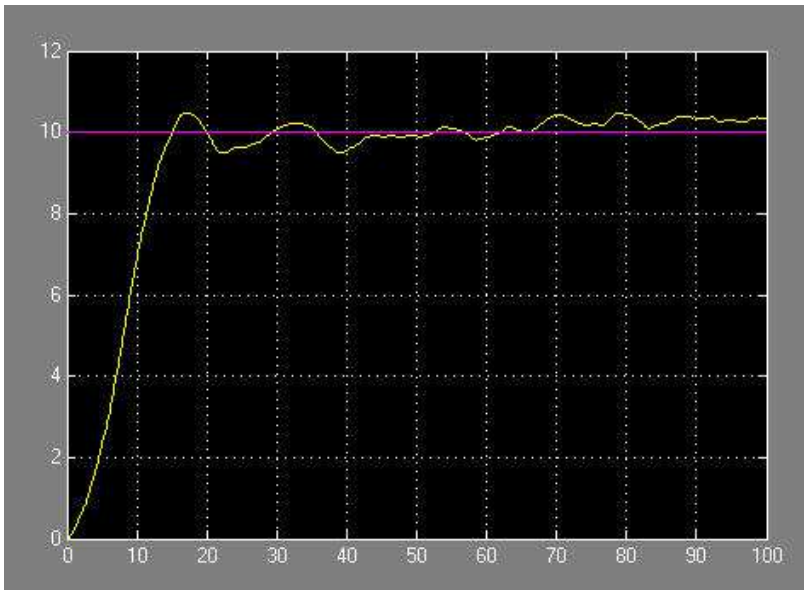

Figure 3.13: Desired Depth of PID for Depth Control

3.5.2 Depth Control with SMC

In depth control with SMC, we assumed that we achieve diving operation vertically without adding the pitch effect which means, we do not use our thrusters to maintain a desired pitch angle in order to dive, where we obtain a direct thrust vector perpendicular to sea surface. On the other hand, by resultant thrust vector which is the sum of the vertical thrust and horizontal thrust, our vehicle will decline on a slope.

After linearizing our system we get the following matrix for depth. Here w stands for heave, q for pitch and z for the depth.

$$\begin{bmatrix} m-Z_{\dot{w}} & mx_G-Z_{\dot{q}} & 0 \\ mx_G-W_{\dot{w}} & (I_y-M_{\dot{q}}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{z} \end{bmatrix} + \begin{bmatrix} Z_w & -mu+Z_q & 0 \\ M_w & mx_Gu+M_q & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ q \\ z \end{bmatrix} = \begin{bmatrix} \tau_Z \\ 0 \\ 0 \end{bmatrix} \tag{3.70}$$

The sliding surface is taken as

$$\sigma = s_1 w + s_2 q + s_3 (z - z_d) \tag{3.71}$$

and our model transformed into

$$\dot{x} = Ax + Bu \tag{3.72}$$

form via MATLAB yielding

$$\dot{x} = \begin{bmatrix} 1.4327 & 1.3356 & 0 \\ -7.6031 & -7.0186 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 2.1937 \\ -8.6137 \\ 0 \end{bmatrix} u \tag{3.73}$$

Then we find A_c matrix

$$A_c = (A - Bk^T) = \begin{bmatrix} -0.5061 & -0.3541 & 0 \\ -0.0099 & -0.3839 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{3.74}$$

where the vector k is found by the pole-placement method. Choosing the poles as

$$p = [0\,; -5.5680\,; -0.0179\,]^T \tag{3.75}$$

we find the k vector as

$$k = [-0.8838\,; 0.7703\,; 0\,]^T \tag{3.76}$$

Equating k_3 to 0 shows us that feedback from z has no effect to stabilize the heave-pitch dynamics. On the other hand, in order to simplify our equation we find the right eigenvector s of A_c corresponding to λ_3, which is equal to 0.

Then the right eigenvector is found as

$$s = [0.6874\,; -0.6340\,; 0.3542\,]^T \Longrightarrow A_c^T s = 0 \tag{3.77}$$

hence our sliding surface σ, takes the form

$$\sigma = 0.6874w - 0.6340q + 0.3542(z - z_d) \tag{3.78}$$

Lastly after finding all the coefficients, our control input turns into

$$\begin{aligned} u = &-0.8838w - 0.7703q \\ &+ \frac{1}{6.9693}(-0.11\,tanh(0.6874w - 0.6340q + 0.3542(z - z_d)/0.05)) \end{aligned} \tag{3.79}$$

Here the sliding surface boundary layer thickness is selected as 0.05 which seems to be a proper value and η is selected as 0.11 in accordance with the balance between robustness and performance.

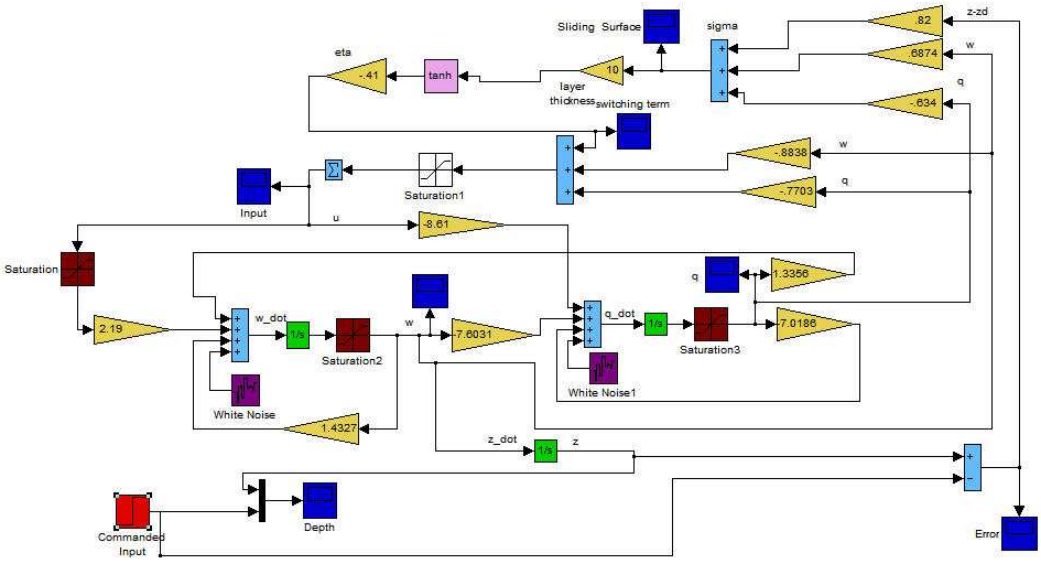

Figure 3.14: Model of Sliding Mode Control for Depth

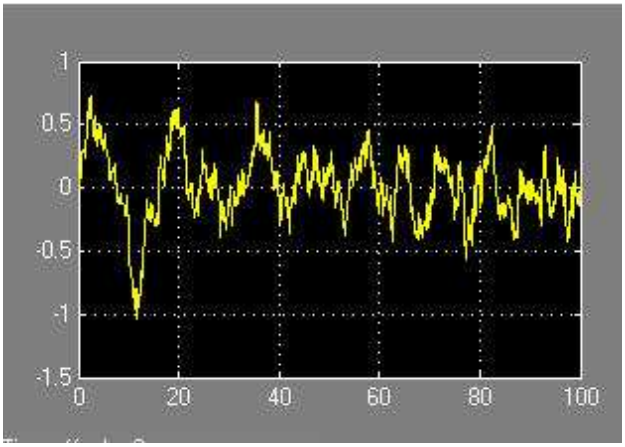

Figure 3.15: Input for Sliding Mode Depth Control

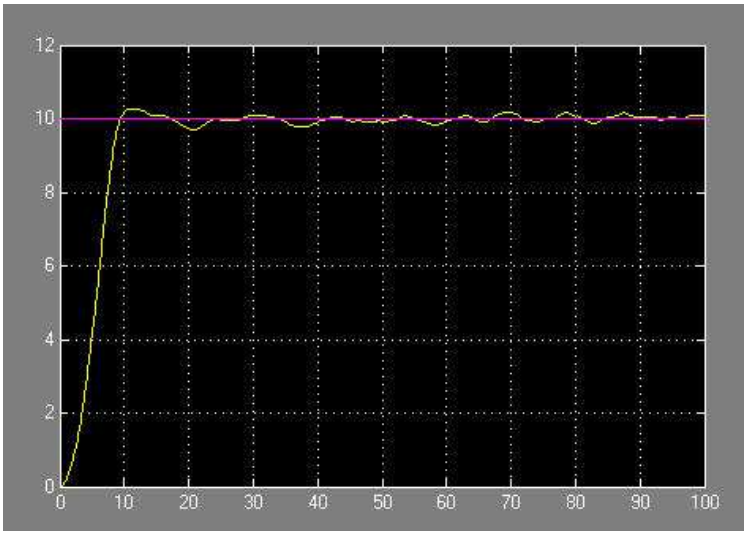

Figure 3.16: Desired Depth for Sliding Mode Control

3.5.3 Depth Control with Optimal Control

Optimal Control is applied to the same matrix used in SMC which is obtained by linearizing the system for depth control. Only the coefficients of the linearized matrix are different. Hence substituting the coefficients we found the following matrix

$$\dot{x} = \begin{bmatrix} 1.4327 & 1.3356 & 0 \\ -7.6031 & -7.0186 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 2.1937 \\ -8.6137 \\ 0 \end{bmatrix} u \tag{3.80}$$

Then we used the optimal control equations stated above and built a MATLAB m-file to obtain the desired output.

Here again Q and R matrices in cost functional are taken as the same with matrices in steer control.

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.81}$$

$$R = 1/4$$

Also we preferred to find the Riccati matrix $\hat{\mathrm{P}}$ from MATLAB with *lqr* command as

$$[K, P, Eig] = lqr\,(A, B, Q, R) \tag{3.82}$$

which yielded the K, P, Eig matrices as

$$K = [-0.5949\ 1.9505\ 2.0000]$$

$$P = \begin{bmatrix} 2.8419 & 0.6208 & 1.7216 \\ 0.6208 & 0.2394 & 0.3804 \\ 1.7216 & 0.3804 & 2.1995 \end{bmatrix}$$

3.6 LQG Design

In LQR design we have supposed that we have all the state information at each step but in real time design it will not be possible to acquire full state information. Environmental disturbances like currents, wind, waves, etc. will cause to system noise and also state information acquired from sensors will not be so definite because of the measurement noise hence system

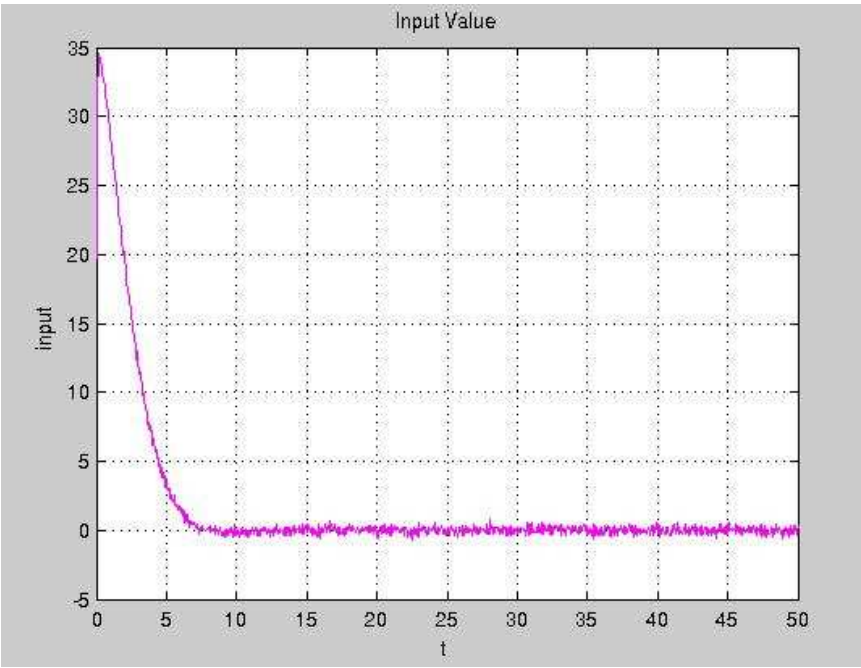

Figure 3.17: Control Input for Optimal Depth Control

will not be controlled efficiently as desired. Generally noise will set in too much oscillation which will end with too much chattering at the input for compensation. Therefore we need a filter in order to obtain optimum state estimates. In that manner Kalman filter is selected, where in case of white noise it is the optimal filter [35]. In our design we assumed that all noises are white otherwise we would design a Colored-Noise Kalman filter. Combining the LQR feedback design with an estimator forms the optimal system known as "Separation Principle". In this theorem, design is achieved in two stages. First, states are obtained from Kalman filter as optimal estimates then optimal control problem is solved with these known states.

In LQG design we followed the Separation Theorem hence first we designed an optimal regulator assuming full-state feedback for linear system then we designed a continuous-time Kalman filter with white noises and known power spectral densities. Lastly we combined optimal regulator and Kalman filter into an optimal compensator which yields an input from estimated state and whence output.

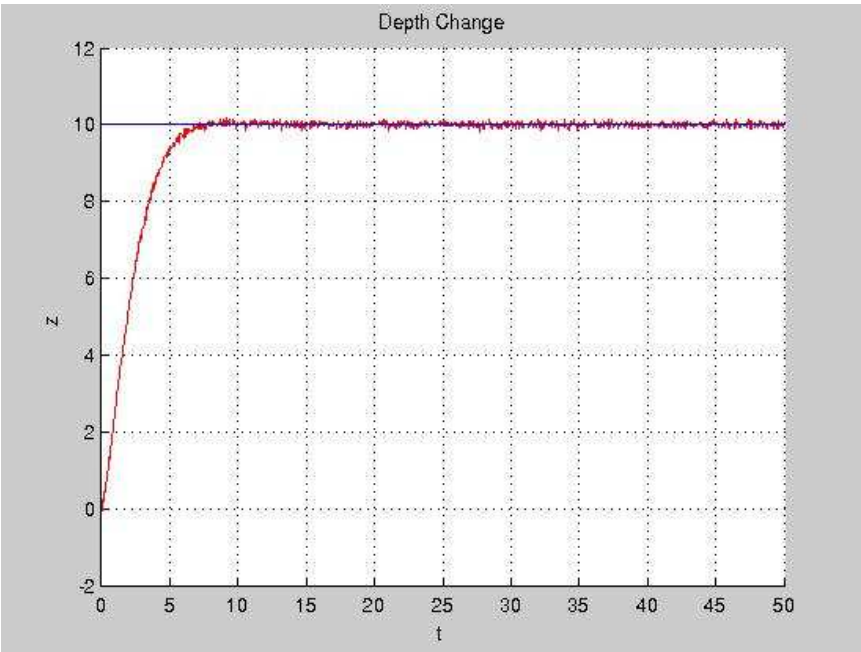

Figure 3.18: Desired Depth with Optimal Control

3.7 Kalman Filter

We start designing the Kalman filter first with a truth model:

$$\dot{x}(t) = A(t)\,x(t) + B(t)\,u(t) + G(t)\,w(t) \tag{3.83}$$

$$y(t) = C(t)\,x(t) + v(t) \tag{3.84}$$

where

x(t) : State vector

y(t) : Measurement vector

A(t) : State matrix

C(t) : Measurement matrix

w(t) : System noise and model uncertainty

v(t) : Measurement noise

Both noises are assumed to be white noises with zero-mean Gaussian and uncorrelated with

each other and with state.

$$E\left\{w(t)\,w^T(t)\right\} = Q(t) \tag{3.85}$$

$$E\left\{v(t)\,v^T(t)\right\} = R(t) \tag{3.86}$$

$$E\left\{v(t)\,w^T(t)\right\} = 0 \tag{3.87}$$

Kalman filter for state and output estimate is build as

$$\dot{\hat{x}}(t) = A(t)\,\hat{x}(t) + B(t)\,u(t) + L(t)\,\left[y - C(t)\,\hat{x}\right] \tag{3.88}$$

Let's define the state error as $e = \hat{x} - x$. Combining the truth model with Kalman form we get

$$\begin{aligned}\dot{e} &= (A(t)\,\hat{x}(t) + B(t)\,u(t) + L(t)\,\left[y - C(t)\,\hat{x}\,\right]) - (A(t)\,x(t) + B(t)\,u(t) + G(t)\,w(t)) \\ &= A(t)\,e + L(t)\,C(t)\,x + L(t)\,v(t) - L(t)\,C(t)\,\hat{x} - G(t)\,w(t) \\ &= (A(t)\,\, - L(t)\,C(t))\,e - G(t)\,w(t) + L(t)\,v(t)\end{aligned} \tag{3.89}$$

We define the state error-covariance matrix as

$$P(t) \equiv E\left\{e(t)\,e^T(t)\right\} \tag{3.90}$$

Considering the linear time-invariant state-space equation

$$\dot{x}(t) = A(t)\,x(t)\, +\, B(t)u(t) \tag{3.91}$$

$$y(t) = C(t)x(t)\, +\, Du(t) \tag{3.92}$$

Using the below formula which is known as *continuous Riccati equation*, continuous time Kalman filter is derived as stated in [19].

$$\begin{aligned}\dot{P}(t) &= A(t)P(t)\, +\, P(t)A^T(t) \\ &\quad - P(t)\,C^T(t)\,R^{-1}(t)\,C(t)\,P(t)\, +\, G(t)\,Q(t)\,G^T(t)\end{aligned} \tag{3.93}$$

The algorithm for continuous time Kalman filter is stated in the table below. Assuming the linear time varying system with white noise sequences and zero-mean Gaussian distributions, first initial values for state and error-covariance matrix should be assigned. Then the Kalman gain matrix is found and used in updating the error-covariance matrix in next step. Here a differential equation is obtained which is solved in Simulink toolbox of MATLAB.

