

NAVIGATION AND PATH PLANNING OF AN  
UNMANNED UNDERWATER VEHICLE

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

BY

UĞUR DOĞAN GÜL

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

SEPTEMBER 2012

Approval of the thesis:

**NAVIGATION AND PATH PLANNING OF AN  
UNMANNED UNDERWATER VEHICLE**

submitted by **UĞUR DOĞAN GÜL** 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 LEBLEBİCİOĞLU  
Supervisor, **Electrical and Electronics Engineering Dept., METU** \_\_\_\_\_

**Examining Committee Members:**

Prof. Dr. M. Kemal ÖZGÖREN  
Mechanical Engineering Dept., METU \_\_\_\_\_

Prof. Dr. M. Kemal LEBLEBİCİOĞLU  
Electrical and Electronics Engineering Dept., METU \_\_\_\_\_

Prof. Dr. Reşit SOYLU  
Mechanical Engineering Dept., METU \_\_\_\_\_

Assist. Prof. Dr. Umut ORGUNER  
Electrical and Electronics Engineering Dept., METU \_\_\_\_\_

Bülent BİLGİN, M.Sc.  
ASELSAN Inc. \_\_\_\_\_

**Date:** 11.09.2012

**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 : Uğur Doğan GÜL

Signature :

## **ABSTRACT**

# **NAVIGATION AND PATH PLANNING OF AN UNMANNED UNDERWATER VEHICLE**

GÜL, Uğur Doğan

M. Sc., Department of Electrical and Electronics Engineering  
Supervisor: Prof. Dr. M. Kemal LEBLEBİCİOĞLU

September 2012, 114 pages

Due to the conditions peculiar to underwater, distinctive approaches are required to solve the navigation and path planning problem of an unmanned underwater vehicle (UUV). In this study, first of all, a detailed 6 degrees-of-freedom (DOF) mathematical model is formed, including the coupled non-linear forces and moments acting on an underwater vehicle. The hydrodynamic coefficients which correspond to the geometry of the vehicle which the model is based on are calculated using the strip theory. After the mathematical model is obtained, by applying appropriate linearization on the model, “Linear Quadratic Regulator (LQR)” control method is implemented to govern the surge, heave, pitch and yaw motions of the underwater vehicle. Path planning algorithm of the vehicle is based on tracking the waypoints. Permutation of the waypoints is obtained by solving the “Travelling Salesman Problem (TSP)” via genetic algorithm. Linked with that, “Rapidly-Exploring Random Trees (RRT)” algorithm is introduced into the path planning algorithm to avoid collisions in environments with obstacles. Underwater navigation solution is based on the “Inertial Navigation System (INS)” outputs, located on the vehicle. To

correct the long-term drift of the inertial solution, “Kalman Filter” based integration algorithm is used and external aids such as “Global Navigation Satellite System (GNSS)”, “Ultra-Short Baseline (USBL)” acoustic navigation system and attitude sensors have been utilized. The control method, path planning and navigation algorithms used in this study are verified by simulation results.

**Keywords:** Underwater vehicle modelling, LQR, RRT, underwater navigation.

## ÖZ

# BİR İNSANSIZ SUALTI ARACININ NAVİGASYONU VE YOL PLANLAMASI

GÜL, Uğur Doğan

Yüksek Lisans, Elektrik Elektronik Mühendisliği Bölümü  
Tez Yöneticisi: Prof. Dr. M. Kemal LEBLEBİCİOĞLU

Eylül 2012, 114 sayfa

Sualtına özgü koşullar nedeniyle, insansız sualtı aracı navigasyon ve yol planlama problemi çözümünde farklı yaklaşımlara ihtiyaç duyulmaktadır. Bu çalışmada, ilk olarak, sualtı aracına etkiyen bağlaşımlı ve doğrusal olmayan kuvvet ve momentleri de içeren 6 serbestlik derecesine sahip detaylı bir matematiksel model oluşturulmuştur. Modellenen aracın geometrisine uygun hidrodinamik katsayılar dilim teorisi kullanılarak hesaplanmıştır. Matematiksel model elde edildikten sonra, sualtı aracının ileri-geri, yukarı-aşağı, yunuslama ve sapma hareketlerini denetlemek için, uygun doğrusallaştırma yapılarak, “Doğrusal Karesel Regülatör” kontrol metodu uygulanmıştır. Aracın yol planlama algoritması temel olarak yol noktalarını takip etme üzerinedir. Bu yol noktalarının sıralaması, genetik algoritma ile “Seyyar Satıcı Problemi” çözümlenerek elde edilmiştir. Bununla bağlantılı olarak, ortamda engeller olduğunda çarpışmalardan sakınmak için “Hızlı-Keşfeden Rastlantısal Ağaçlar” algoritması da yol planlama algoritmasına dahil edilmiştir. Sualtı navigasyon çözümünde araç üzerinde bulunan “Ataletsel Navigasyon Sistemi (ANS)” çıktıları esas alınmıştır. Ataletsel çözümün uzun süreçteki sapmalarını düzeltmek için ise “Kalman Süzgeç” tabanlı entegrasyon algoritması kullanılmış ve

“Küresel Uydu Navigasyon Sistemi”, “Çok Kısa Bazlı” akustik navigasyon sistemi ve yönelim sensörleri gibi dış yardımlardan yararlanılmıştır. Bu çalışmada kullanılan kontrol metodu, yol planlama ve navigasyon algoritmaları benzetim sonuçları ile doğrulanmıştır.

Anahtar Kelimeler: Sualtı aracı modelleme, LQR, RRT, sualtı navigasyonu.

**to my dear family,**

## **ACKNOWLEDGEMENTS**

I would like to express my sincere gratitude to my supervisor Prof. Dr. M. Kemal Leblebiciođlu for his guidance, encouragement, advice and support throughout the thesis study.

I would like to thank Kenan Ahıska and Ayhan Özgür for their valuable ideas and suggestions.

I am grateful to ASELSAN Inc. for their support during the course of my M.Sc. education.

Finally, I would like to express special thanks to my family and my friends for their love and support.