Table 3.1: Continuous Time Kalman Filter

Model	$\dot{x}(t) = A(t)x(t) + B(t)u(t) + G(t)w(t),\ w(t) \sim N(0, Q(t))$ $y(t) = C(t)x(t) + v(t),\ v(t) \sim N(0, R(t))$
Initialize	$\hat{x}(t_0) = \hat{x}_0$ $P_0 = E\left\{e(t_0)e^T(t_0)\right\}$
Gain	$K(t) = P(t)C^(T)(t)R^{-1}(t)$
Covariance	$\dot{P}(t) = A(t)P(t) + P(t)A^T(t)$ $-P(t)C^T(t)R^{-1}(t)C(t)P(t) + G(t)Q(t)G^T(t)$
Estimate	$\dot{\hat{x}}(t) = A(t)\hat{x}(t) + B(t)u(t) + L(t)\left[y - C(t)\hat{x}(t)\right]$

Combining the optimal regulator with Kalman filter we retrieve a compensator. Regulator's weighting matrices **Q** and **R** and Kalman filter's spectral noise densities **V** and **Z** play important role in designing compensator. Using the state information obtained from Kalman filter we get the following LQG equation which we built in Simulink.

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - C\hat{x}) \\ &= (A - BK - LC)\hat{x} + Ly \end{aligned} \tag{3.94}$$

with the input value

$$u = -K\hat{x} \tag{3.95}$$

where $K = PBR^{-1}$.

Our design for LQR/LQG Control is based on separation principle therefore combining each controller on the same Simulink diagram would be more appropriate in order to compare their efficiencies.
Here

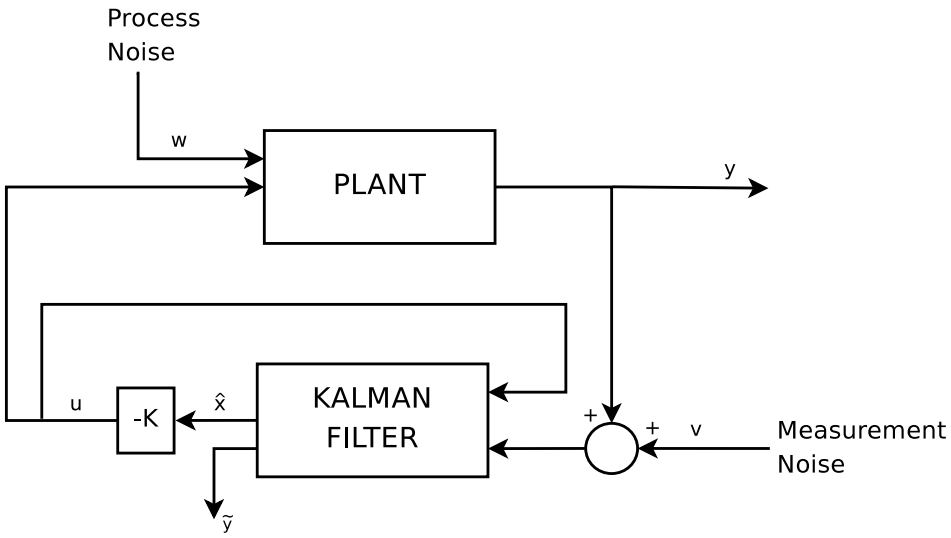

Figure 3.19: LQG Design

3.8 Summary

In this chapter, we tried different control methods and remarked the outputs and successes of each method. Because of its simplicity and easy applicability first we tried PID method. In real world, many successful designs with PID method has been reported [12] recently. On the other hand, when the chaotic specifications are thought of sea conditions, better control methods are developed for underwater systems. SMC is one these control methods with its robustness. Different studies showed that [13], SMC is a robust and effective method to compensate environmental disturbances and unexpected effects. In our algorithms, we have also analyzed that applying same amount of noise to the system, SMC is more successful in compensating these effects when compared with PID and LQR. On the other hand, most of the time because of chattering property of SMC while trying to keep the error on a sliding surface swiftly, extra battery power is needed, which can be a disadvantage of SMC for a long operation.

We assumed that full state feedback information is available while designing PID and SMC methods which is far from the reality. Hence to get rid of this deficiency, we analyzed an

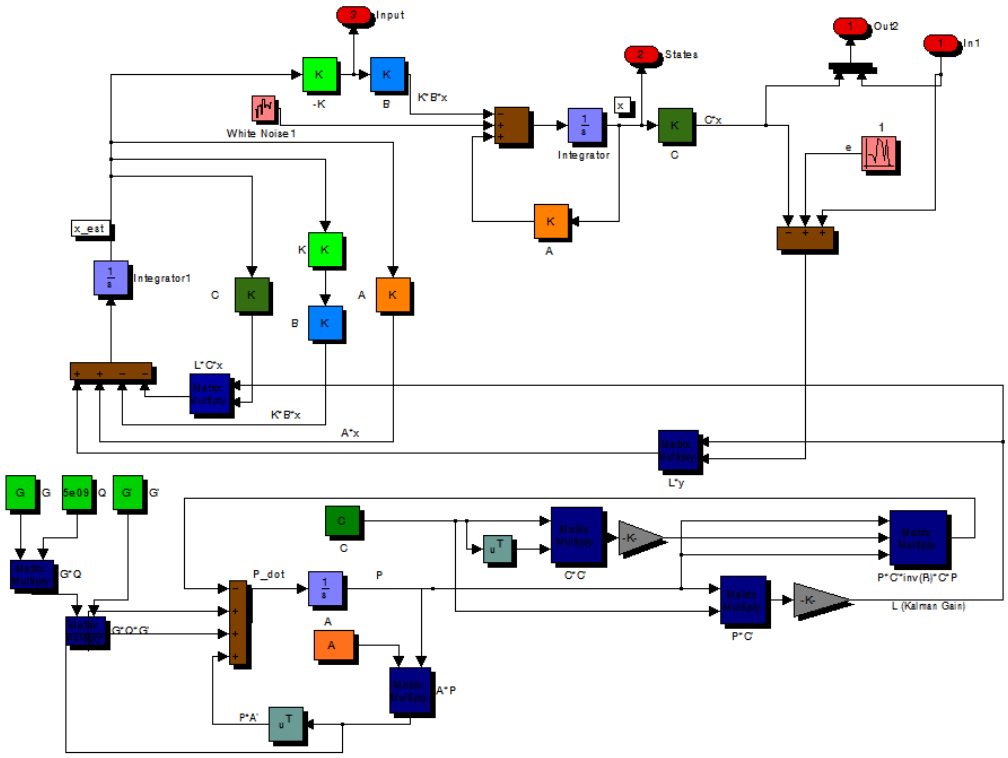

Figure 3.20: Simulink LQG Steer Sub-block

LQG design where the Kalman filter is a prerequisite. In that manner, first we designed an LQR system and combined Kalman filter with it to obtain LQG model. We probed the success of Kalman filter in filtering the undesired noise and also gained the advantage of acquiring the information of some components of state which can not be measured by the sensors in real time operation. Advantage of LQG method over LQR is obvious which has been demonstrated with related figures.

Probing all the methods we generated SMC seems to be most effective control method in harsh sea conditions. Using the state information obtained from Kalman filter will probably increase its efficiency and bring important advantages to the system.

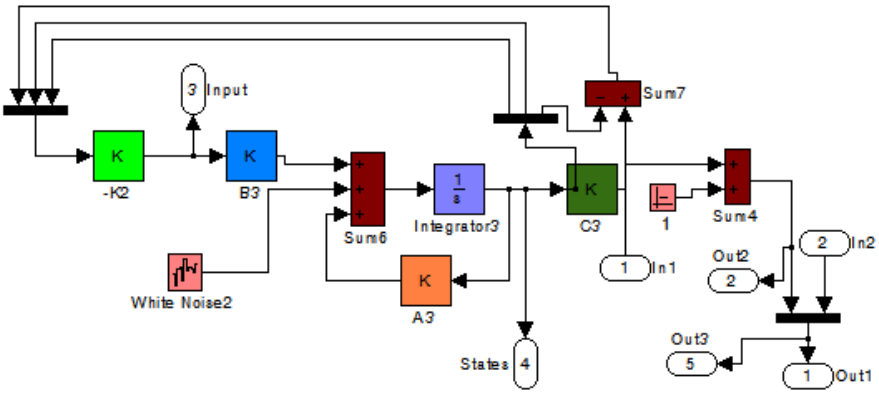

Figure 3.21: Simulink LQR Steer Sub-block

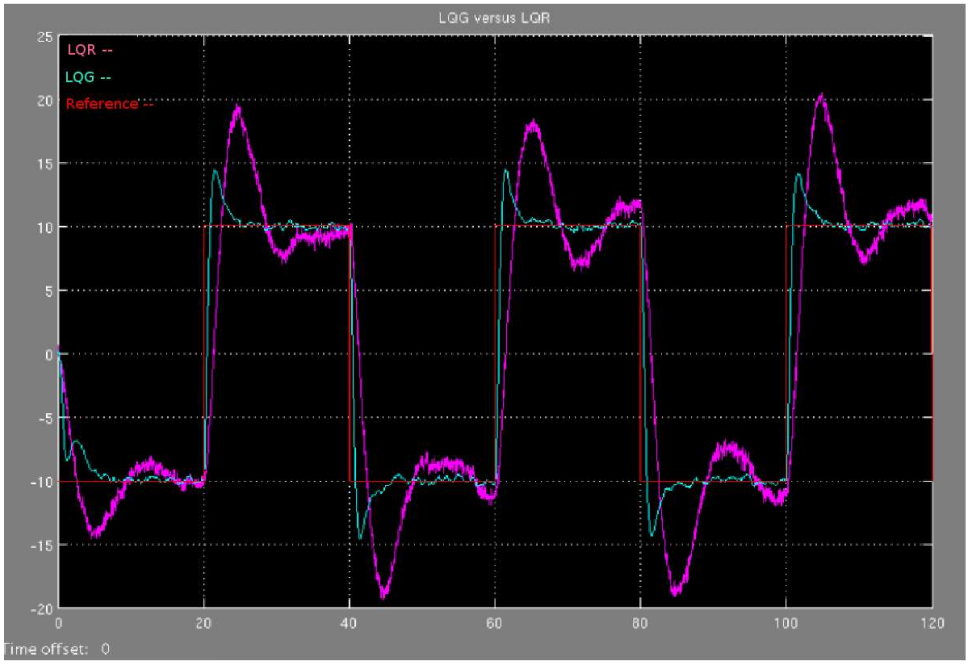

Figure 3.22: LQG vs. LQR

CHAPTER 4

PARAMETER ESTIMATION

4.1 Introduction

In this chapter we worked on the problem of parameter estimation which is directly related with system identification. System identification is defined as the deduction of system characteristics from measured data [18]. It is commonly referred as an inverse problem in a way that one tries to find the input when the output is known. In a general manner what we tried in this chapter is estimating the hydrodynamic coefficients of our vehicle from our measurements which are obtained from controllers.

In Section 4.2, a solution to our parameter estimation problem is probed from window of least squares method. A sequential method that corrects the parameters at each step is used. In Section 4.3, parameter estimation problem is faced from other side and Genetic algorithms as an optimization method is used. Defining a nonlinear cost function for our problem we applied different simulations in order to estimate our coefficients from controllers outputs.

The goal in parameter estimation is to predict the value of a quantity which is assumed to be time invariant. In case of time-variance change of parameters must be slow compared with the state variables in order to estimate the values successfully [35]. In our estimation problem we assumed that all our parameters (hydrodynamic coefficients) are time-invariant. Parameter estimation for underwater vehicle consists of predicting some of the hydrodynamic coefficients by analyzing the measurements, which can be modeled as

$$\tilde{y}(i) = h\left[i, x, v(i)\right], \qquad i = 1, 2, \ldots, k \tag{4.1}$$

where $\tilde{y}(i)$ denotes the measurements, x parameters and $v(i)$ measurement noise. Hence we

must find a function of k observations

$$\hat{x}(k) \triangleq \hat{x}\left[k, Y^k\right] \tag{4.2}$$

here observations are shown as

$$Y^k \triangleq \{\tilde{y}(i)\}_{i=1}^{k} \tag{4.3}$$

The measurement noise $v(i)$ is assumed to be zero mean Gaussian noise with variance σ^2,

$$v(i) \sim N\left(0, \sigma^2\right) \tag{4.4}$$

4.2 Linear Sequential Estimation

Basically linear sequential estimation method is achieved by linear least squares algorithm. In linear least squares, our goal is to estimate the unknown vector x from given measurements. Our problem can be modeled as

$$\tilde{y}_i = H_i\hat{x} + v_i \qquad i = 1, \ldots, k \tag{4.5}$$

where

$\tilde{y}_i$:measurements

H_i :measurement matrix

$\hat{x}$:vector of unknowns

v_i :measurement noise

Our model (4.5) is linear with its nature, if there would be a nonlinear function $h(x(t), t)$ instead of $H(i)$, we would have a nonlinear parameter estimation problem. Here, we try to estimate x such that the estimation error is minimum, hence we grab that solution by minimizing the square of the estimation errors with inserted weight matrix in the cost function,

$$J = \frac{1}{2} e^T W e \tag{4.6}$$

We are looking for $\hat{x}$ that minimizes J. Here W is an $n \times n$ symmetric matrix. e stands for error and defined as

$$e = \tilde{y}_i - H_i\hat{x}$$

Our cost function (4.6) can be written as

$$\begin{aligned} J(k) &= \sum_{i=1}^{k} [\tilde{y}_i - H_i\hat{x}]^T W_i [\tilde{y}_i - H_i\hat{x}] \\ &= \tilde{y}_k^T W_k \tilde{y}_k - \tilde{y}_k^T W_k H_k \hat{x} - \hat{x}^T H_k^T W_k \tilde{y}_k + \hat{x}^T H_k^T W_k H_k \hat{x} \end{aligned} \tag{4.7}$$

Utilizing the necessary condition for $\hat{x}$ to be minimum, which is

$$\frac{\partial J}{\partial \hat{x}} = 0 \tag{4.8}$$

we find

$$\frac{\partial J}{\partial \hat{x}} = -2\tilde{y}_k^T W_k H_k + 2\hat{x}^T H_k^T W_k H_k = 0 \tag{4.9}$$

equating both sides yields

$$\tilde{y}_k^T W_k H_k = \hat{x}^T H_k^T W_k H_k$$

Then taking the transpose of both sides we get

$$H_k^T W_k H_k \hat{x} = H_k^T W_k \tilde{y}_k$$

Lastly leaving $\hat{x}$ we find the least squares solution as

$$\hat{x} = \left(H_k^T W_k H_k\right)^{-1} H_k^T W_k \tilde{y}_k \tag{4.10}$$

From the sufficient condition of minimum, $\frac{\partial^2 J}{\partial \hat{x}^2} > 0$, which means setting the gradient to zero, it is clear in (4.9) that W must be positive definite in order to equation to be positive.