# TABLE OF CONTENTS

<b>ABSTRACT</b> .....	<b>IV</b>
<b>ÖZ</b> .....	<b>VI</b>
<b>ACKNOWLEDGEMENTS</b> .....	<b>IX</b>
<b>TABLE OF CONTENTS</b> .....	<b>X</b>
<b>LIST OF TABLES</b> .....	<b>XIII</b>
<b>LIST OF FIGURES</b> .....	<b>XIV</b>
<b>LIST OF ABBREVIATIONS</b> .....	<b>XVII</b>
<b>CHAPTERS</b>	
<b>1 INTRODUCTION</b> .....	<b>1</b>
1.1 CLASSIFICATION.....	1
1.2 HISTORY .....	1
1.3 UUV TECHNOLOGY .....	2
1.4 APPLICATION AREAS .....	2
1.5 ULISAR.....	2
1.6 ORGANIZATION .....	5
<b>2 MATHEMATICAL MODELLING OF UUV'S</b> .....	<b>6</b>
2.1 INTRODUCTION.....	6
2.2 KINEMATICS .....	6
2.2.1 Notation of Kinematic Quantities .....	7
2.2.2 Coordinate Frames.....	8
2.2.2.1 Earth-Centered Inertial Frame.....	9
2.2.2.2 Earth-Centered Earth-Fixed Frame .....	9
2.2.2.3 Local Navigation Frame.....	10
2.2.2.4 Body-Fixed Frame .....	10
2.2.3 Euler Angles .....	11
2.3 RIGID BODY DYNAMICS .....	11
2.4 HYDRODYNAMIC EFFECTS .....	14

2.4.1 Added Mass .....	14
2.4.2 Hydrodynamic Damping.....	19
2.5 HYDROSTATIC EFFECTS.....	22
2.6 THRUSTERS' DYNAMICS.....	23
2.7 EQUATIONS OF MOTION .....	24
<b>3 CONTROLLER DESIGN .....</b>	<b>27</b>
3.1 STATE-SPACE REPRESENTATION .....	27
3.2 LINEARIZATION .....	27
3.3 LINEAR QUADRATIC REGULATOR (LQR) .....	28
<b>4 GUIDANCE.....</b>	<b>36</b>
4.1 INTRODUCTION.....	36
4.2 WAY POINT GUIDANCE BY LINE OF SIGHT (LOS) .....	37
<b>5 PATH PLANNING .....</b>	<b>43</b>
5.1 INTRODUCTION.....	43
5.2 PATH PLANNING PROBLEM.....	44
5.3 GENETIC ALGORITHM .....	45
5.4 MOTION PLANNING PROBLEM.....	53
5.5 RAPIDLY-EXPLORING RANDOM TREES (RRT) ALGORITHM .....	56
<b>6 NAVIGATION .....</b>	<b>66</b>
6.1 INTRODUCTION.....	66
6.2 INERTIAL NAVIGATION SYSTEMS .....	68
6.2.1 Inertial Measurement Unit .....	69
6.2.1.1 Accelerometers .....	69
6.2.1.2 Gyroscopes.....	70
6.2.1.3 Error Characteristics .....	70
6.2.2 Inertial Navigation Processor.....	71
6.2.2.1 Kinematics .....	72
6.2.2.2 Earth Model .....	75
6.2.2.3 Inertial Navigation Equations .....	79
6.2.2.4 INS Error Propagation .....	82
6.3 EXTERNAL AIDS .....	83
6.3.1 Global Navigation Satellite System (GNSS) .....	83
6.3.2 Acoustic Navigation .....	85
6.3.3 Attitude Measurement .....	87
6.4 INTEGRATED NAVIGATION .....	87

6.4.1 Integration Architecture .....	89
6.4.2 State Selection and System Model.....	89
6.4.3 Measurement Model .....	93
6.4.4 Estimation Algorithm .....	94
6.5 SIMULATION RESULTS.....	96
<b>7 CONCLUSION AND FUTURE WORKS.....</b>	<b>107</b>
7.1 CONCLUSION .....	107
7.2 FUTURE WORKS .....	108
<b>REFERENCES .....</b>	<b>109</b>
<b>APPENDICES</b>	
<b>A DRAG COEFFICIENT .....</b>	<b>113</b>
<b>B GNSS POSITION ERROR.....</b>	<b>114</b>
<b>CURRICULUM VITAE</b>	

## LIST OF TABLES

### TABLES

Table 1.1 ULISAR UUV Specifications .....	4
Table 2.1 Notation used for underwater vehicles [4] .....	7
Table 2.2 Axial added mass empirical parameter $\alpha$ [8] .....	15
Table 5.1 Positions of the locations used in GA simulation .....	49
Table B.1 GNSS position error standard deviation for different settings [5] .....	114

# LIST OF FIGURES

## FIGURES

Figure 1.1 Hull of ULISAR UUV .....	3
Figure 1.2 Main components of ULISAR UUV [2] .....	4
Figure 2.1 ECI and ECEF coordinate frames.....	8
Figure 2.2 Local navigation frame .....	9
Figure 2.3 Body-fixed frame.....	10
Figure 2.4 Bare hull of the vehicle [10] .....	16
Figure 2.5 Dimensions of the bare hull .....	16
Figure 2.6 Hull with wings [10].....	18
Figure 2.7 Thruster configuration – Bottom view [10].....	23
Figure 3.1 Plant and optimal feedback controller for linear tracking problems [13].	29
Figure 3.2 Forward speed step response .....	32
Figure 3.3 Forward thrusts for forward speed step response .....	32
Figure 3.4 Heave speed step response.....	33
Figure 3.5 Vertical thrusts heave speed step response .....	33
Figure 3.6 Pitch rotation step response .....	34
Figure 3.7 Vertical thrusts for step pitch rotation .....	34
Figure 3.8 Yaw rotation step response .....	35
Figure 3.9 Forward thrusts for step yaw rotation.....	35
Figure 4.1 Way points and circle of acceptance [17].....	37
Figure 4.2 Waypoint guidance by LOS simulation in 3D.....	39
Figure 4.3 Waypoint guidance with $\rho_0 = 1m$ .....	40
Figure 4.4 Waypoint guidance with $\rho_0 = 3.2m$ .....	41
Figure 4.5 Waypoint guidance by LOS simulation for a reconnaissance mission....	42
Figure 5.1 GA based TSP simulation for the population size of 50 .....	50
Figure 5.2 GA based TSP simulation for the population size of 100 .....	51