Least squares estimator is unbiased which can be defined as

$$E[e_k] = 0 \implies E[x - \hat{x}_k] = 0 \implies E[\hat{x}_k] = x \tag{4.11}$$

where $e_k \triangleq x - \hat{x}_k$.

An estimator $\hat{x}(y)$ is defined as the unbiased estimator of x if the expected value of the estimator is equal to x, $E\{\hat{x}(y)\} = x$, $\forall x$ [35].

Selecting $W = R^{-1}$ and substituting it in (4.10) we get

$$\begin{aligned} E[\hat{x}_k] &= \left(H_k^T R_k^{-1} H_k\right)^{-1} H_k^T R^{-1} E[\tilde{y}_k] \\ &= \left(H_k^T R_k^{-1} H_k\right)^{-1} H_k^T R^{-1} E[H_k\hat{x} + v_k] \end{aligned} \tag{4.12}$$

Hence estimation error is found as

$$\begin{aligned} e_k &= x - \hat{x}_k \\ &= x - \underbrace{\left(H_k^T R_k^{-1} H_k\right)^{-1} H_k^T R^{-1} E\left[H_k \hat{x}\right.}_{x} + v_k] \\ &= -\left(H_k^T R_k^{-1} H_k\right)^{-1} H_k^T R^{-1} v_k \end{aligned} \tag{4.13}$$

Next we find the information matrix as the expected value of the square of the bias

$$\begin{aligned} P(k) &= E\left[e(k)\, e(k)^T\right] \\ &= (H_k^T R_k^{-1} H_k)^{-1} H_k^T R_k^{-1} \underbrace{v_k v_k^T}_{R_k} R_k^{-1} H_k (H_k^T R_k^{-1} H_k)^{-1} \\ &= H_k^{-1} R_k (H_k^T)^{-1} H_k^T R_k^{-1} R_k R_k^{-1} H_k H_k^{-1} R_k (H_k^T)^{-1} \end{aligned} \tag{4.14}$$

After cancellation we get

$$P(k) = \left[H_k^T R_k^{-1} H_k\right]^{-1} \tag{4.15}$$

Here existence of the inverse means that the information matrix is finite and the eigenvalues of $H_k^T R_k^{-1} H_k$ are directly related with the condition number which is the ratio of the largest singular value to smallest one shows the effort for invertibility. Small condition number denotes that the matrix can be easily inverted, on the contrary large condition number defines poor invertibility or nearly singular matrices.

Substituting equation (4.15) in (4.10), our least squares solution simplifies to

$$\hat{x} = P_k H_k^T R_k^{-1} \tilde{y}_k \tag{4.16}$$

Taking the inverse of both sides and writing for $k+1$ our information matrix transforms into

$$P_{k+1}^{-1} = H_{k+1}^T R_{k+1}^{-1} H_{k+1} \tag{4.17}$$

In order to write a sequential form we define the variables partitioned as

$$y_{k+1} = \begin{bmatrix} y_k \\ y(k+1) \end{bmatrix} \tag{4.18}$$

$$H_{k+1} = \begin{bmatrix} H_k \\ H(k+1) \end{bmatrix} \tag{4.19}$$

$$v_{k+1} = \begin{bmatrix} v_k \\ v(k+1) \end{bmatrix} \tag{4.20}$$

$$R_{k+1} = \begin{bmatrix} R_k & 0 \\ 0 & R(k+1) \end{bmatrix} \tag{4.21}$$

which will bring the advantage of writing our information matrix in a sequential form

$$\begin{aligned} P_{k+1}^{-1} &= \begin{bmatrix} H_k^T & H(k+1)^T \end{bmatrix} \begin{bmatrix} R_k & 0 \\ 0 & R(k+1) \end{bmatrix}^{-1} \begin{bmatrix} H_k \\ H(k+1) \end{bmatrix} \\ &= \begin{bmatrix} H_k^T R_k^{-1} & H(k+1)^T R(k+1)^{-1} \end{bmatrix} \\ &= H_k^T R_k^{-1} H_k + H(k+1)^T R(k+1)^{-1} H(k+1) \end{aligned} \tag{4.22}$$

Hence it can be written in a more compact form as

$$P_{k+1}^{-1} = P_k^{-1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1} \tag{4.23}$$

Equation (4.23) means that information gained at the $k+1$ step is the sum of information at k step and the new information about x that is obtained from measurement $\tilde{y}(k+1)$. Owing to write the estimation in a more calculation accordant, we will write the information matrix in a more compact form by using the matrix inversion lemma [7] which states that

$$F = [A + BCD]^{-1} \tag{4.24}$$

here

$$\begin{aligned} F &= n \times n \,\text{matrix} \\ A &= n \times n \,\text{matrix} \\ B &= n \times m \,\text{matrix} \\ C &= m \times m \,\text{matrix} \\ D &= m \times n \,\text{matrix} \end{aligned}$$

If we have the assumption that all parts have inverses, then we can write the above inverse as

$$F = A^{-1} - A^{-1} B \left(DA^{-1} B + C^{-1} \right)^{-1} DA^{-1} \tag{4.25}$$

In our sequential algorithm selecting the components of information matrix as they adapt in the matrix inversion lemma

$$\begin{aligned} F &= P_{k+1} \\ A &= P_k^{-1} \\ B &= H_{k+1}^T \\ C &= R_{k+1} \\ D &= H_{k+1} \end{aligned}$$

and placing them in the matrix inverse lemma yields

$$\begin{aligned} P_{k+1} &= \left[P_k^{-1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1}\right]^{-1} \\ &= P_k - P_k H_{k+1}^T \left(H_{k+1} P_k H_{k+1}^T + R_{k+1}^{-1}\right)^{-1} H_{k+1} P_k \end{aligned} \tag{4.26}$$

In order to write the above equation more briefly we define new variables as

$$S_{k+1} = H_{k+1} P_k H_{k+1}^T + R_{k+1} \tag{4.27}$$

$$W_{k+1} = P_k H_{k+1}^T S_{k+1}^{-1} \tag{4.28}$$

hence we can write information matrix recursion

$$P_{k+1} = \left[I - W_{k+1} H_{k+1}\right] P_k \tag{4.29}$$

which is equal to

$$P_{k+1} = P_k - W_{k+1} S_{k+1} W_{k+1}^T \tag{4.30}$$

The estimation formula can be written in a recursive form with the assumption that all the inverses exist

$$\hat{x}_k = \underbrace{\left(H_k^T R_k^{-1} H_k\right)}_{P_k} H_k^T R_k^{-1} \tilde{y}_k \tag{4.31}$$

for $k+1$ it is written as

$$\begin{aligned} \hat{x}_{k+1} &= P_{k+1} H_{k+1}^T R_{k+1}^{-1} \tilde{y}_{k+1} \\ &= P_{k+1} \begin{bmatrix} H_k^T & H(k+1)^T \end{bmatrix} \begin{bmatrix} R_k & 0 \\ 0 & R(k+1) \end{bmatrix}^{-1} \begin{bmatrix} \tilde{y}_k \\ \tilde{y}(k+1) \end{bmatrix} \\ &= P_{k+1} H_k^T R_k^{-1} \tilde{y}_k + P_{k+1} H_{k+1}^T R_{k+1}^{-1} \tilde{y}_{k+1} \end{aligned} \tag{4.32}$$

when we substitute P_{k+1} defined in (4.30) for the first term on the right-hand side in (4.32)

$$\begin{aligned} \hat{x}_{k+1} &= P_k H_k^T R_k^{-1} \tilde{y}_k - W_{k+1} H_{k+1} P_k H_k^T R_k^{-1} \tilde{y}_k + \underbrace{P_{k+1} H_{k+1}^T R_{k+1}^{-1}}_{W_{k+1}} \tilde{y}_{k+1} \\ &= \left[I - W_{k+1} H_{k+1}\right] \underbrace{P_k H_k^T R_k^{-1} \tilde{y}_k}_{\hat{x}_k} + W_{k+1} \tilde{y}_{k+1} \end{aligned} \tag{4.33}$$

which yields a more simplified form when we write it as

$$\begin{aligned} \hat{x}_{k+1} &= \hat{x}_k - W_{k+1} H_{k+1} \hat{x}_k + W_{k+1} \tilde{y}_{k+1} \\ &= \hat{x}_k - W_{k+1} \left[\tilde{y}_{k+1} - H_{k+1} \hat{x}_k\right] \end{aligned} \tag{4.34}$$

The recursion form of linear sequential estimation is shown in Table (4.1).

Table 4.1: Linear Sequential Estimation

Initialize	$P_1 = \left[\frac{1}{\alpha^2} I + H_1^T W_1 H_1\right]^{-1}$ $\hat{x}_1 = P_1 \left[\frac{1}{\alpha} \beta + H_1^T W_1 \tilde{y}_1\right]^{-1}$
Update	$\hat{x}_{k+1} = \hat{x}_k + K_{k+1} (\tilde{y}_{k+1} - H_{k+1} \hat{x}_k)$
Gain	$K_{k+1} = P_k H_{k+1}^T \left[H_{k+1} P_k H_{k+1}^T + W_{k+1}^{-1}\right]^{-1}$
Information Matrix	$P_{k+1} = [I - K_{k+1} H_{k+1}] P_k$

4.2.1 Steering Parameter Estimation

Applying the above sequential algorithm to the simplified linearized steering motion equations stated below, we find the estimated parameters.

$$\begin{bmatrix} m - Y_{\dot{v}} & -Y_{\dot{r}} \\ -N_{\dot{v}} & (I_z - N_{\dot{r}}) \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} Y_v & mu - Y_r \\ N_v & N_r \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ \tau_N \end{bmatrix} \tag{4.35}$$

In order to reduce the coupling effect of the parameters occurring because of the multiplication we omit the parameters $Y_{\dot{r}}$ and $N_{\dot{v}}$ [3]. Leaving the derivatives on the left side alone

$$(m - Y_{\dot{v}})\, \dot{v} = Y_v v + (Y_r - mU)\, r$$

$$(I_z - N_{\dot{r}})\, \dot{r} = N_v v + N_r r N_\tau \tau \tag{4.36}$$

$$\tag{4.37}$$

we define the derivatives as

$$\dot{v} = \underbrace{\left(\frac{Y_v}{m - Y_{\dot{v}}}\right)}_{\delta} v + \underbrace{\left(\frac{Y_r - mU}{m - Y_{\dot{v}}}\right)}_{\varepsilon} r$$

$$\dot{r} = \underbrace{\left(\frac{N_v}{I_z - N_{\dot{r}}}\right)}_{\alpha} v + \underbrace{\left(\frac{N_r}{I_z - N_{\dot{r}}}\right)}_{\beta} r + \underbrace{\left(\frac{N_\tau}{I_z - N_{\dot{r}}}\right)}_{\gamma} \tau \tag{4.38}$$

$$\tag{4.39}$$

arranging the equations to apply the algorithm we get

$$\frac{v_{t+1}-v_t}{\Delta t} = \delta v + \varepsilon r \qquad \Longrightarrow \qquad v_{t+1} = (\delta v\,\Delta t + 1)\,v + \varepsilon\,\Delta t\,r \tag{4.40}$$

which can be written as

$$v_{t+1} = \underbrace{\begin{bmatrix} v & r \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} (\delta v\,\Delta t + 1) \\ \varepsilon\,\Delta t \end{bmatrix}}_{\Theta} \tag{4.41}$$

4.2.2 Diving Parameter Estimation

Before using the sequential parameter estimation algorithm we arrange our diving model and configure it in order to make it applicable to obtain unknown parameters. Hence simplifying the model we get

$$\begin{bmatrix} m - Z_{\dot{\omega}} & -Z_{\dot{q}} \\ -M_{\dot{\omega}} & (I_y - M_{\dot{q}}) \end{bmatrix} \begin{bmatrix} \dot{\omega} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_{\omega} & mU - Z_q \\ M_{\omega} & M_q \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} + \begin{bmatrix} \tau_Z \\ \tau_M \end{bmatrix} \tag{4.42}$$

then leaving the derivatives alone yields

$$\begin{aligned} \dot{\omega} &= \left(\frac{Z_{\omega}}{m - Z_{\dot{\omega}}}\right) w + \left(\frac{Z_q - mU}{m - Z_{\dot{\omega}}}\right) q + \left(\frac{\tau_Z}{m - Z_{\dot{\omega}}}\right) u \\ \dot{q} &= \left(\frac{M_{\omega}}{I_y - M_{\dot{q}}}\right) w + \left(\frac{M_q}{I_y - M_{\dot{q}}}\right) q + \left(\frac{\tau_M}{I_y - M_{\dot{q}}}\right) u \end{aligned} \tag{4.43}$$

Evaluating the sequential algorithm both for steering and diving motions we find the estimated linear hydrodynamic coefficients and errors as

Table 4.2: Parameter Estimation via LSE

Parameters	Initial Value	Estimated Value	Absolute Error	Percent Error
Y_v	-0.0510	-0.0310	0.0200	%39.21
Y_r	0.0300	0.0280	0.0020	% 6.66
N_v	-0.0074	-0.0077	0.0003	% 3.67
N_r	-0.0160	-0.0035	0.0195	%78.12
Z_ω	-0.3000	-0.2144	0.0856	%28.52
Z_q	-0.1400	0.1038	0.2438	%174.15
M_ω	-0.0029	-0.0027	0.0002	% 8.47
M_q	-0.0016	-0.0017	0.0001	% 6.44

4.3 Parameter Estimation via Genetic Algorithm

Genetic Algorithm (GA) is an optimization and search method based on the rules of genetics and natural selection [9]. Goal of the genetic algorithms is to select a state that maximizes the fitness which is equal to minimizing the cost function. Fitness is directly related with the survivability, which is wanted to maximize.

Since GA is based on genetics, some terms used in problem solution are to be defined first. A *gene* is the basic unit of heredity which includes the sequence of enzymes called *genetic code*. This code does not vary in an organism. GA is solved by using chromosomes which breed to form new generation. This group of interbreeding individuals is called *population*. Genes are found in two forms that show different characteristics. Each of these forms is known as *allele*. The combination of the alleles influence the traits of the organism. One of the alleles becomes dominant and the other one becomes recessive by which the genetic code is transferred to next generations in a natural way.

Though many different algorithms are developed which are somewhat superior to other mates, general advantages of GA can be stated as [9]

- No need of derivation or gradient of cost, which can be a burden job for a complex cost function.
- Generates a list of optimum variables, not just a single one.

- GA can be applied to both continuous and discrete variables.
- Evaluation of a large number of variables is possible and it not a big issue for GA.
- Efficient for parallel computers.
- Variables of the problem can be encoded, bringing the advantage of running optimization with these encoded variables.
- GA can run with numerically generated data, experimental data or analytical functions.

Hydrodynamic coefficients of the system are small and must be precisely defined enough to run in an optimization problem to minimize the cost effectively. It will be a burdensome job to define our small coefficients in binary GA therefore we preferred to use real-coded GA, which is sometimes called continuous GA.

Figure (4.2) gives a summary of the real-coded GA. The main difference of real-coded GA from binary coded GA is that instead of zeros and ones the genes are represented with floating-numbers between defined range.

The GA process starts with fitting the variables to a chromosome therefore selection of the chromosome plays an important role. The chromosome is written as an array of $1 \times N_{var}$ elements where N_{var} shows the number of the variables in the problem which is equal to the number of genes in a chromosome. Hence a chromosome can be stated as

$$chromosome = [\, p_1, p_2, p_3, \ldots p_{N_{var}} \,] \tag{4.44}$$

At each step after generation of new chromosome, a cost is calculated with respect to the variables $p_1, p_2, p_3, \ldots p_{N_{var}}$.

$$cost = f(chromosome) = f(p_1, p_2, p_3, \ldots p_{N_{var}}) \tag{4.45}$$

GA is an search technique therefore to avoid too much computation effort, a reasonable region must be searched. This is achieved by constraining the genes between upper and lower boundaries. If the region of interest is not known at the beginning, a region with enough diversity for GA to find the solution must be selected.

Before starting the algorithm an initial population of N_{pop} is selected where N_{pop} shows the number of population or number of chromosomes. Once the population is generated, one will have a matrix of $N_{pop} \times N_{var}$. Generally for the ease of calculation the values are normalized

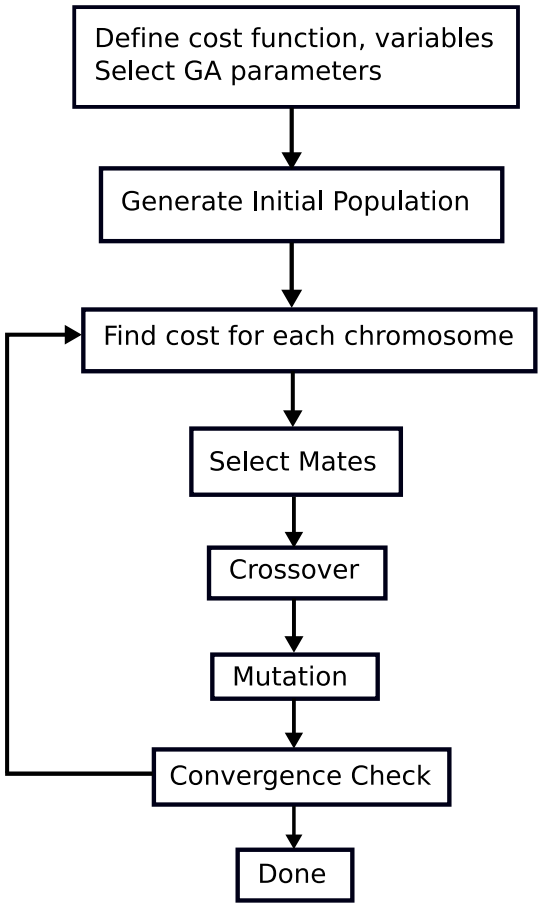

Figure 4.1: Real-Coded Genetic Algorithm Flowchart

in selection phase but during evaluation, unnormalized values are used in the cost function. Unnormalized values of the chromosomes can be selected as [9]

$$p = (p_{hi} - p_{lo})\, p_{norm} + p_{lo} \tag{4.46}$$

where

p_{hi} = upper boundary for the variables in a chromosome
p_{lo} = lower boundary for the variables in a chromosome
p_{norm} = normalized values of the variables in a chromosome

Normalization of the variables (V) can be achieved by

$$V_{norm} = \frac{V - V_{min}}{V_{max} - V_{min}} \qquad , V_{norm} \in [0,1] \tag{4.47}$$

Next all the chromosomes are evaluated. In GA, the chromosomes with high fitnesses are selected to reproduce offsprings for the next generation. A pairing algorithm is applied here to select the chromosomes for crossover phase. For pairing process we preferred roulette wheel weighting. Here the chromosomes have probabilities which are inversely proportional with their costs, chromosome with lowest cost will have maximum probability and inversely chromosome with highest cost will have minimum probability because of the fact that we are looking for the minimum. A normalized cost is calculated for each chromosome by subtracting the cost of best chromosome from eliminated group which satisfies that costs of the chromosomes in our population are all negative. Then the costs are normalized by the following formulae

$$P_n = \left| \frac{C_n}{\sum_m^{N_{keep}} C_m} \right| \tag{4.48}$$

By this pairing method selection probability of the chromosomes with low costs are greater than ones with high costs. Therefore tendency towards the minimum cost becomes faster, however if the costs of the chromosomes are close each other this method weights nearly even.

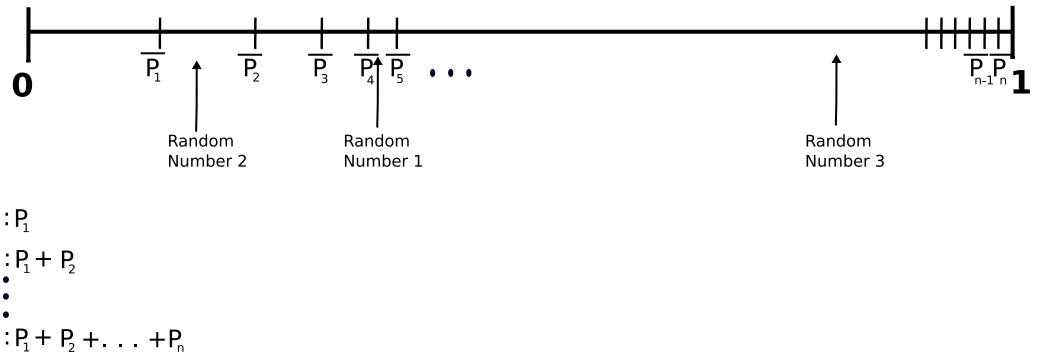

Figure 4.2: Roulette Wheel like selection

Using roulette wheel selection all the chromosomes are selected by two pairs. Now they are ready for the crossover operation to breed next generation. Crossover operation is achieved as

$$ch_1 = [\quad ch_{11} \quad | \quad ch_{12} \quad] \qquad ch_2 = [\quad ch_{21} \quad | \quad ch_{22} \quad] \tag{4.49}$$

$$off_1 = [\quad off_{11} \quad | \quad off_{12} \quad] \qquad off_2 = [\quad off_{21} \quad | \quad off_{22} \quad] \tag{4.50}$$

Here, off states the new offsprings. A number between zero and one is selected, $\alpha \in [0,1]$.

Separation in variables is achieved by selecting a random number between one and N_{var}. The components of offsprings are defined as

$$\begin{aligned} off_{11} &= \alpha \; ch_{11} + (1-\alpha)ch_{21} \\ off_{12} &= (1-\alpha)ch_{12} + \alpha \; ch_{22} \\ off_{21} &= (1-\alpha)ch_{11} + \alpha \; ch_{21} \\ off_{22} &= \alpha \; ch_{12} + (1-\alpha)ch_{22} \end{aligned}$$

Then we apply mutation operation. Sometimes GA may converge too quickly into one region and if this region comprises local minimums algorithm generally ends in one of the local minimums. In order to avoid this tendency, new regions are introduced to algorithm for searching which is achieved by mutation in the variables. Though is shows difference due to the type of the problem a mutation rate of % 5 - %20 are applicable. We preferred %10 as mutation rate. Number of mutations is found by $N_{Mut} = \mu_{Rate} \times (N_{pop} - 1) \times N_{var}$ where N_{Mut} stands for the number of mutations and N_{Rate} defines mutation rate. Easiest way of mutation is achieved by adding a normally distributed random number to the variable selected for mutation which can be shown as

$$p_{new} = p_{old} + \sigma N(0,1) \tag{4.51}$$

where σ shows standart deviation of the normal distribution N (0,1) is the standart normal distribution with mean 0, and variance 1. Selection of the standart deviation plays an important role here because of the real values that stay between zero and one. Selecting a big deviation number just pushes the selected values to boundaries.

At this phase the chromosomes that will pass to next generation will be selected. Due to the elitism the optimum chromosome with minimum cost is not included for selection. Hence $N_{pop} - 3$ chromosomes will be keep for the next generation. One free space will be used for the elitist chromosome and other two spaces will be the boundary chromosomes which will be obtained after applying boundary mutation. Boundary mutation will be applied to two randomly selected chromosomes. Generally mutating %40 of a chromosome will be enough. Selecting a random number between zero and one, if the random number is smaller than 0.5 than selected genes will be mutated to lower boundary otherwise they will be made equal to upper boundary of each gene.

Algorithm will run until the desired convergence rate is reached. Analyzing the convergence rate of GA with Markov chains, studies showed that large population size and low mutation rate leads to solution more efficiently [9]. Alternative ways to stop the algorithm if it did not converged can be stated as

- If the best chromosome of GA does not change after *n* iterations then stop. Algorithm found a solution or it stucked in a local minimum.
- If the standart deviation and mean of the population's cost reached a level then stop the algorithm because the chromosomes will no longer change.
- Algorithm may found the correct answer, check whether the solution is compared with the best chromosome not others.
- If algorithm does not stop for one of the reasons above, then limit the number of iterations.

After a number iterations if the algorithm does not converge to a good solution changing the population size and mutation rate may remedy the problem.

4.3.1 Parameter Retrieval via Genetic Algorithm

In parameter estimation via GA we used a black box which is based on measurement data [21]. Applying a sequence of input to a unknown system we obtained an output. By the measurements from output we solved a back problem. All the simulations were generated in Simulink.

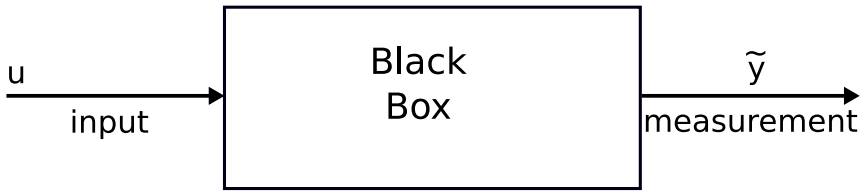

Figure 4.3: Parameter Estimation Procedure

We started parameter estimation process first by constructing cost function. Since our mathematical model is in form

$$M\dot{x} = Ax + Bu + v \tag{4.52}$$

Better evaluation of the model can be achieved by writing our model as

$$\dot{x} = M^{-1}Ax + M^{-1}Bu + v \tag{4.53}$$

where v denotes noise. Since our goal is to minimize the error, we can define our cost as

$$Cost = \| \dot{x} - M^{-1}Ax + M^{-1}Bu \|^2 \tag{4.54}$$

Hence running GA with our measurements, the following solutions for steering and depth damping hydrodynamic coefficients are obtained.

Table 4.3: Steering Parameters found by Genetic Algorithm (After 192 Steps)

Parameters	Initial Value	Estimated Value	Absolute Error	Percent Error
Y_v	-0.05100	-0.05102	0.00002	% 0.039
Y_r	0.03000	0.02203	0.00797	%26.566
N_v	-0.00740	-0.00739	0.00001	% 0.135
N_r	-0.01600	-0.01798	0.00198	%12.375
$Y_{\dot{v}}$	-0.03330	-0.02853	0.00477	%14.324
$Y_{\dot{r}}$	-0.01380	-0.01845	0.00465	%33.695
$N_{\dot{v}}$	0.01380	0.01233	0.00147	%10.652
$N_{\dot{r}}$	-0.00920	-0.00948	0.00028	% 3.043

4.4 Summary

In this chapter, the goal was to estimate the hydrodynamic coefficients from system measurements. We tried two different methods. First we applied linear sequential estimation which is a recursive parameter estimation method. This method calculates a new update for the parameter vector each time new data comes in. Computation time has to be constant for each parameter calculation therefore it is easily applicable for real time applications. Basic idea of this method is to calculate the new parameter estimate at time k by adding some correction vector to previous parameter estimate [21].

Next we used genetic algorithms for parameter estimation. Because of the system and measurement noises, output of the system does not directly give information about the structure of the system. Therefore we tried to estimate the system by analyzing measurements such that, minimizing the errors would give us maximum information. Solution of the parameter

Table 4.4: Depth Parameters found by Genetic Algorithm (After 513 Steps)

Parameters	Initial Value	Estimated Value	Absolute Error	Percent Error
Z_w	-0.30000	-0.30027	0.00027	% 0.09
Z_q	-0.14000	0.16066	0.02066	%14.757
M_w	-0.00290	-0.00292	0.00002	% 0.689
M_q	-0.01600	-0.01669	0.00069	% 4.312
$Z_{\dot{w}}$	-0.03330	-0.03148	0.00178	% 5.351
$Z_{\dot{q}}$	-0.01380	-0.01340	0.00040	% 2.898
$M_{\dot{w}}$	-0.01380	0.01418	0.00038	% 2.753
$M_{\dot{q}}$	-0.00463	-0.00448	0.00015	% 3.239

estimation problem with GA ended with satisfying results, where estimation errors of some of the parameters are far beyond from errors found with linear sequential estimation. The success of GA comes from the fact that it is an optimization search method which directly concentrates on the solution for our case.

But when the applicabilities of both methods are considered, using linear sequential estimation for online studies and GA for offline estimation seems to be more logical. LSE runs in a short time when compared with GA therefore that makes it preferable for online estimation but because of its unsatisfactory results using GA for offline estimation will give good results.

CHAPTER 5

GUIDANCE, PATH PLANNING AND OBSTACLE AVOIDANCE

5.1 Introduction

In this chapter we described guidance and obstacle avoidance with path planning. Section 5.2 gives information about path planning and how to keep our underwater vehicle far from obstacles while not moving away from goal point. For path planning we preferred to solve online problem which seems to be more applicable for real time situations. We assumed that at the start of the operation, no information about the locations of the obstacles are available. As our vehicle advances towards the goal point, obstacles start to give echo that is when the information about obstacles are obtained. Obstacles are modeled using constructive solid geometry and path planning as an optimization problem is solved both with sequential quadratic programming (SQP) and Fletcher-Reeves methods. Solutions about algorithms are expressed within graphs at the end of the section.

In Section 5.3, information about guidance methods in literature is given. Then, as a simple practical approach line of sight guidance method is described, which is the most applied guidance algorithm in underwater environment. Generating some random numbers, efficiency of our Simulink guidance model is tested. Solutions are presented at the end of the section.

5.2 Path Planning and Obstacle Avoidance

Path planning is an essential requirement in different conditions especially in military operations for underwater vehicles. When time and energy come forth as important issues, path planning becomes an indispensable method in order to consume less energy because of the limited battery capacity. Selecting the environment as two dimensional space brings an important advantage where in 3D, necessity to define the position and attitude information increases the number of the variables hence problem becomes more complex for online solving.

For path planning we assumed that the problem will be online obstacle avoidance where safety of underwater vehicle is the first priority in our condition. Goal is to reach the global minimum point and avoiding from local minimums. In order to achieve a successful operation, space free from obstacles is considered as a set of inequalities for nonlinear programming problem [31]. Our study was based on the potential field theory where the vehicle is represented as a point under the influence of an artificial field and area around the obstacles show collision region. Therefore the potential function will be the sum of effects that will push the vehicle from the obstacles and pull it towards to the goal point. Due to the fact that potential field approach can not avoid the vehicle from local minimums, combining the constructive solid geometry (CSG) with it local minimums can be avoided. Detailed information about CSG can be found in [6].

Three possibilities exist in object defining procedure for CSG. A solid **S** defined in 3D Euclidian space, set of its interior points are shown by **I**, points on the boundary are denoted by **B** and points that are out of boundary are represented with **T** with bringing the fact

$$\mathbf{I} \cup \mathbf{B} \cup \mathbf{T} = \mathbf{E}^3 \tag{5.1}$$

A non-negative continuous function $f(c)$ defines a solid in $\mathbf{E}^3$ where intersection and union operations of *n* objects can be defined respectively as

$$f^I(c) = max(f_1(c), f_2(c), \ldots, f_n(c)) \tag{5.2}$$

and union can be defined as

$$f^U(c) = min(f_1(c), f_2(c), \ldots, f_n(c)) \tag{5.3}$$

Assuming the operation condition a cluttered area we modeled our obstacles as circles with different diameters in 2D. A circle in CSG can be modeled as

$$(x-a)^2 + (y-b)^2 = R^2 \tag{5.4}$$

where R denotes the diameter of the circle and a, b are distances from main axes. Hence our free space becomes

$$R^2 - (x-a)^2 - (y-b)^2 \leq 0 \tag{5.5}$$

In calculations for the optimum path, SQP (Sequential Quadratic Programming) method is used. General properties of SQP can be denoted as

- It is one of the most widely used algorithms for nonlinear constrained optimization.
- The Karush-Kuhn-Tucker (KKT) conditions are enforced in an iterative manner.
- An approximate Quadratic Programming sub-problem is solved at each major iteration (The Quadratic Programming problem solved at each iteration of SQP is an approximation of the original problem with linear constraints and quadratic objective).
- The solution to the QP problem gives a search direction.
- Using the search direction a line search is carried out.
- At each major iteration an approximation of the Hessian is updated using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method.