Figure 5.3 GA based TSP simulation for the population size of 200 .....	52
Figure 5.4 3D path generated by the GA based TSP algorithm.....	53
Figure 5.5 Obstacles and imminent collision regions in state space [32] .....	56
Figure 5.6 Extension of RRT .....	58
Figure 5.7 Standard RRT algorithm simulation result .....	59
Figure 5.8 Goal-biased RRT algorithm simulation result.....	60
Figure 5.9 RRT with high goal-biasing in a sparse environment simulation result...	61
Figure 5.10 RRT with low goal-biasing in a dense environment simulation result...	62
Figure 5.11 RRT with varying goal-biasing simulation result.....	63
Figure 5.12 TSP/RRT integration simulation result 1 .....	64
Figure 5.13 TSP/RRT integration simulation result 2 .....	65
Figure 5.14 TSP/RRT integration simulation result 3 .....	65
Figure 6.1 Ranging and bearing [5] .....	67
Figure 6.2 Dead-reckoning [5].....	68
Figure 6.3 Inertial navigation system.....	69
Figure 6.4 The ellipsoid model of the Earth’s surface .....	76
Figure 6.5 Curvilinear position components.....	77
Figure 6.6 Inertial navigation equations.....	80
Figure 6.7 GNSS user equipment .....	84
Figure 6.8 GNSS ranging [5] .....	84
Figure 6.9 USBL ranging and bearing [36] .....	86
Figure 6.10 Overall integrated navigation system .....	88
Figure 6.11 Integrated navigation system .....	89
Figure 6.12 True position of underwater vehicle in local navigation coordinate frame .....	98
Figure 6.13 True velocity of underwater vehicle in ECEF coordinate frame.....	98
Figure 6.14 True attitude of underwater vehicle.....	99
Figure 6.15 INS-indicated position of underwater vehicle in local navigation coordinate frame.....	99
Figure 6.16 INS-indicated velocity of underwater vehicle in ECEF coordinate frame .....	100

Figure 6.17 INS-indicated attitude of underwater vehicle .....	100
Figure 6.18 GNSS-indicated position of surface vessel in local navigation coordinate frame.....	101
Figure 6.19 USBL-indicated range and bearing from the surface vessel to the underwater vehicle in ECEF coordinate frame .....	101
Figure 6.20 INS position error of underwater vehicle in ECEF coordinate frame ..	102
Figure 6.21 Estimated position error of underwater vehicle in ECEF coordinate frame.....	102
Figure 6.22 INS velocity error of underwater vehicle in ECEF coordinate frame ..	103
Figure 6.23 Estimated velocity error of underwater vehicle in ECEF coordinate frame.....	103
Figure 6.24 INS attitude error of underwater vehicle .....	104
Figure 6.25 Estimated attitude error of underwater vehicle.....	104
Figure 6.26 Position estimation error of integrated navigation in ECEF coordinate frame.....	105
Figure 6.27 Velocity estimation error of integrated navigation in ECEF coordinate frame.....	105
Figure 6.28 Attitude estimation error of integrated navigation.....	106
Figure 6.29 Trajectories of the surface vessel and the underwater vehicle in the local navigation frame.....	106
Figure A.1 The drag coefficient as a function of Reynolds number for different geometrical objects.....	113

## LIST OF ABBREVIATIONS

AUV	Autonomous underwater vehicle
DOF	Degrees of freedom
ECEF	Earth-centered Earth-fixed
ECI	Earth-centered inertial
FOG	Fiber-optic gyroscope
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IMU	Inertial measurement unit
INS	Inertial navigation system
LBL	Long baseline
LOS	Line of sight
LQR	Linear quadratic regulator
PMX	Partially-mapped crossover
PRM	Probabilistic roadmap method
RLG	Ring laser gyroscope
ROV	Remotely operated vehicle
RRT	Rapidly-exploring random trees
SBL	Short baseline
SNAME	Society of Naval Architects and Marine Engineers
TCM	Thruster control matrix
TSP	Travelling salesman problem
UAV	Unmanned aerial vehicle
USBL	Ultra-short baseline
UUV	Unmanned underwater vehicle

# CHAPTER 1

## INTRODUCTION

### 1.1 Classification

UUV's constitute one of the two main branches of submersible vehicles, which are called as manned and unmanned systems. Manned submersible vehicles are well-known systems used for both military and non-military missions. The UUV's also fall into a number of different sub-classes. The simplest UUV's are the ones which are towed behind a ship, acting as platforms for various sensor systems. Another type of UUV is called Remotely Operated Vehicle (ROV), which is a tethered vehicle and controlled by a remote operator. A third type of UUV's is called Autonomous Underwater Vehicle (AUV), which is untethered and can control itself to carry out a given task.

### 1.2 History

Even though manned submersible vehicles exist for more than 200 years, the initial development of AUV's began in 1960's. During 1970's and 1980's the potential of AUV's systems were realized, new laboratories were established and testbeds were built in various research centers. In 1980's the advances in technology, i.e., computers and software system, accelerated the development of AUV's and Proof of Concept prototypes were developed, tested and used. In 1990's funding in AUV development was broadened due to international collaborations. Also in that decade, researchers were focused on mission based AUV development. In 2000's and also today the AUV technology is commercialized and AUV market have been defined. While the economic viability of the technology is still needs to be proven, there are many technological problems waiting to be solved [1].

### **1.3 UUV Technology**

The current research for UUV's can be listed as; autonomy, energy, navigation, sensors and communication. As more complex objectives were assigned to UUV's, more intelligent systems are need to be designed, so autonomy is an important focus on the UUV research. Especially multiple cooperating UUV systems have not been investigated significantly. Energy is a limiting factor for the development of enduring UUV's, new battery technologies and solar energy solutions are being researched. Since the inertial sensors suffers from drift of the navigation solution with time, and accurate sensors are too expensive for non-military users, new underwater technologies were being researched, including acoustic navigation systems, feature matching techniques and so on. AUV development also encourages the development of smarter, lower power, highly reliable and smaller sensors as well as optical and acoustic imaging systems. Lastly, the acoustic communication research is focused on lowering the error rate and expanding the range. Another aspect of underwater communication is the requirement of network-based communication between multiple collaborating UUV's [1].

### **1.4 Application Areas**

For the last decades, the application areas of UUV's have been fast growing, and more tasks and missions have been defined. Commercial UUV's are used by oil and gas industry for seafloor mapping and bathymetric survey. Military UUV's are used for mine detection, monitoring of protected areas, and anti-submarine warfare. UUV's are also used by scientists to study lakes, ocean and ocean floor.

### **1.5 ULISAR**

The mathematical model in this thesis is based on ULISAR UUV, which is developed by the support of TUBITAK. It is a small-size ROV whose main objective is defined as underwater reconnaissance. ULISAR UUV exhibits a novel mechanical

design, optimized for stability, durability, energy-minimization and modularity [2]. The hull of ULISAR UUV is illustrated in Figure 1.1 Hull of ULISAR UUV.