The algorithm that we applied for solving the problem and detailed information can be found in [16].

In order to test the efficiency of the algorithm we checked if it is capable of passing through the narrow the gaps and analyzed that algorithm can converge the goal point successfully in such conditions. At this test the objective function is selected as

$$f(x,y) = (x-4)^2 + (y-4)^2 \tag{5.6}$$

and the obstacles are modeled according to CSG as

$$x^2 + y^2 - 8.988 \leq 0 \tag{5.7}$$
$$x^2 + (y-4)^2 - 1 \leq 0 \tag{5.8}$$
$$x^2 + (y+4)^2 - 1 \leq 0 \tag{5.9}$$
$$(x-4)^2 + y^2 - 1 \leq 0 \tag{5.10}$$
$$(x+4)^2 + y^2 - 1 \leq 0 \tag{5.11}$$

Solution for this case is shown in Figure (5.1).

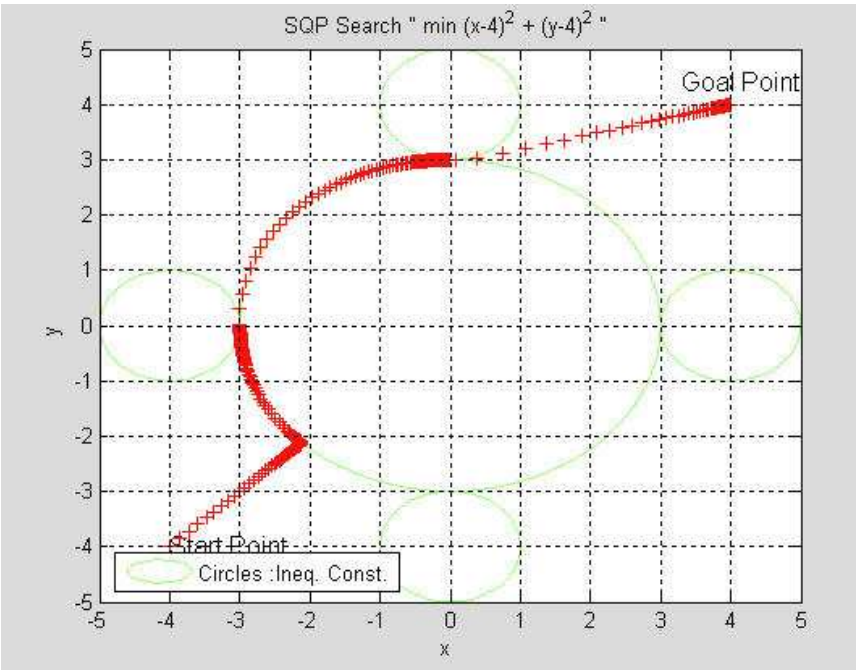

Figure 5.1: Algorithm Passing Through Narrow Gaps

We tried the algorithm with different starting points, in order to set the most difficult scenario we selected the goal point as $[4,4]$ and starting point $[-7,-8]$. Then we realized that algorithm converged to local minimum and stuck at that point with the following obstacle mapping.

$$x^2+y^2-1\leq 0 \tag{5.12}$$

$$x^2+(y-4)^2-1\leq 0 \tag{5.13}$$

$$x^2+(y+4)^2-9\leq 0 \tag{5.14}$$

$$(x-4)^2+y^2-1\leq 0 \tag{5.15}$$

$$(x+4)^2+y^2-9\leq 0 \tag{5.16}$$

Solution for that mapping is shown in Figure (5.2). Here solution leads us to the local minimum.

After changing the start point and shift it from the region, where converging chance to local minimum is high, algorithm managed to reach the goal point. Vehicle path, reaching the goal point is shown in Figure (5.3).

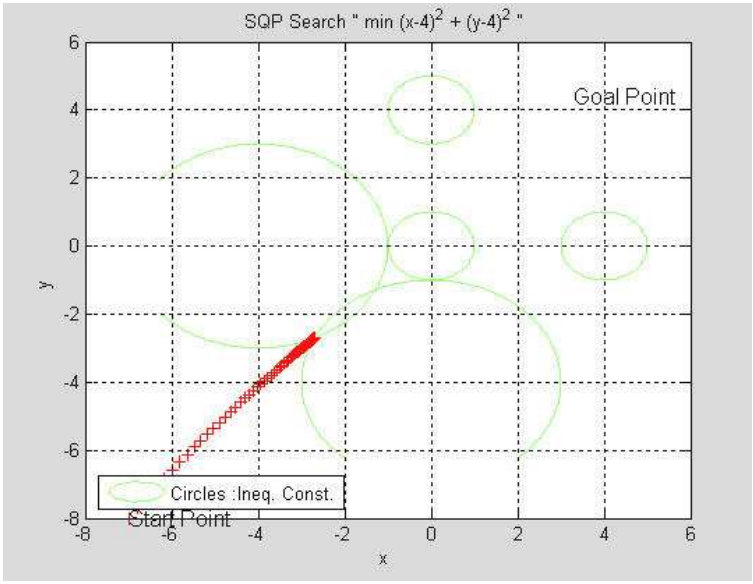

Figure 5.2: Converging Algorithm to Local Minimum

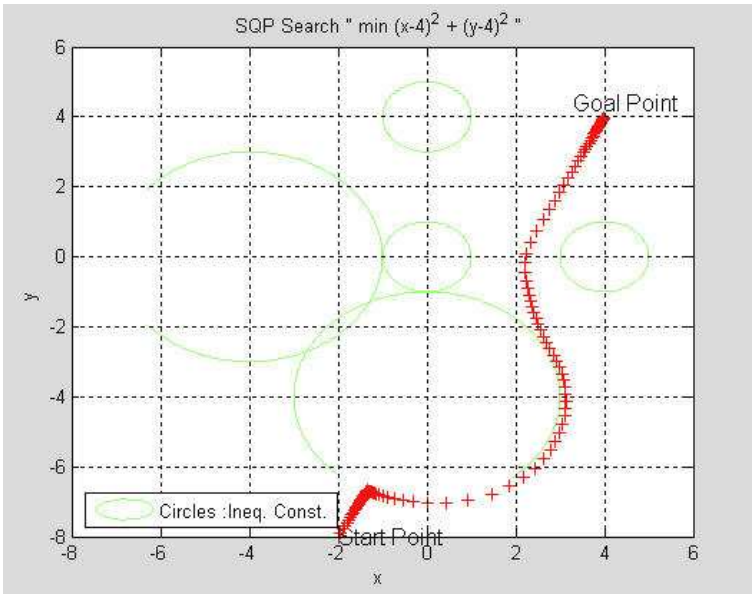

Figure 5.3: Converging Algorithm to Goal Point

In order to enhance the algorithm, using the same obstacles we added intersection of obstacles as a new constraint to the system therefore algorithm managed to avoid from obstacles without sticking in local minimum. Algorithm avoiding from the local minimum is shown in Figure (5.4) and Figure (5.5).

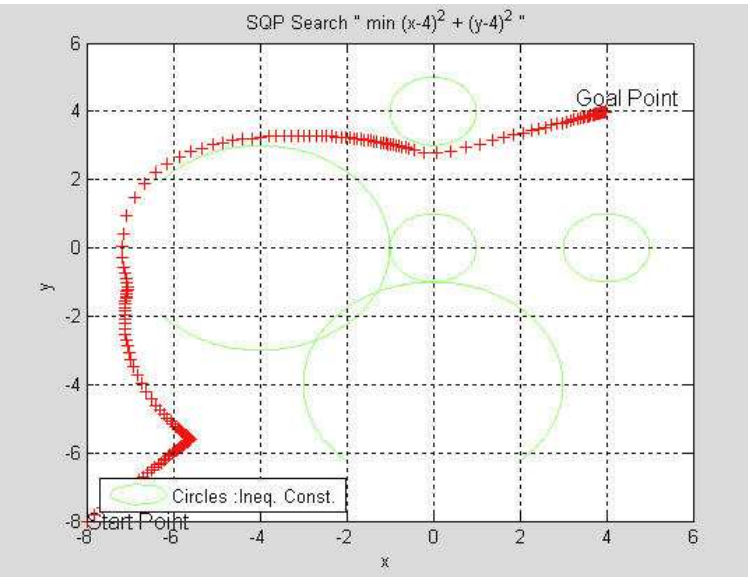

Figure 5.4: Solution Avoiding The Local Minimum

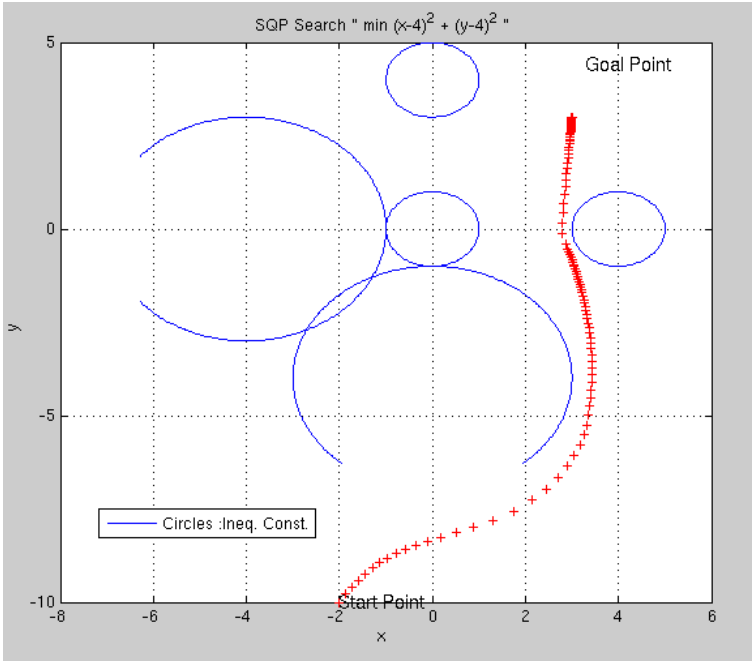

Figure 5.5: Solution Reaching The Goal Point

This path planning problem is also solved using Fletcher-Reeves method with random generated obstacles. In this solution the goal is to increase the distance between obstacles and the vehicle in order to avoid endangering the operation. Here a weighting cost function is defined which fulfills increasing the cost when the vehicle approaches obstacles and decreases the cost when the vehicle goes away from obstacles. Since the target is reaching the goal point, while avoiding from obstacles, a weighting function is used which balances the efficiency of the algorithm in reaching the goal point and keeping distance from obstacles.

The cost function with five constraints, used in calculations is stated below.

$$f(x,y) = 0.4(x-4)^2 + 0.4(y-4)^2 + 0.6\Big(\frac{1}{\sqrt{(x-x_{obs1})^2+(y-y_{obs1})^2}-R_{obs1}} + \frac{1}{\sqrt{(x-x_{obs2})^2+(y-y_{obs2})^2}-R_{obs2}} + \frac{1}{\sqrt{(x-x_{obs3})^2+(y-y_{obs3})^2}-R_{obs3}} + \frac{1}{\sqrt{(x-x_{obs4})^2+(y-y_{obs4})^2}-R_{obs4}} + \frac{1}{\sqrt{(x-x_{obs5})^2+(y-y_{obs5})^2}-R_{obs5}}\Big) \tag{5.17}$$

The path through the random generated obstacles is shown in Figure (5.6).

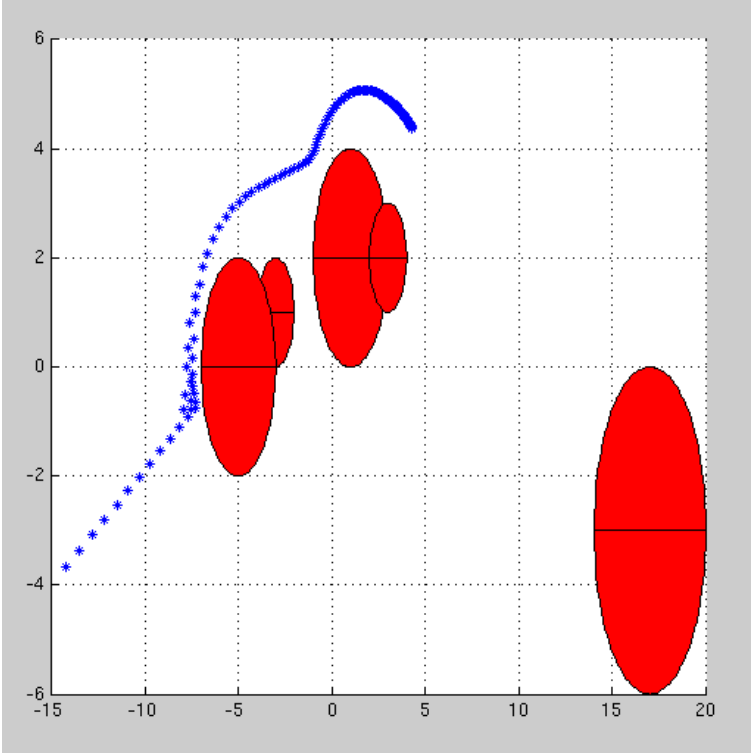

Figure 5.6: Algorithm Through Random Obstacles

When compared with Fletcher-Reeves method, it is seen that the BFGS method solves the problem in less steps therefore in a shorter time.

5.3 Guidance

Autonomy for an underwater vehicle requires the design of a proper guidance system. For a safe operation, vehicle needs a reliable Navigation, Guidance and Control (NGC) system, where guidance is the dominant element. With the success of NGC systems after 50s in aerospace technology, underwater navigation systems found an impetus by simulating systems designed for missiles and rockets.

NGC systems on-board of an underwater vehicles work in a harmony with other components. Navigation system provides the information of the target, then guidance system evaluates this information and calculates the heading angle which becomes the input for the controller. Controller keeps the vehicle on the heading angle that comes from guidance system. For a remotely operated vehicle, operator plays the mission of a guidance system and sends the heading angles directly to the controller [30]. When the definition of the guidance is defined as determining the course, attitude and speed of the vehicle, its main duty can be denoted as deciding the best trajectory to be followed by the vehicle based on target location and vehicle capability. Since sensing, information processing and correction are the main tasks that enhance the efficiency of a guidance system, the position of the target with respect to the vehicle and environmental conditions directly effects the accuracy. Though the capability of a guidance system can be measured with its computation and sensing power, its efficiency will be limited directly with the vehicle dynamics and actuator limitations.

Generally in literature the following guidance methods are preferred most of which are adapted from advances from aerospace.

- Waypoint Guidance by Line of Sight (LOS)
- Lyapunov Based Guidance
- Vision Based Guidance
- Proportional Navigation Guidance
- Guidance by Chemical Signals
- Guidance via Magnetometers for Cable Tracking
- Electromagnetic Guidance

Our vehicle operating near sea bottom will not have the opportunity of knowing its location precisely. She will get the position information of the surface vehicle and will try to correct

this value by acoustic transducers and sonar. Since the bottom vehicle will conceive the depth information with bearing and distance from the surface vehicle, she will be capable of estimating her own position by these information. Though it is more logical to use guidance system for the bottom vehicle, for our case it will be used on surface vehicle because of the precise position information that she will obtain from *Navstar GPS* (Global Positioning System). With the improvements in last decade, accuracy of the GPS system enhanced significantly. Error of a normal GPS system between 10m to 100m is decreased to 5 cm with carrier-phase differential GPS (CDGPS).

As mentioned in Chapter 3 output of our steering and depth controllers are degrees where the operator introduces to the system as the reference. But in real time operations especially in pipeline tracking and mine counter measure operations, operator does not interest how the system turns or dives. In those circumstances the path, vehicle follows or the points, that vehicle has to pass through come into prominence. In order to lift the responsibility of achieving such tasks from operator, a guidance system has to be designed for underwater vehicles. In those systems it is assumed that the vehicle can advance with constant forward speed. Desired routes generally are represented by waypoints. Adding the environmental conditions wind, current and wave informations, efficiency can be increased.