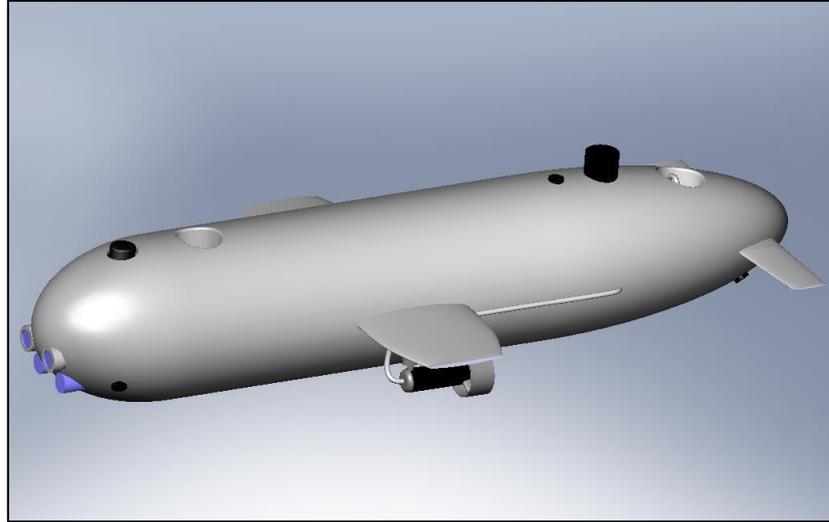


Figure 1.1 Hull of ULISAR UUV

ULISAR UUV is equipped with an inertial measurement unit, an imaging sonar, two B/W cameras and two lights, an electronic box with PC-104 stack and video grabber inside, Li-Po batteries, an acoustic modem and a pressure sensor. ULISAR contains 4 thrusters to perform independent surge, yaw, heave and pitch motions, and contains stabilizers and fins to be stabilized at roll passively [2]. The main components of the vehicle are shown in Figure 1.2 and the specifications are given in Table 1.1. More information about the ULISAR project and the development of the UUV can be found in [3].

Table 1.1 ULISAR UUV Specifications

Length	1.60 m
Mass	64 kg
Diameter	35 cm
Weight	627 kgm/s
Buoyancy	631 kgm/s
Endurance	1 hr with 0.75 m/s
Surge Speed	1.02 m/s
Heave Speed	0.20 m/s
Diving Capacity	80 m
Sonar Range	100 m

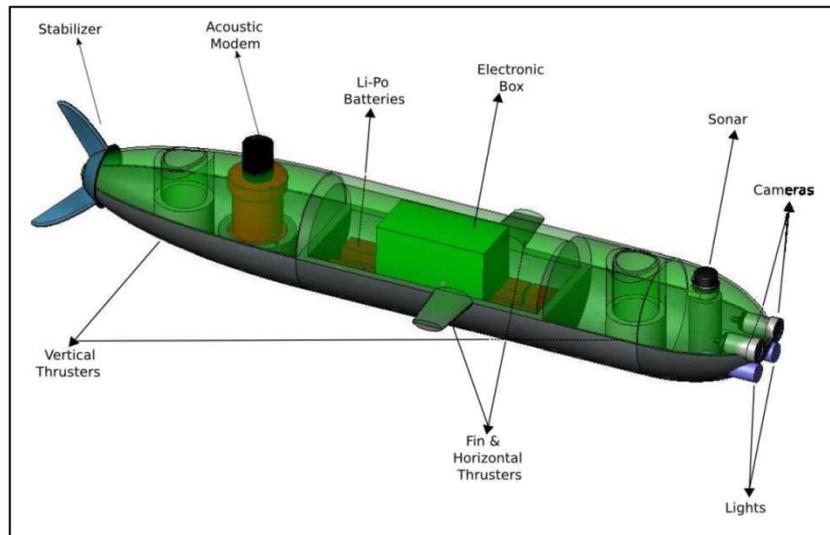


Figure 1.2 Main components of ULISAR UUV [2]

## 1.6 Organization

The organization of the thesis is as follows:

- Chapter 1 gives brief information about history, technology and application areas of UUV's and mentions about the ULISAR UUV,
- Chapter 2 forms the 6 DOF coupled and non-linear mathematical model of the ULISAR UUV and defines the rigid body, hydrodynamic and hydrostatic equations of motion on this model,
- Chapter 3 linearizes the mathematical model and applies LQR control technique to control the surge, heave, pitch and yaw motions of the vehicle,
- Chapter 4 introduces waypoint guidance by LOS algorithm to drive the vehicle towards the given waypoints while achieving smooth turns during the update of the next waypoint,
- Chapter 5 describes the path planning problem expressed as a Traveling Salesman Problem which is solved by a genetic algorithm and the RRT based motion planner avoiding collisions in an environment with obstacles,
- Chapter 6 explains the proposed navigation solution for UUV's, integrating the Inertial Navigation System with Global Navigation Satellite System, acoustic navigation system, and aiding sensor information and shows the simulation results,
- Chapter 7 gives a summary and discussion about the results of this study, and concludes by mentioning some future work to be considered.

## CHAPTER 2

### MATHEMATICAL MODELLING OF UUV'S

#### 2.1 Introduction

A complete and precise mathematical model of a physical system is always advantageous to form a high performance control system. Modelling of UUV's requires an understanding in the fields of statics and dynamics. Statics is the study of bodies in equilibrium, i.e., at rest or moving with constant velocity. Dynamics however is concerned with bodies having accelerated motion. In this chapter, first the geometrical aspects of the motion will be discussed including various coordinate frames and respective angles. After that the analysis of the forces causing the motion will be discussed. First the rigid-body dynamics of a body will be described, and then hydrodynamic and hydrostatic forces and moments will be examined. Later thrusters' dynamics will be briefly introduced. At the end the governing equations of motion will be formed and it will be shown how these equations can be simplified particularly for the ULISAR underwater vehicle which our model is based on.

#### 2.2 Kinematics

Since 6 independent coordinates are necessary to determine the position and orientation of a rigid body, the motion of the underwater vehicle will be discussed in 6 *degrees of freedom* (DOF). Among those, the first 3 coordinates and their time derivatives describe the position and translational motion along x, y, and z-axes, while the last 3 coordinates and their time derivatives describe attitude and rotational motion. For underwater vehicles, the 6 different motion components are named as:

surge, sway, heave, roll, pitch, and yaw. Table 2.1 gives a list of the kinematic quantities describing the motion in 6 DOF in SNAME notation.

Table 2.1 Notation used for underwater vehicles [3]

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	$\phi$
5	Rotation in the y-axis (pitch)	M	q	$\theta$
6	Rotation in the z-axis (yaw)	N	r	$\psi$

### 2.2.1 Notation of Kinematic Quantities

Different authors describe kinematic quantities in different notations and some of those have potential to cause confusion. Among the others, notation of Groves [5] is found to be more complete and clear, so it is used throughout this thesis. In this notation any kinematic quantity involves three coordinate frames:

- Object frame: The frame whose motion is described,  $\alpha$ ;
- Reference frame: The frame with which that motion is respect to,  $\beta$ ;
- Resolving frame: The set of axes in which that motion is represented,  $\gamma$ .

The notation used is:

$$x_{\beta\alpha}^{\gamma}$$

where the vector,  $x$ , refers to a kinematic property of frame  $\alpha$  with respect to frame  $\beta$ , expressed in the frame  $\gamma$  axes.

## 2.2.2 Coordinate Frames

A coordinate frame provides an origin and a set of axes in terms of which the motion of the vehicle is described. To describe the motion of an underwater vehicle, only two coordinate frames are enough to define, namely the body-fixed frame and the Earth-centered Earth-fixed frame. The position and attitude of the vehicle are defined with respect to the Earth-fixed frame whereas the linear and angular velocities of the vehicle are defined with respect to the body-fixed frame. Although the Earth-fixed frame rotates about Earth's z-axis, in mathematical modelling part, we can assume it as inertial, since this rotation has no significant impact on the vehicle motion. However, for navigation this assumption is inaccurate; the Earth's rotation has impact on navigation computation. Therefore, in navigation, an inertial frame is also defined and called the Earth-centered inertial frame. In addition, again for navigation, we define a fourth coordinate frame, called the local navigation frame. For the sake of completeness all of these coordinate frames are defined in this section.

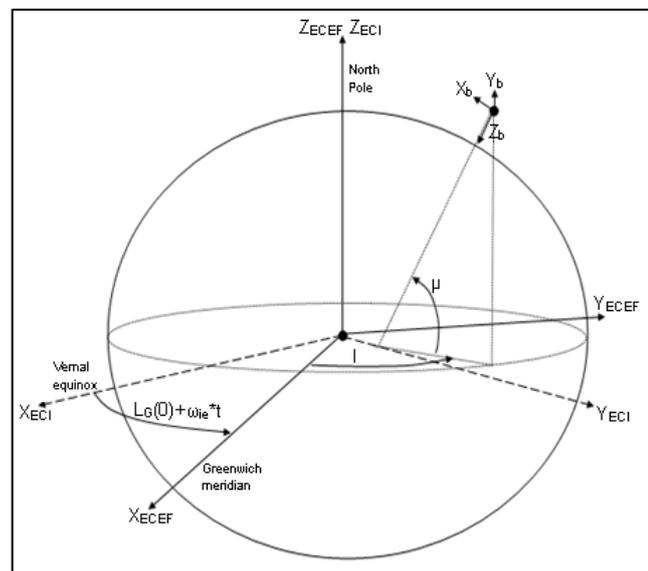


Figure 2.1 ECI and ECEF coordinate frames

### 2.2.2.1 Earth-Centered Inertial Frame

Earth-centered inertial frame (ECI) is denoted by the symbol  $i$ , centered at the Earth's center of mass and oriented with respect to the Earth's spin axis and the stars; therefore it is inertial. The z-axis points along the Earth's axis of rotation from the center to the true North Pole. The x- and y-axes lie within the equatorial plane. They do not rotate with the Earth, but the y-axis always lies 90 degrees ahead of the x-axis in the direction of rotation [5]. ECI coordinate frame is seen on Figure 2.1.

### 2.2.2.2 Earth-Centered Earth-Fixed Frame

Earth-centered Earth-fixed frame (ECEF) frame is similar to the ECI frame, except that all axes remain fixed with respect to the Earth. Therefore it rotates with respect to the ECI frame, about the z-axis which coincides with ECI z-axis. The x-axis points from the center to the intersection of the equator with the IERS reference meridian, which defines 0 degree longitude. The y-axis completes the right-handed orthogonal set; pointing from the center to the intersection of the equator with the 90° east meridian [5]. ECEF coordinate frame is also seen on Figure 2.1.

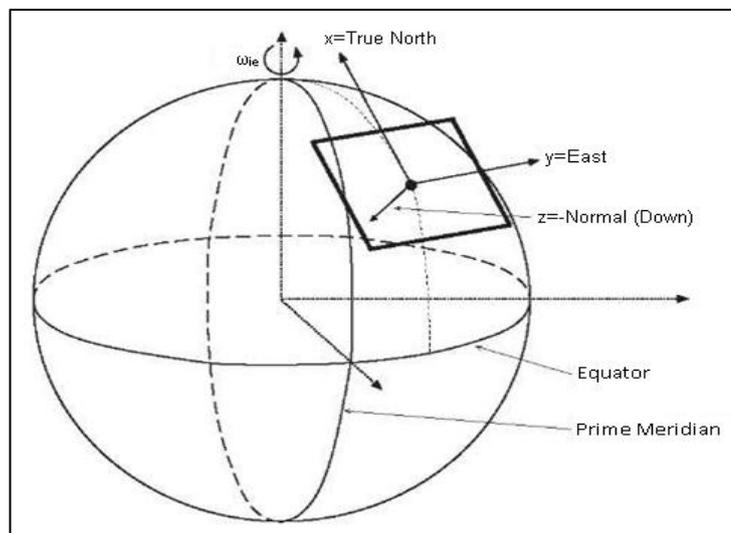


Figure 2.2 Local navigation frame

### 2.2.2.3 Local Navigation Frame

Local navigation frame (NED frame) is denoted by the symbol  $n$ , and its origin is the point the navigation solution is sought for. In this study, the center of the navigation solution is taken as the center of inertia of the underwater vehicle. The  $z$ -axis, also known as the down (D) axis, is defined as the normal to the surface of the reference ellipsoid, pointing toward the center of the Earth. The  $x$ -axis, or north (N) axis, is the projection in the plane orthogonal to the  $z$ -axis of the line from the vehicle to the North Pole. By completing the orthogonal set, the  $y$ -axis always points east and is hence known as the east (E) axis [5]. Local navigation frame is seen on Figure 2.2.

### 2.2.2.4 Body-Fixed Frame

Body-fixed frame is denoted by the symbol  $b$ , and its origin is the center of inertia of the underwater vehicle. The origin is also coincident with that of the local navigation frame, but the axes remain fixed with respect to the body and are generally defined as  $x$  = forward,  $z$  = down,  $y$  = right, completing the orthogonal set [5]. Body-fixed frame is seen on Figure 2.3.