In order to define the position of the vehicle first the kinematics must be described. Since the operation is on 2D it is assumed that $\theta = \phi = 0$. From the general kinematics equations we have the following equations.

$$\dot{x} = u\cos\psi - v\sin\psi \tag{5.18}$$

$$\dot{y} = u\sin\psi + v\cos\psi \tag{5.19}$$

$$\dot{\psi} = r \tag{5.20}$$

Since the above equations are nonlinear in the states, by applying a linear approximation that the earth-fixed coordinate system can be rotated such that desired heading is $\psi_d = 0$ and moving the origin of the coordinate system to the initial point, our heading angle ψ will be small where we can make the assumption,

$$\sin\psi \approx \psi; \qquad\qquad \cos\psi \approx 1 \tag{5.21}$$

In the linearization process perturbing equation (5.18) yields

$$\dot{x}_0 + \Delta\dot{x} = (u_0 + \Delta u)\cos(\psi_0 + \Delta\psi) - (v_0 + \Delta v)\sin(\psi_0 + \Delta\psi) \tag{5.22}$$

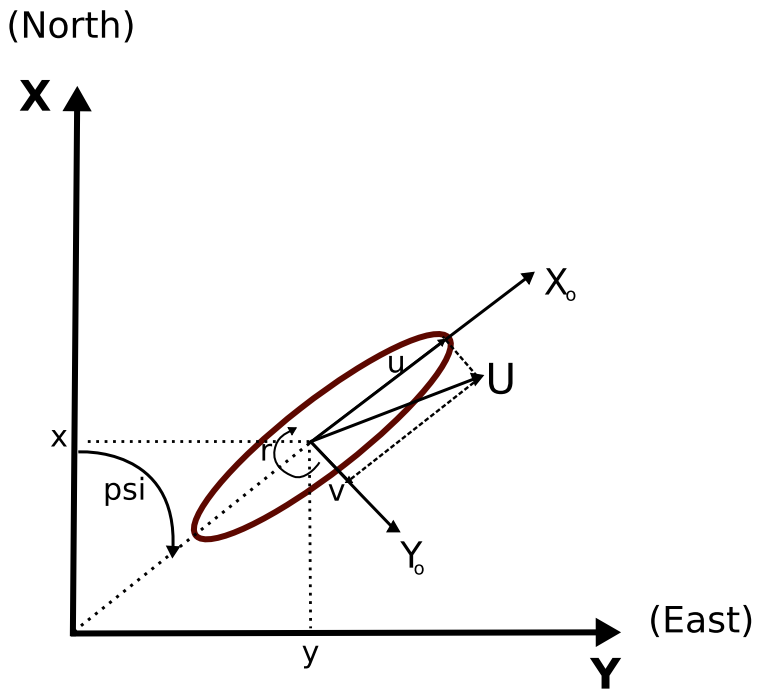

Figure 5.7: Coordinate system for Guidance

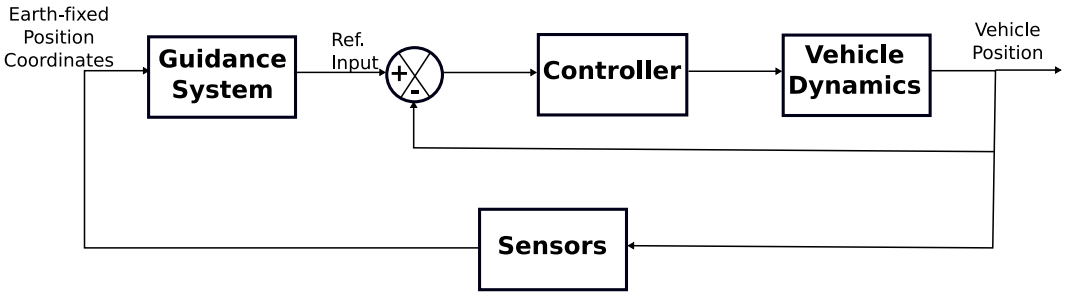

Figure 5.8: Guidance and Control System

Trigonometric functions stated above can be extended as

$$cos\,(\psi_0 + \Delta\psi) = cos\,\psi_0\,cos\,\Delta\psi - sin\,\psi_0\,sin\,\Delta\psi \tag{5.23}$$

$$sin\,(\psi_0 + \Delta\psi) = sin\,\psi_0\,cos\,\Delta\psi + sin\,\Delta\psi\,cos\,\psi_0 \tag{5.24}$$

Using the simplifications stated in (5.21) trigonometric equations simplify to

$$cos\,(\psi_0+\Delta\psi) \approx cos\,\psi_0 - \Delta\psi sin\,\psi_0 \tag{5.25}$$

$$sin\,(\psi_0+\Delta\psi) \approx sin\,\psi_0 + \Delta\psi cos\,\psi_0 \tag{5.26}$$

Substituting initial condition, $\dot{x}_0 = u_0\,cos\,\psi_0 - v_0\,sin\,\psi_0$ in the left hand side of (5.22) and simplified trigonometric functions in the right hand side yields

$$\begin{aligned} u_0\,cos\,\psi_0 - v_0\,sin\,\psi_0 + \Delta\dot{x} = u_0\,cos\,\psi_0 - (u_0\,sin\,\psi_0)\,\Delta\psi + \Delta u\,cos\,\psi_0 - v_0\,sin\,\psi_0 \\ - (v_0\,cos\,\psi_0)\,\Delta\psi - \Delta v\,sin\,\psi_0 \end{aligned} \tag{5.27}$$

$$\Delta\dot{x} = \Delta u\,cos\,\psi_0 - \Delta v\,sin\,\psi_0 - (u_0\,sin\,\psi_0)\,\Delta\psi - (v_0\,cos\,\psi_0)\,\Delta\psi \tag{5.28}$$

As a fact of linearization getting rid of Δ in (5.28) gives

$$\dot{x} = u\,cos\,\psi_0 - v\,sin\,\psi_0 - (u_0\,sin\,\psi_0)\,\Delta\psi - (v_0\,cos\,\psi_0)\,\Delta\psi \tag{5.29}$$

Applying the same procedure to perturb y in Earth-fixed frame we get the following equations.

$$\dot{y} = u\,sin\,\psi + v\,cos\,\psi \tag{5.30}$$

then perturbation yields

$$\dot{y}_0 + \Delta\dot{y} = (u_0+\Delta u)\,sin\,(\psi_0+\Delta\psi) \quad + \quad (v_0+\Delta v)\,cos\,(\psi_0+\Delta\psi) \tag{5.31}$$

Substituting initial conditions and after elimination gives

$$\Delta\dot{y} = \Delta u\,sin\,\psi_0 + \Delta v\,cos\,\psi_0 + (u_0\,cos\,\psi_0)\,\Delta\psi - (v_0\,sin\,\psi_0)\,\Delta\psi) \tag{5.32}$$

which is equal to

$$\dot{y} = u\,sin\,\psi_0 + v\,cos\,\psi_0 + (u_0\,cos\,\psi_0)\,\Delta\psi - (v_0\,sin\,\psi_0)\,\Delta\psi \tag{5.33}$$

5.3.1 Line of Sight (LOS) Guidance

When an operation path is defined with respect to the waypoints, line of sight guidance is an efficient method. Here the desired waypoints $[x_d(k), y_d(k)]$ are entered to the system and the vehicle is guided with the goal of bringing the vehicle near the points. Guidance system generates an angle based output to the input for the controller which is defined as

$$\psi_d(t) = atan\left(\frac{y_d(t)-y(t)}{x_d(t)-x(t)}\right) \tag{5.34}$$

Because of the nature of tangent, quadrant of the angle must be taken into consideration in the calculations. To check the proximity of vehicle to the waypoints a measure of error must be defined. When the vehicle enters in the defined proximity of the waypoint, new goal must be selected as the next waypoint. Here measure of error can be defined as

$$\sqrt{(x_d(t) - x(t))^2 + (y_d(t) - y(t))^2} \leq \tilde{e} \tag{5.35}$$

Though it may be defined by the importance of the mission, in most cases measure of guidance error $\tilde{e}$ is taken to be not more than two vehicle lengths, $\tilde{e} \leq 2L$.

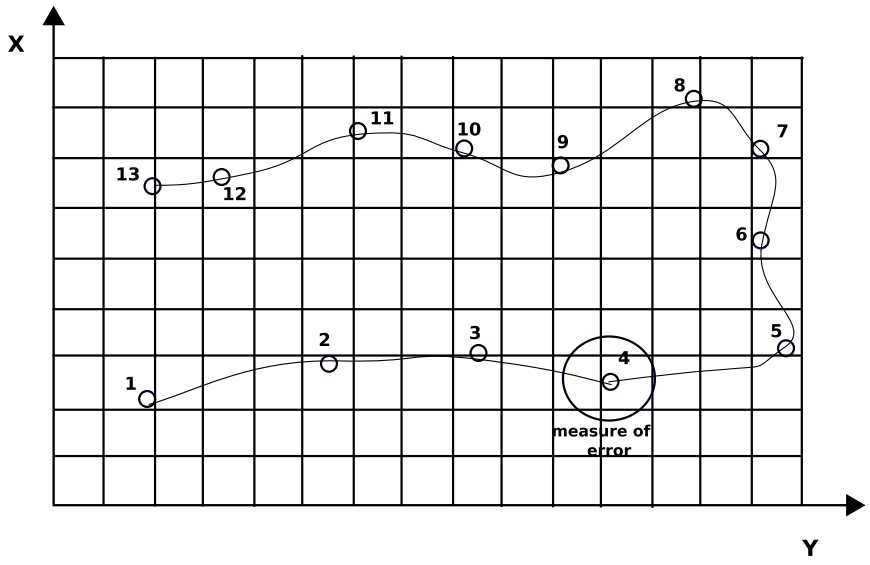

Figure 5.9: Line of Sight Guidance

We used Simulink for modeling the guidance system. In LOS guidance, a series of points which represent the waypoints are generated then these points are introduced to guidance system. Guidance system calculates the distance between these points and generates a steering angle for the controller. Controller imports that steering angle and commands the system to get closer to the points consecutively. After each steer angle update to the controller, guidance system checks whether the proximity condition is satisfied. When the system enters in the measure of the proximity circle, guidance system refers to the next waypoint and procedure goes on till to the last waypoint. Simulink diagram of our model is shown in Appendix D.

The evaluation of our algorithm is achieved by first selecting 12 random points then contin-

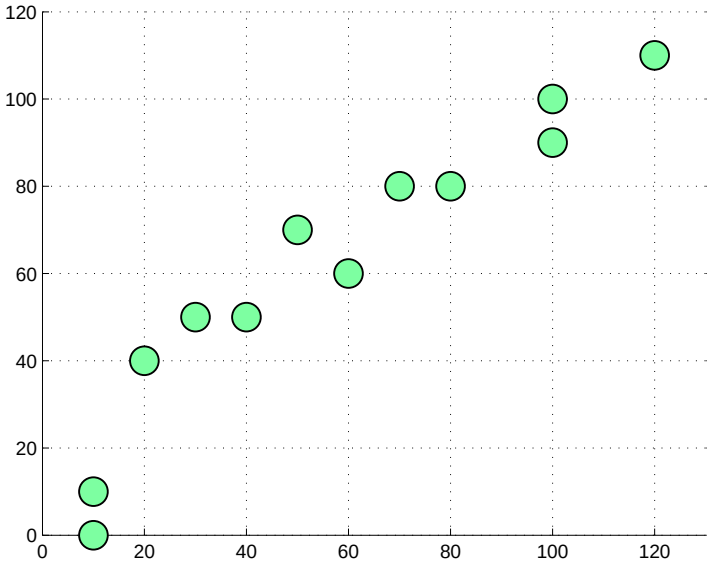

Figure 5.10: Waypoints

ues with calculating errors and steering angles between each consecutive waypoints. These errors and steering angles are imposed to the simulink model. Guidance unit checks the proximity of errors with respect to the defined value, which is two times length of our vehicle in our case. Instead of checking for the entrance of vehicle to the acceptance circle, we probed errors till they show increase attitude after decreasing. We applied the guidance correcting steering angle to the vehicle till error decreases. When the error starts to increase guidance systems directs to the next point to the controller and system starts to turn to the new steering angle. With the above logic, the path that our guidance system followed is shown in Figure (5.11). Some waypoints are not covered because of the early commands that guidance system injects to the controller.

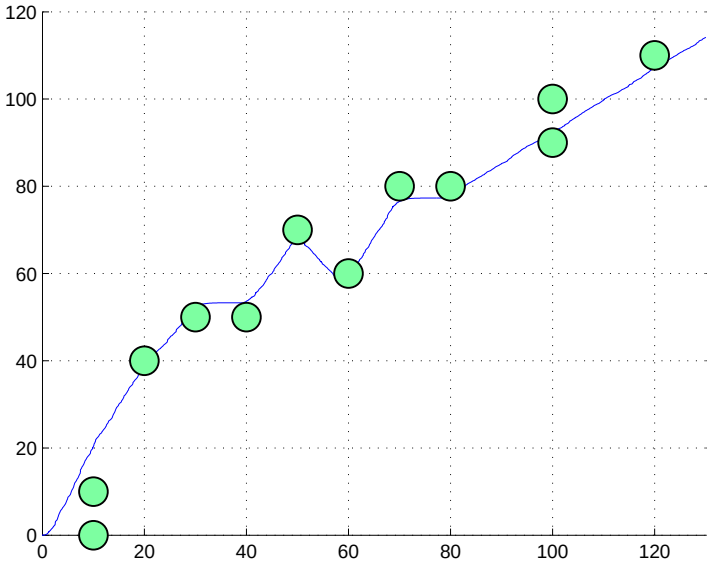

Figure 5.11: Path Generated by Guidance System

5.3.2 Lyapunov Based Guidance

This method is directly based on Lyapunov theory which states that

If,

1. $V(x) > 0$ (positive definite)
2. $\dot{V}(x) < 0$ (negative definite)
3. $V(x) \longrightarrow \infty$ as $\|x\| \longrightarrow \infty$ (radially unbounded) then the equilibrium point x^* satisfying $f(x^*) = 0$ is *globally asymptotic stable*, which also means that $\|x\| \longrightarrow 0$ as $t \longrightarrow \infty$.

A new law is generated where the vehicle is planned to move from a start point to a goal point with desired heading. A Lyapunov function is selected defining the distance between two points and by selecting appropriate velocities to Lyapunov function, the Lyapunov theory is satisfied. The function simulating the error converges to zero as time increases.

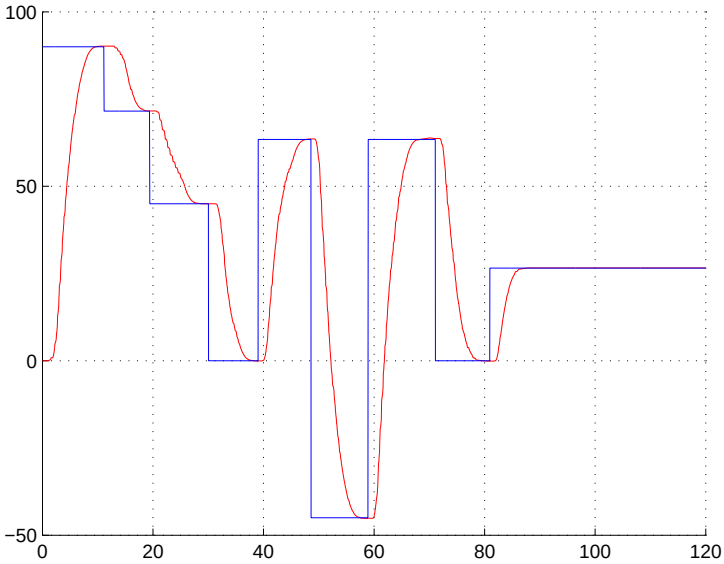

Figure 5.12: Steer Angles by Guidance and Response of Controller

5.3.3 Vision Based Guidance

Generally vision based guidance is used for cable tracking and docking problems. This method is applied with different equipments. For docking problems a beacon has been used. Beacon is used as light emitter which is detected by the photo detectors on the underwater vehicle. After acquisition of light, detectors feed guidance system and heading angle is generated. This is similar with procedure used in heat seeking missiles [25]. A high sun spot for the cameras becomes a huge disadvantage and source of error for shallow waters.

Another manner used in vision based guidance is achieved by evaluating images that are taken from two different cameras [4]. Images that are taken from different cameras are processed by correlating the features and direction to each feature and range is determined. Then the pixel disparity of the images are found by calibration and the direction and range to each feature is filled up to the guidance system to determine the next step. Also the similar process with one camera is developed [2].

Also a laser based light tracker system is developed for guidance where the light is processed with respect to the center of screen. If it is not detected at first glance, all the pixels are

scanned row by row. In situation of detection, its angle and elevation is evaluated and send to the vision processor. More than one light source makes the system invalid.

5.3.4 Proportional Navigation Guidance (PNG)

Basically generated for missiles, PNG method is also applicable to underwater vehicles. This method is useful especially for docking purposes because for stationary targets, method is similar to LOS guidance. It is modeled as

$$U_c = N V_c \lambda \tag{5.36}$$

where N is tuning parameter, V_c is closing velocity, λ is LOS angle and U_c is the command input [30].

5.3.5 Guidance by Chemical Signals

This guidance method is used for the vehicles that mimics sea creatures and these systems are called biomimics. One of the vehicles for this purpose mimics a lobster and tries to identify the location of a chemical discharge by sensing it with its conductivity sensors.

5.3.6 Guidance via Magnetometers for Cable Tracking

As it is defined with its name, this method's main goal is to efficiently track cables for underwater vehicles. Though for unburied cables vision based guidance can be efficient, in case of buried cables to protect them from fishery and anchors, performance of visual guidance decreases seriously. Hence to move out the disadvantages of other guidance methods, magnetometers are used to sense the cables and information from sensors are fed to the guidance system to track these cables successfully.

5.3.7 Electromagnetic Guidance

This method is comprised of magnetic coils on the dock and on-board of vehicle. When the environmental conditions are harsh with a high turbidity and low light, most of the guidance methods become inapplicable. Electromagnetic guidance rectifies this problem by running in every condition from shallow waters to very depth of oceans. It is a very accurate system but its range is limited with a distance of 25m-30m [17].

CHAPTER 6

CONCLUSION

6.1 Summary of the Results

After the improvements in robotics many different underwater vehicles were built in last decades. Requirements especially in military environments accelerates the underwater technology towards autonomous underwater vehicles. Since the interference from operator is minimum in autonomous vehicles, importance of the precise controllers and guidance systems plays an important role. Therefore we focused on the different controllers and probed for the efficient one. Because of determining the hydrodynamic coefficients during pool tests becomes a wearing process, we tried to estimate some of the coefficients from our measurements and applied parameter estimation techniques. Again when autonomy for a vehicle is considered, obstacle avoidance and path planning is an inevitable component for a successful navigation, therefore utilizing the advantages of constructive solid geometry we used an online path planning algorithm. Last of all we generated a guidance system based on line of sight principle.

Since the conclusions of the studies are given at the end of each chapter, an extra effort is not taken to show them again.

6.2 Discussion and Future Work

Considering the richness of the underwater research environment, still there exists many unexplored areas. First of all since ULISAR underwater vehicle is in construction phase, simulations achieved in this study will be applied to our vehicle as soon as she will be ready

for sea deployment.

On the other hand for an efficient navigation, a design for optimum thrust for different motions should be designed for our vehicle. Since we have two thrusters for each motion in different planes, a solution is to be found in order to use batteries optimum because when the motion is left to operator's initiative there exist infinitely many possibilities to execute a command.

Also for autonomy of the vehicle before mission planning, assuming that boundaries of the operation area are totally known a path that will cover all the operation area can be generated based on sonar and other sensors.

BIBLIOGRAPHY

[1] S. Cody A. J. Healey, S. M. Rock and J. P. Brown. Toward an improved understanding of thruster dynamics for underwater vehicles. *IEEE Journal of Oceanic Engineering*, 20(4):354–361, 1995.

[2] A. Balasuriya and T. Ura. Autonomous target tracking by underwater robots based on vision. In *Proceedings of IEEE UT 98*, pages 191–197, 1998.

[3] Nicolas Binois. Modeling and estimation for morpheus dynamics. Master's thesis, Florida Atlantic University, 2003.

[4] D. Wettergreen C. Gaskett and A. Zelinsky. Autonomous guidance and control for an underwater robotic vehicle. *International Conference on Field and Service Robotics*, 1999.

[5] M. Caccia and G. Veruggio. Guidance and control of a reconfigurable unmanned underwater vehicle. In *Control Engineering Practice 8*, pages 21–37, 2000.

[6] H. Chiyokura. *Solid modeling with designbase*. Addison Wesley Publishing Limited,New York, 1988.

[7] J. L. Crassidis and J. L. Junkins. *Optimal Estimation of Dynamic Systems*. CRC Press, 2004.

[8] T. I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley & Sons, 1994.

[9] R. L. Haupt and S. E. Haupt. *Practical Genetic Algorithms*. Wiley-Interscience, second edition, 2004.

[10] A. J. Healey and D. Lienard. Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering*, 18(3):327–339, 1993.

[11] K. Kim J. Kim and H. S. Choi. Depth and heading control for autonomous underwater vehicle using estimated hydrodynamic coefficients. *Department of Naval Architecture, Seoul National University, Korea*, pages 429–435.

[12] B. Jalving. The ndre-auv flight control system. *IEEE Journal of Oceanic Engineering*, 19(4):497–501, 1994.

[13] K. Kim and H. S. Choi. Navigation and control for a test bed auv-snuuv i. *IEEE*, (1):89–94, 2004.

[14] D. E. Kirk. *Optimal Control Theory: An Introduction*. Dover Publications, 2004.

[15] Sir H. Lamb. *Hydrodynamics*. Dover Publications, 1945.

[16] H. D. Sherali M. Bazaraa and C. M. Shetty. *Nonlinear Programming:Theory and Algorithms*. Wiley-Interscience, 2006.

[17] P. R. Blankinship M. D. Feezor, F. Y. Sorrell and J. G. Bellingham. Autonomous underwater vehicle homing/docking via electromagnetic guidance. *IEEE Journal of Oceanic Engineering*, 26(4):515–521, 2001.

[18] R. E. Maine and K. W. Iliff. *Identification of Dynamic Systems*. NASA Reference Publication 1138, 1985.

[19] D. S. Naidu. *Optimal Control Systems*. CRC Press, 2003.

[20] Y. Nakamura and S. Savant. Nonlinear tracking control of autonomous underwater vehicles. In *Proceedings of IEEE Int. Conf. of Robotics and Automation*, volume 3, pages A4–A9, 1992.

[21] O. Nelles. *Nonlinear System Identification*. Springer, 2000.

[22] J. N. Newman. *Marine Hydrodynamics*. The MIT Press, third edition, 1980.

[23] F. A. Papoulias R. Cristi and A. J. Healey. Adaptive sliding mode control of autonomous underwater vehicles in the dive plane. *IEEE Journal of Oceanic Engineering*, 15(3):152–160, 1990.

[24] R. Allen R. K. Lea and S. L. Merry. A comparative study of control techniques for an underwater vehicle. *International Journal of Systems Science*, 30(9):347–964, 1999.

[25] S. Cowen S. Briest and J. Dombrowski. Underwater docking of autonomous undersea vehicles using optical terminal guidance. In *MTS/IEEE Conference Proceedings*, volume 2, pages 1143–1147, 1997.

[26] C. Silpa-Anan, S. Abdallah, and D. Wettergreen. Development of autonomous underwater vehicle towards visual servo control. In *Department of Naval Architecture, Seoul National University, Korea*.

[27] J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[28] S. S. Tabai, F. El-Hawary, and M. El-Hawary. Hybrid adaptive control of autonomous underwater vehicles. In *Proceedings of Symposium of Autonomous Underwater Vehicle Technology*, pages 275–282, 1994.

[29] A. Tewari. *Modern Control Design with Matlab and Simulink*. John Wiley & Sons, 2002.

[30] S. M. Ahmad W. Naeem, R. Sutton and R. S. Burns. A review of guidance laws applicable to unmanned underwater vehicles. *Marine and Industrial Dynamic Analysis Group, The University of Plymouth, UK*.

[31] Y. Wang and D. M. Lane. Subsea vehicle path planning using nonlinear programming and constructive solid geometry. In *IEE Proc. Control Theory Appl. Vol. 144*, pages 143–152, 1997.

[32] K. Wendel. *Hydrodynamic Masses and Hydrodynamic Moments of Inertia*. Technical report, 1956.

[33] L. L. Whitcomb and D. R. Yoerger. Development, comparison and preliminary experimental validation of nonlinear dynamic thruster models. *IEEE Journal of Oceanic Engineering*, 24(4):481–494, 1999.

[34] D. Whitley. A genetic algorithm tutorial. In *Computer Science Department, Colorado State University*.

[35] X. R. Li Y. Bar-Shalom and T. Kirubarajan. *Estimation with Applications To Tracking and Navigation*. John Wiley & Sons, 2001.

[36] D. N. Yoerger and J. E. Slotine. Robust trajectory control of underwater vehicles. *IEEE Journal of Oceanic Engineering*, 10(4):462–470, 1985.

[37] J. Yuh. Design and control of autonomous underwater robots: A survey. In *Automatic Robots 8, 7-24, Kluwer Academic Publishers*, 2000.

APPENDIX A

NONLINEAR EQUATIONS OF MOTION

Nonlinear equations of motion for our vehicle:

$$
\begin{aligned}
\tau_X &= m\dot{u} - mvr + m\omega q + mz_G(pr + \dot{q}) + X_{\dot{u}}\dot{u} + X_{\dot{\omega}}(\dot{\omega} + uq) + X_{\dot{q}}\dot{q} + Z_{\dot{\omega}}\omega q \\
&\quad + Z_{\dot{q}}q^2 + X_{\dot{v}}\dot{v} + X_{\dot{p}}\dot{p} + X_{\dot{r}}\dot{r} - Y_{\dot{v}}vr - Y_{\dot{p}}rp - Y_{\dot{r}}r^2 - X_{\dot{v}}ur - Y_{\dot{\omega}}\omega r \\
&\quad + Y_{\dot{\omega}}vq + Z_{\dot{q}}pq - (Y_{\dot{q}-Z_{\dot{r}}})qr \\
&\quad - X_u u - X_{|u|u}|u|u + X_v v + X_\omega \omega + X_p p + X_q q + X_r r + (W - B)s\theta \\
\tau_Y &= m\dot{v} - m\omega p + mur + mz_G(qr - \dot{p}) + X_{\dot{v}}\dot{u} + Y_{\dot{\omega}}\dot{w} + Y_{\dot{q}}\dot{q} + Y_{\dot{v}}\dot{v} \\
&\quad + Y_{\dot{p}}\dot{p} + Y_{\dot{r}}\dot{r} + X_{\dot{v}}vr - Y_{\dot{\omega}}vp + X_r r^2 + (X_{\dot{p}} - Z_{\dot{r}})rp - Z_{\dot{p}}p^2 - X_{\dot{\omega}}(up - \omega r) \\
&\quad + X_{\dot{u}}ur - Z_{\dot{\omega}}\omega p - Z_{\dot{q}}pq + X_{\dot{q}}qr \\
&\quad - Y_v v - Y_{|v|v}|v|v + Y_u u + Y_\omega \omega + Y_p p + Y_q q + Y_r r - (W - B)c\theta s\phi \\
\tau_Z &= m\dot{\omega} - muq + mvp - mz_G(p^2 + q^2) + X_{\dot{\omega}}(\dot{u} - \omega q) + Z_{\dot{\omega}}\dot{\omega} + Z_{\dot{q}}\dot{q} - X_{\dot{u}}uq \\
&\quad - X_{\dot{q}}q^2 + Y_{\dot{\omega}}\dot{v} + Z_{\dot{p}}\dot{p} + Z_{\dot{r}}\dot{r} + Y_{\dot{v}}vp + Y_{\dot{r}}rp + Y_{\dot{p}}p^2 + X_{\dot{v}}up + Y_{\dot{\omega}}\omega p \\
&\quad - X_{\dot{v}}vq - (X_{\dot{p}} - Y_{\dot{q}})pq - X_{\dot{r}}qr \\
&\quad - Z_w w - Z_{|\omega|\omega}|\omega|\omega + Z_u u + Z_v v + Z_p p + Z_q q + Z_r r - (W - B)c\theta c\phi \\
\tau_K &= I_x\dot{p} + (I_z - I_y)qr - (\dot{r} + pq)I_{xz} + (r^2 - q^2)I_{yz} + (pr - \dot{q})I_{xy} - mz_G(\dot{v} - \omega p + ur) \\
&\quad + X_{\dot{p}}\dot{u} + Z_{\dot{p}}\dot{\omega} + K_{\dot{q}}\dot{q} - X_{\dot{v}}\omega u + X_{\dot{r}}uq - Y_{\dot{\omega}}\omega^2 - (Y_{\dot{q}} - Z_{\dot{r}})\omega q + M_{\dot{r}}q^2 \qquad \text{(A.1)} \\
&\quad + Y_{\dot{p}}\dot{v} + K_{\dot{p}}\dot{p} + K_{\dot{r}}\dot{r} + Y_{\dot{\omega}}v^2 - (Y_{\dot{q}} - Z_{\dot{r}})vr + Z_{\dot{p}}vp - M_{\dot{r}}r^2 - K_{\dot{q}}rp + X_{\dot{\omega}}uv \\
&\quad - (Y_{\dot{v}} - Z_{\dot{\omega}})v\omega - (Y_{\dot{r}} + Z_{\dot{q}})\omega r - Y_{\dot{p}}\omega p - X_{\dot{q}}ur + (Y_{\dot{r}} + Z_{\dot{q}})vq + K_{\dot{r}}pq - (M_{\dot{q}} - N_{\dot{r}})qr \\
&\quad - K_p p - K_{|p|p}|p|p + K_u u + K_v v + K_\omega \omega + K_q q + K_r r + (z_G W)c\theta s\phi \qquad \text{(A.2)}
\end{aligned}
$$

$$
\begin{aligned}
\tau_M &= I_y\dot{q} + (I_x - I_z)\,rp - (\dot{p} + qr)\,I_{xy} + \left(p^2 - r^2\right)I_{zx} + (qp - \dot{r})\,I_{yz} - mz_G\,(\dot{u} - vr + \omega q) \\
&\quad + X_{\dot{q}}\,(\dot{u} + \omega q) + Z_{\dot{q}}\,(\dot{\omega} - uq) + M_{\dot{q}}\dot{q} - X_{\dot{\omega}}\left(u^2 - \omega^2\right) - (Z_{\dot{\omega}} - X_{\dot{u}})\,\omega u + Y_{\dot{q}}\dot{u} \\
&\quad + K_{\dot{q}}\dot{p} + M_{\dot{r}}\dot{r} + Y_{\dot{p}}ur - Y_{\dot{r}}vp - K_{\dot{r}}\left(p^2 - r^2\right) + (K_{\dot{p}} - N_{\dot{r}})\,rp - Y_{\dot{\omega}}uv + X_{\dot{\omega}}v\omega \\
&\quad - (X_{\dot{r}} + Z_{\dot{p}})\,(up - \omega r) + (X_{\dot{p}} - Z_{\dot{r}})\,(\omega p + ur) - M_{\dot{r}}pq + K_{\dot{q}}qr \\
&\quad + M_q q + M_{|q|q}|q|q + M_u u + M_v v + M_\omega \omega + M_p p + M_r r\,(z_G W)\,s\theta \\
\tau_N &= I_z\dot{r} + (I_y - I_z)\,pq - (\dot{q} + rp)\,I_{yz} + \left(q^2 - p^2\right)I_{xy} + (rp - \dot{p})\,I_{zx} + X_{\dot{r}}\dot{u} + Z_{\dot{r}}\dot{\omega} \\
&\quad + M_{\dot{r}}\dot{q} + X_{\dot{v}}u^2 + Y_{\dot{\omega}}\omega u - (X_{\dot{p}} - Y_{\dot{q}})\,uq - Z_{\dot{q}}\omega q - K_{\dot{q}}q^2 + Y_{\dot{r}}\dot{v} + K_{\dot{r}}\dot{p} \\
&\quad + N_{\dot{r}}\dot{r} - X_{\dot{v}}v^2 - X_{\dot{r}}vr - (X_{\dot{p}} - Y_{\dot{q}})\,vp + M_{\dot{r}}rp + K_{\dot{q}}p^2 - (X_{\dot{u}} - Y_{\dot{v}})\,uv - X_{\dot{\omega}}v\omega \\
&\quad + (X_{\dot{q}} + Y_{\dot{p}})\,up + Y_{\dot{r}}ur + Z_{\dot{q}}\omega p - (X_{\dot{q}} + Y_{\dot{p}})\,vq - (K_{\dot{p}} - M_{\dot{q}})\,pq - K_{\dot{r}}qr \\
&\quad + N_r r + N_{|r|r}|r|r + N_u u + N_v v + N_\omega \omega + N_p p + N_q q
\end{aligned}
$$

APPENDIX B

REYNOLDS NUMBER

In the middle of the head of our vehicle stands a point called stagnation point where the pressure is in its highest value compared to the other points of vehicle and the velocity of water is zero at this point during navigation. Here we see a laminar flow till to the occurence of seperation where the turbulent flow starts. From the figure it is seen that the Reynolds number between 10^3 and 3×10^5 denotes the laminar flow, after 3×10^5 flow becomes turbulent and increase of momentum avoids seperation therefore the drag decreases immediately [22].