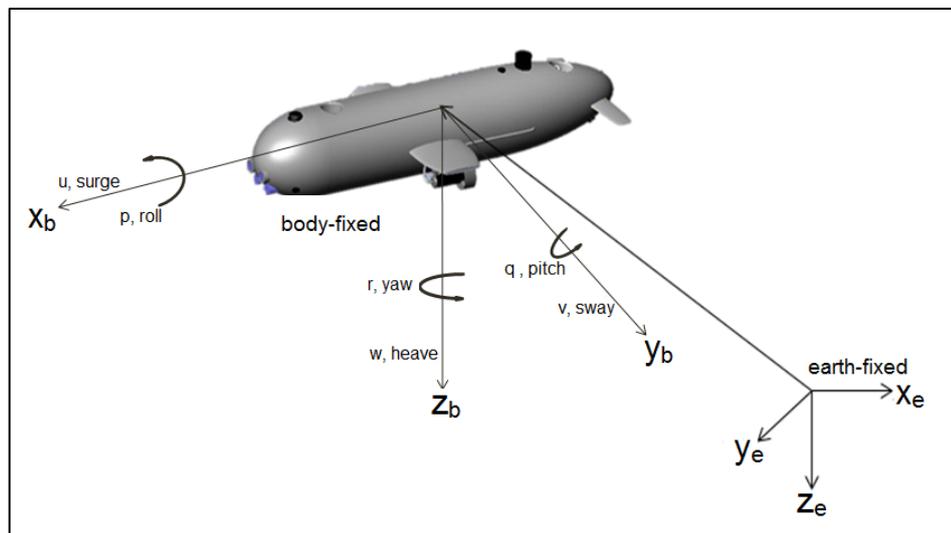


Figure 2.3 Body-fixed frame

### 2.2.3 Euler Angles

Euler angles are used to describe the attitude of the vehicle with respect to the Earth-fixed coordinates or local navigation frame coordinates. Attitude is represented by 3 successive rotations about roll, pitch, and yaw axes which transforms the reference frame into the vehicle's body frame. Euler angles are also used to convert linear and angular velocity components resolved in body frame coordinates into rate of change of position and attitude quantities resolved in ECEF coordinates. If we define position, attitude, linear velocity and angular velocity vectors respectively as following:

$$r_{eb}^e = \begin{bmatrix} x_{eb}^e \\ y_{eb}^e \\ z_{eb}^e \end{bmatrix}, \Psi_{eb} = \begin{bmatrix} \phi_{eb} \\ \theta_{eb} \\ \psi_{eb} \end{bmatrix}, v_{eb}^b = \begin{bmatrix} u_{eb}^b \\ v_{eb}^b \\ w_{eb}^b \end{bmatrix}, \omega_{eb}^b = \begin{bmatrix} p_{eb}^b \\ q_{eb}^b \\ r_{eb}^b \end{bmatrix}$$

Then the following coordinate transformations can be applied [3]:

$$\dot{r}_{eb}^e = J_1(\Psi_{eb})v_{eb}^b \quad (2.1)$$

$$\dot{\Psi}_{eb} = J_2(\Psi_{eb})\omega_{eb}^b \quad (2.2)$$

where

$$J_1(\Psi_{eb}) = \begin{bmatrix} \cos\theta_{eb}\cos\psi_{eb} & \begin{pmatrix} -\cos\phi_{eb}\sin\psi_{eb} \\ +\sin\phi_{eb}\sin\theta_{eb}\cos\psi_{eb} \end{pmatrix} & \begin{pmatrix} \sin\phi_{eb}\sin\psi_{eb} \\ +\cos\phi_{eb}\sin\theta_{eb}\cos\psi_{eb} \end{pmatrix} \\ \cos\theta_{eb}\sin\psi_{eb} & \begin{pmatrix} \cos\phi_{eb}\cos\psi_{eb} \\ +\sin\phi_{eb}\sin\theta_{eb}\sin\psi_{eb} \end{pmatrix} & \begin{pmatrix} -\sin\phi_{eb}\cos\psi_{eb} \\ +\cos\phi_{eb}\sin\theta_{eb}\sin\psi_{eb} \end{pmatrix} \\ -\sin\theta_{eb} & \begin{pmatrix} \sin\phi_{eb}\cos\theta_{eb} \\ \cos\phi_{eb}\cos\theta_{eb} \end{pmatrix} & \begin{pmatrix} \cos\phi_{eb}\cos\theta_{eb} \\ \sin\phi_{eb}\cos\theta_{eb} \end{pmatrix} \end{bmatrix} \quad (2.3)$$

$$J_2(\Psi_{eb}) = \begin{bmatrix} 1 & \sin\phi_{eb}\tan\theta_{eb} & \cos\phi_{eb}/\tan\theta_{eb} \\ 0 & \cos\phi_{eb} & -\sin\phi_{eb} \\ 0 & \sin\phi_{eb}/\cos\theta_{eb} & \cos\phi_{eb}/\cos\theta_{eb} \end{bmatrix} \quad (2.4)$$

Here  $J_1(\Psi_{eb})$  is also known as the earth-to-body coordinate transformation matrix, which will be utilized further in navigation chapter.

### 2.3 Rigid Body Dynamics

Rigid-body equations of motions are derived from Euler's first and second axioms. If we denote  $r_G = [x_G \ y_G \ z_G]^T$  as the distance from the origin of the body-fixed

coordinate system to the vehicle's center of gravity,  $m$  as the mass of the body,  $f_b = [X_b \ Y_b \ Z_b]^T$  as the force vector referred to the body's center of gravity, the rigid-body equation of translational motion can be expressed as [3]:

$$m(\dot{v}_{eb}^b + \omega_{eb}^b \times v_{eb}^e + \dot{\omega}_{eb}^b \times r_G + \omega_{eb}^b \times (\omega_{eb}^b \times r_G)) = f_b \quad (2.5)$$

Similarly if we denote  $I_b$  as the body's inertia tensor and  $m_b = [K_b \ M_b \ N_b]^T$  as the moment vector referred to body's center of gravity, the rigid-body equation of rotational motion can be expressed as [3]:

$$I_b \dot{\omega}_{eb}^b + \omega_{eb}^b \times (I_b \omega_{eb}^b) + m r_G \times (\dot{v}_{eb}^b + \omega_{eb}^b \times v_{eb}^e) = m_b \quad (2.6)$$

Using these equations general 6 DOF rigid-body equations of motion can be written in component form as following:

$$m \left[ \dot{u}_{eb}^b - v_{eb}^b r_{eb}^b + w_{eb}^b q_{eb}^b - x_G (q_{eb}^{b^2} + r_{eb}^{b^2}) + y_G (p_{eb}^b q_{eb}^b - \dot{r}_{eb}^b) + z_G (p_{eb}^b r_{eb}^b + \dot{q}_{eb}^b) \right] = X_b \quad (2.7)$$

$$m \left[ \dot{v}_{eb}^b - w_{eb}^b p_{eb}^b + u_{eb}^b r_{eb}^b - y_G (r_{eb}^{b^2} + p_{eb}^{b^2}) + z_G (q_{eb}^b r_{eb}^b - \dot{p}_{eb}^b) + x_G (q_{eb}^b p_{eb}^b + \dot{r}_{eb}^b) \right] = Y_b \quad (2.8)$$