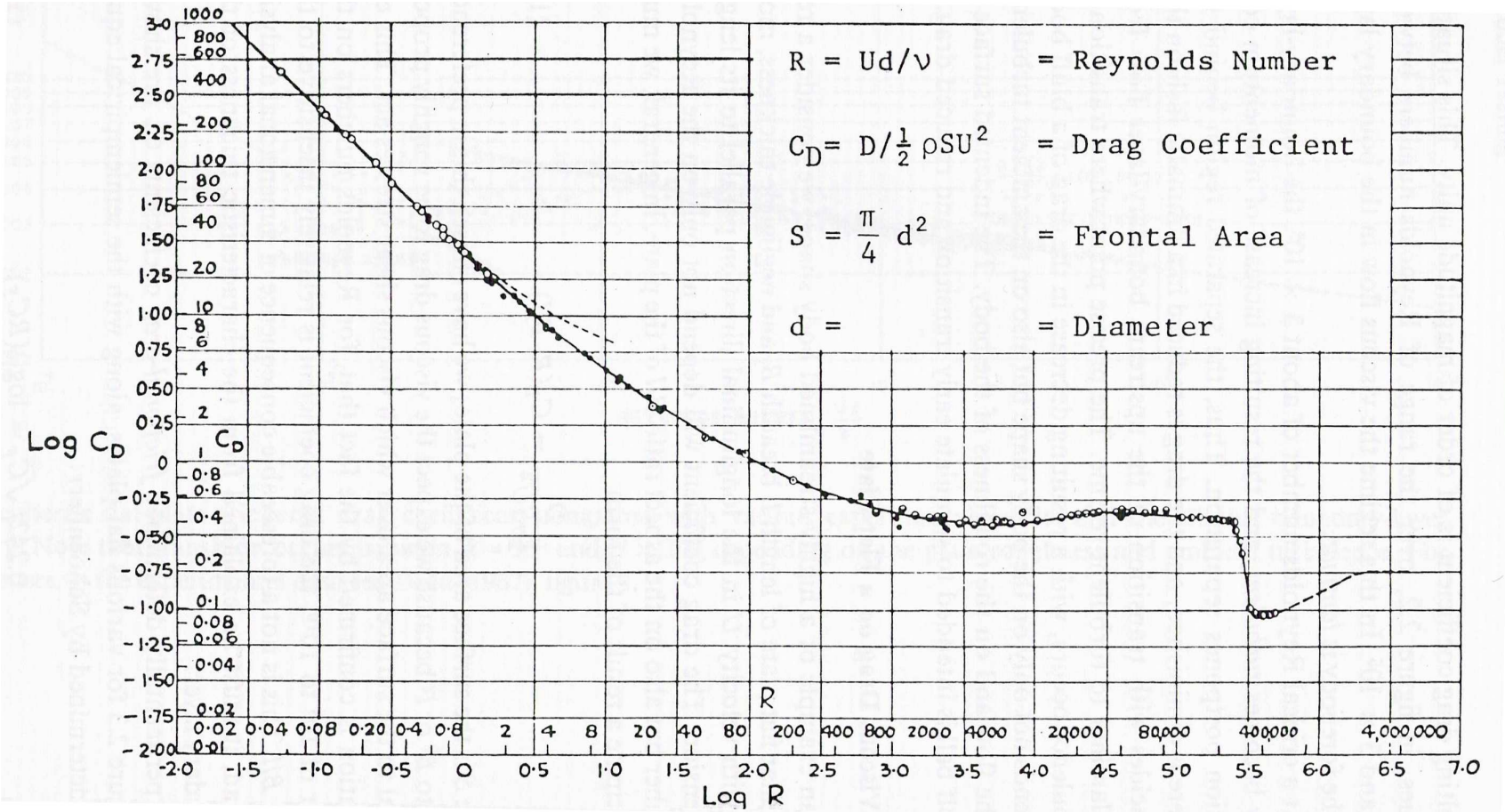

Figure B.1: The drag coefficient for a sphere [22].

APPENDIX C

SIMULINK PID MODELS

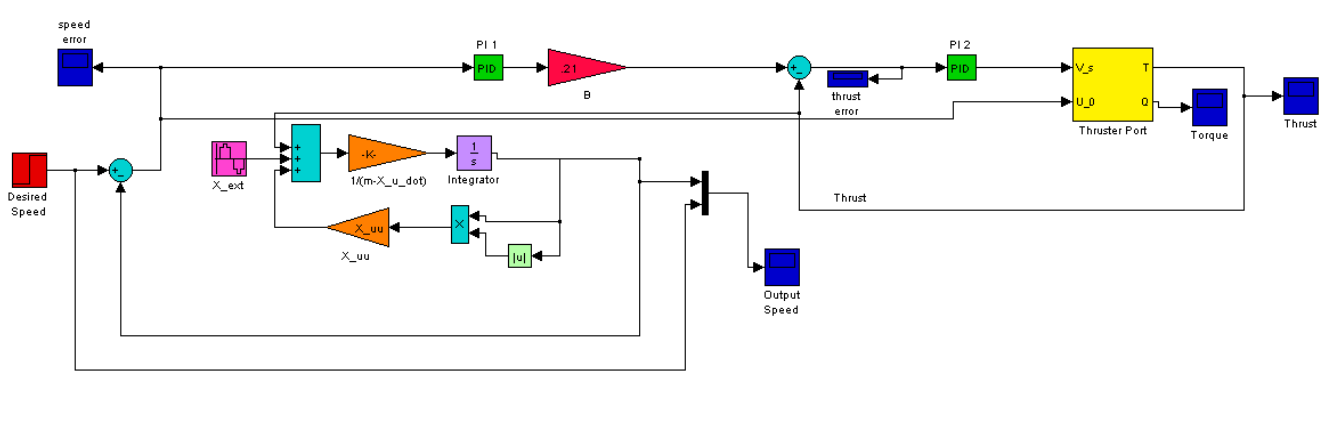

Figure C.1: Simulink PID Speed Model

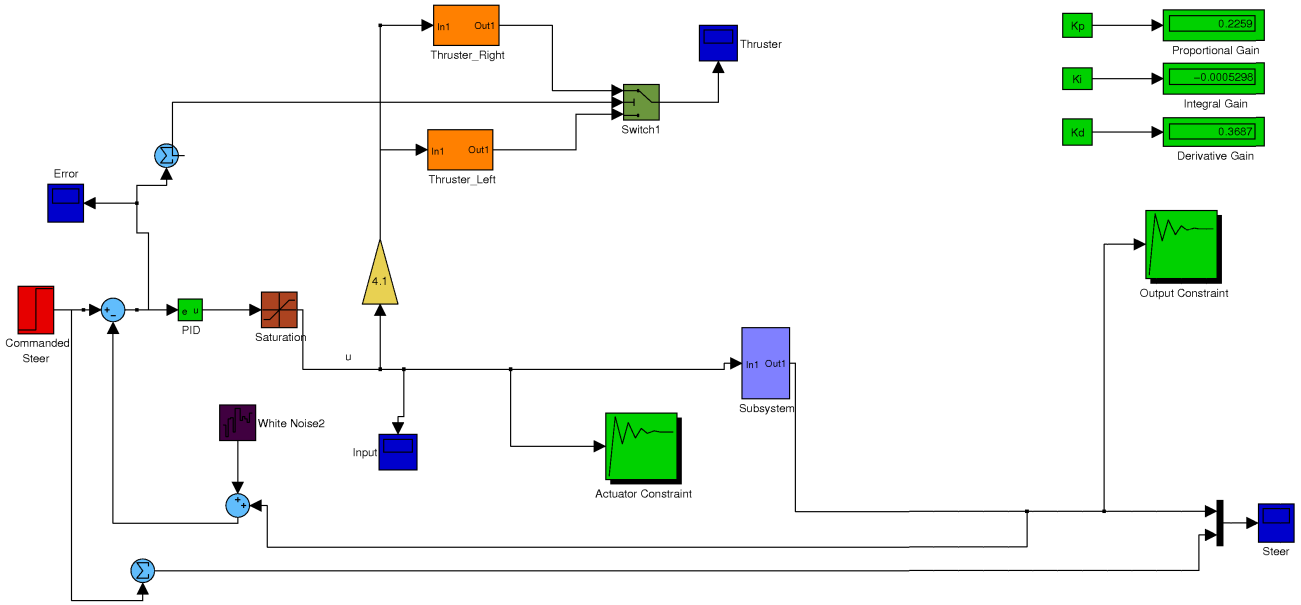

Figure C.2: Simulink PID Steering Model

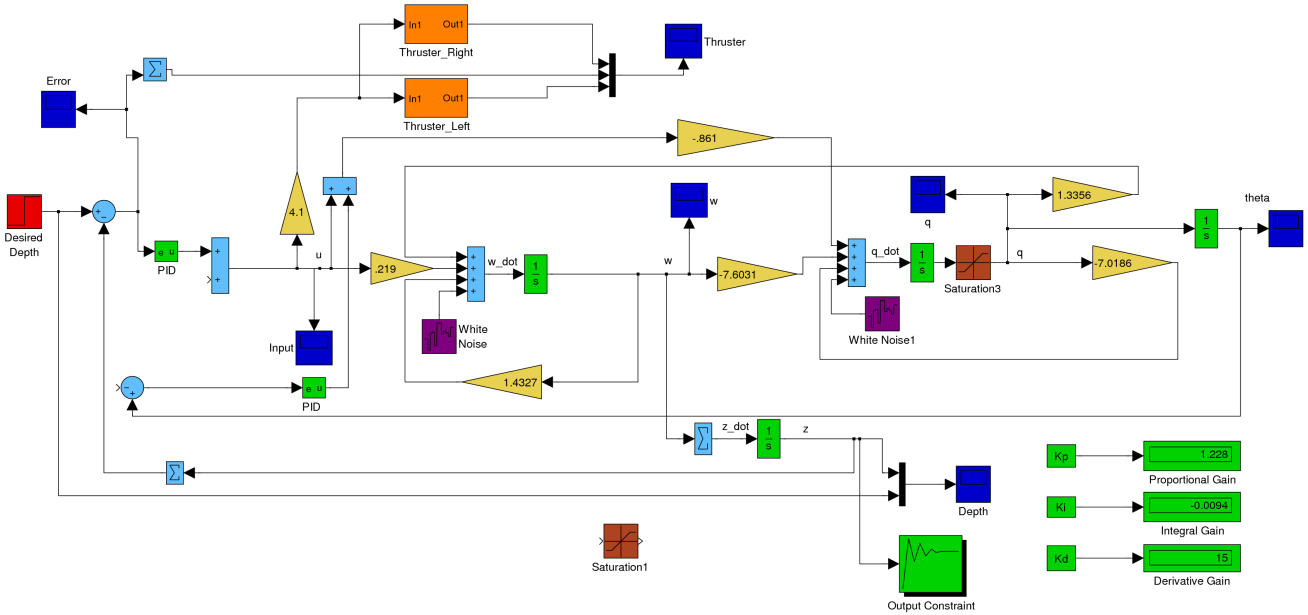

Figure C.3: Simulink PID Depth Model

APPENDIX D

GUIDANCE MODEL

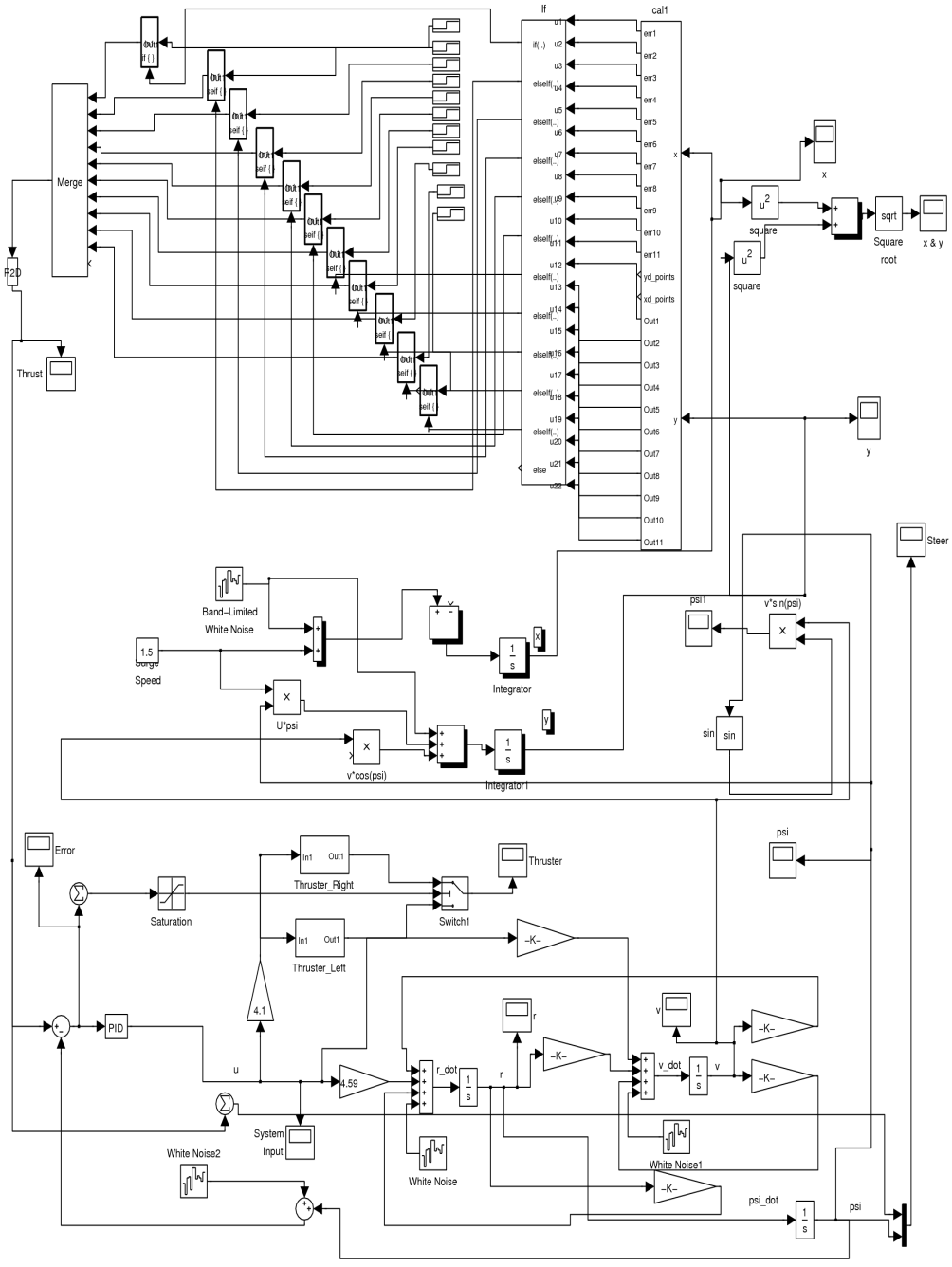

Figure D.1: Simulink Guidance Model