$$m \left[ \dot{w}_{eb}^b - u_{eb}^b q_{eb}^b + v_{eb}^b p_{eb}^b - z_G (p_{eb}^{b^2} + q_{eb}^{b^2}) + x_G (r_{eb}^b p_{eb}^b - \dot{q}_{eb}^b) + y_G (r_{eb}^b q_{eb}^b + \dot{p}_{eb}^b) \right] = Z_b \quad (2.9)$$

$$I_x \dot{p}_{eb}^b + (I_z - I_y) q_{eb}^b r_{eb}^b - (\dot{r}_{eb}^b + p_{eb}^b q_{eb}^b) I_{xz} + (r_{eb}^{b^2} - q_{eb}^{b^2}) I_{yz} + (p_{eb}^b r_{eb}^b - \dot{q}_{eb}^b) I_{xy} + m [y_G (\dot{w}_{eb}^b - u_{eb}^b q_{eb}^b + v_{eb}^b p_{eb}^b) - z_G (\dot{v}_{eb}^b - w_{eb}^b p_{eb}^b + u_{eb}^b r_{eb}^b)] = K_b \quad (2.10)$$

$$I_y \dot{q}_{eb}^b + (I_x - I_z) r_{eb}^b p_{eb}^b - (\dot{p}_{eb}^b + q_{eb}^b r_{eb}^b) I_{xy} + (p_{eb}^{b^2} - r_{eb}^{b^2}) I_{zx} + (q_{eb}^b p_{eb}^b - \dot{r}_{eb}^b) I_{yz} + m [z_G (\dot{u}_{eb}^b - v_{eb}^b r_{eb}^b + w_{eb}^b q_{eb}^b) - x_G (\dot{w}_{eb}^b - u_{eb}^b q_{eb}^b + v_{eb}^b p_{eb}^b)] = M_b \quad (2.11)$$

$$I_z \dot{r}_{eb}^b + (I_y - I_x) p_{eb}^b q_{eb}^b - (\dot{q}_{eb}^b + p_{eb}^b r_{eb}^b) I_{yz} + (q_{eb}^{b^2} - p_{eb}^{b^2}) I_{xy} + (r_{eb}^b q_{eb}^b - \dot{p}_{eb}^b) I_{zx} + m [x_G (\dot{v}_{eb}^b - w_{eb}^b p_{eb}^b + u_{eb}^b r_{eb}^b) - y_G (\dot{u}_{eb}^b - v_{eb}^b r_{eb}^b + w_{eb}^b q_{eb}^b)] = N_b \quad (2.12)$$

If we define body-fixed and body-referenced linear and angular velocity vector as  $v_{eb}^b = [v_{eb}^b \quad \omega_{eb}^b]$  and external forces and moments vector as  $\tau_{RB} = [f_b \quad m_b]$ , then 6 DOF rigid-body equations of motion can be expressed in a more compact form as:

$$M_{RB}\dot{v}_{eb}^b + C_{RB}(v_{eb}^b)v_{eb}^b = \tau_{RB} \quad (2.13)$$

Here,  $M_{RB}$  is the rigid-body inertia matrix, and can be uniquely parameterized as [3]:

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_x & -I_{xy} & -I_{xz} \\ mz_G & 0 & -mx_G & -I_{yx} & I_y & -I_{yz} \\ -my_G & mx_G & 0 & -I_{zx} & -I_{zy} & I_z \end{bmatrix} \quad (2.14)$$

$C_{RB}$  is the matrix representation of Coriolis and centripetal forces. Even though there is no unique parameterization for  $C_{RB}$  matrix, it can be expressed in skew-symmetric form as [3]:

$$C_{RB} \triangleq \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -m(y_G q_{eb}^b + z_G r_{eb}^b) & m(y_G p_{eb}^b + w_{eb}^b) & m(z_G p_{eb}^b - v_{eb}^b) & 0 & 0 & 0 \\ m(x_G q_{eb}^b - w_{eb}^b) & -m(z_G r_{eb}^b + x_G p_{eb}^b) & m(z_G q_{eb}^b + u_{eb}^b) & 0 & 0 & 0 \\ m(x_G r_{eb}^b + v_{eb}^b) & m(y_G r_{eb}^b - u_{eb}^b) & -m(x_G p_{eb}^b + y_G q_{eb}^b) & 0 & 0 & 0 \\ m(y_G q_{eb}^b + z_G r_{eb}^b) & -m(x_G q_{eb}^b - w_{eb}^b) & -m(x_G r_{eb}^b + v_{eb}^b) & 0 & 0 & 0 \\ -m(y_G p_{eb}^b + w_{eb}^b) & m(z_G r_{eb}^b + x_G p_{eb}^b) & -m(y_G r_{eb}^b - u_{eb}^b) & 0 & 0 & 0 \\ -m(z_G p_{eb}^b - v_{eb}^b) & -m(z_G q_{eb}^b + u_{eb}^b) & m(x_G p_{eb}^b + y_G q_{eb}^b) & 0 & 0 & 0 \\ 0 & -I_{yz} q_{eb}^b - I_{xz} p_{eb}^b + I_z r_{eb}^b & I_{yz} r_{eb}^b + I_{xy} p_{eb}^b - I_y q_{eb}^b & 0 & 0 & 0 \\ I_{yz} q_{eb}^b + I_{xz} p_{eb}^b - I_z r_{eb}^b & 0 & -I_{xz} r_{eb}^b - I_{xy} q_{eb}^b + I_x p_{eb}^b & 0 & 0 & 0 \\ -I_{yz} r_{eb}^b - I_{xy} p_{eb}^b + I_y q_{eb}^b & I_{xz} r_{eb}^b + I_{xy} q_{eb}^b - I_x p_{eb}^b & 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.15)$$

## 2.4 Hydrodynamic Effects

### 2.4.1 Added Mass

Various forces and moments are exerted on a vehicle due to its motion in underwater. This section briefly describes the major hydrodynamic effects and their contribution to the equations of motion

When a rigid-body surrounded by some fluid is moving, the surrounding fluid is accelerated by the motion of that body. The inertia of the fluid causes a reaction force applied on the body and it is called the added mass contribution. The added mass is a function of geometry of the body. The hydrodynamic added mass force  $X_A$  along x-axis, due to acceleration  $\dot{u}$  can be written as [6]:

$$X_A = X_{\dot{u}}\dot{u} \quad (2.16)$$

where

$$X_{\dot{u}} \triangleq \frac{\partial X}{\partial \dot{u}} \quad (2.17)$$

Equation (2.17) can be generalized for objects having 6 DOF motion to define the  $6 \times 6$  added mass and inertia matrix  $M_A$  as:

$$M_A \triangleq - \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \quad (2.18)$$

Since ULISAR underwater vehicle has near top-bottom and port-starboard symmetry, the contributions from the off-diagonal elements can be reduced. The simplified added mass matrix can be expressed as:

$$M_A \cong - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{\dot{v}} & 0 & 0 & 0 & Y_{\dot{r}} \\ 0 & 0 & Z_{\dot{w}} & 0 & Z_{\dot{q}} & 0 \\ 0 & 0 & 0 & K_{\dot{p}} & 0 & 0 \\ 0 & 0 & M_{\dot{w}} & 0 & M_{\dot{q}} & 0 \\ 0 & N_{\dot{v}} & 0 & 0 & 0 & N_{\dot{r}} \end{bmatrix} \quad (2.19)$$

Approximating the vehicle hull shape by an ellipsoid, the axial added mass term  $X_{\dot{u}}$  can be estimated by the following empirical formula [7]:

$$X_{\dot{u}} = -\frac{4\alpha\rho\pi}{3} \left(\frac{l}{2}\right) \left(\frac{d}{2}\right)^2 \quad (2.20)$$

where  $\rho$  is the water density,  $l$  and  $d$  is the length and diameter of the vehicle respectively.  $\alpha$  is an empirical parameter, and it is measured by Blevins [8] as given in Table 2.2:

Table 2.2 Axial added mass empirical parameter  $\alpha$  [8]

$(l/d)$	$\alpha$
1.0	0.5000
1.5	0.3038
2.0	0.2100
2.5	0.1563
3.0	0.1220
5.0	0.05912
7.0	0.03585

To estimate the cross flow added mass terms strip theory can be utilized. Strip theory is used to estimate 3D hydrodynamic coefficients for slender bodies. The approach is dividing the body into a number of 2D strips, calculating 2D coefficients for each strip, and integrating this over the length of the body to obtain 3D coefficients.

The added mass per unit length of a cylindrical strip is given as [9]:

$$m_a(x) = \pi\rho b(x)^2 \quad (2.21)$$

where  $b(x)$  is the width of the vehicle as a function of the vehicle's length.

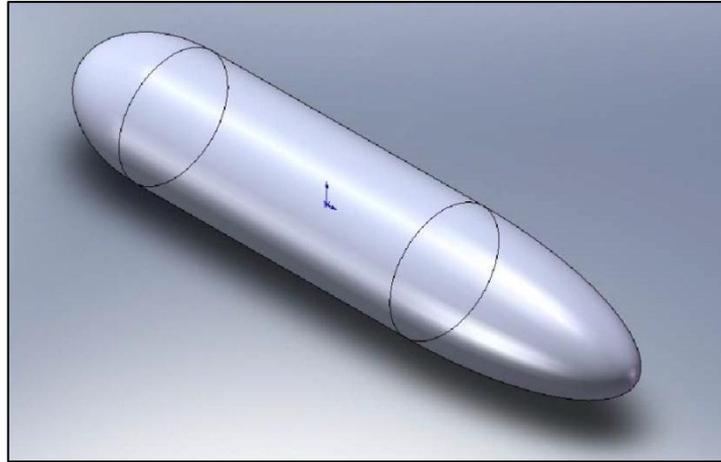


Figure 2.4 Bare hull of the vehicle [10]

Figure 2.4 shows the bare hull shape of ULISAR underwater vehicle without the wings. This geometry can be approximated as a combination of two ellipsoids with different eccentricity and a cylinder in between. Because of this structure, the body is divided into three sections and the strip theory is applied for each section separately. Figure 2.5 shows the vertical cross-section of the body with dimensions of the bare hull.

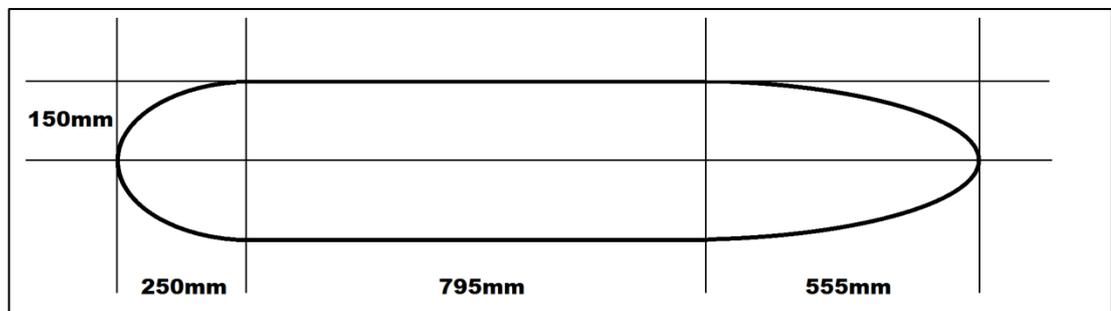


Figure 2.5 Dimensions of the bare hull

Using this structure  $b(x)$  can be calculated by:

$$b(x) = \begin{cases} \sqrt{0.15^2 - \left(\frac{0.15}{0.25}\right)^2 (x - 0.25)^2}, & 0 \leq x \leq 0.25, & L_1 \\ 0.15, & 0.25 < x < 1.045, & L_2 \\ \sqrt{0.15^2 - \left(\frac{0.15}{0.555}\right)^2 (x - 1.045)^2}, & 1.045 \leq x \leq 1.60, & L_3 \end{cases} \quad (2.22)$$

Then the cross flow added mass terms can be calculated by the following theoretical equations [7]:

$$Y_{\dot{v}} = - \int_0^L m_a(x) dx = - \int_0^{L_1} m_a(x) dx - \int_{L_1}^{L_2} m_a(x) dx - \int_{L_2}^{L_3} m_a(x) dx \quad (2.23)$$

$$M_{\dot{w}} = - \int_0^L m_a(x) x dx = - \int_0^{L_1} m_a(x) x dx - \int_{L_1}^{L_2} m_a(x) x dx - \int_{L_2}^{L_3} m_a(x) x dx \quad (2.24)$$

$$M_{\dot{q}} = - \int_0^L m_a(x) x^2 dx = - \int_0^{L_1} m_a(x) x^2 dx - \int_{L_1}^{L_2} m_a(x) x^2 dx - \int_{L_2}^{L_3} m_a(x) x^2 dx \quad (2.25)$$

The remaining cross flow terms are found by the symmetry:

$$Z_{\dot{w}} = Y_{\dot{v}} \quad N_{\dot{r}} = M_{\dot{q}} \quad N_{\dot{v}} = M_{\dot{w}} = -Z_{\dot{q}} = -Y_{\dot{r}} \quad (2.26)$$

Even though, for the axis symmetric body cross sections, the rolling added mass  $K_{\dot{p}}$  is zero, the rolling added mass contribution for the section of the body containing the stabilization wings is non-zero. Figure 2.6 shows the hull shape with wings of ULISAR underwater vehicle.

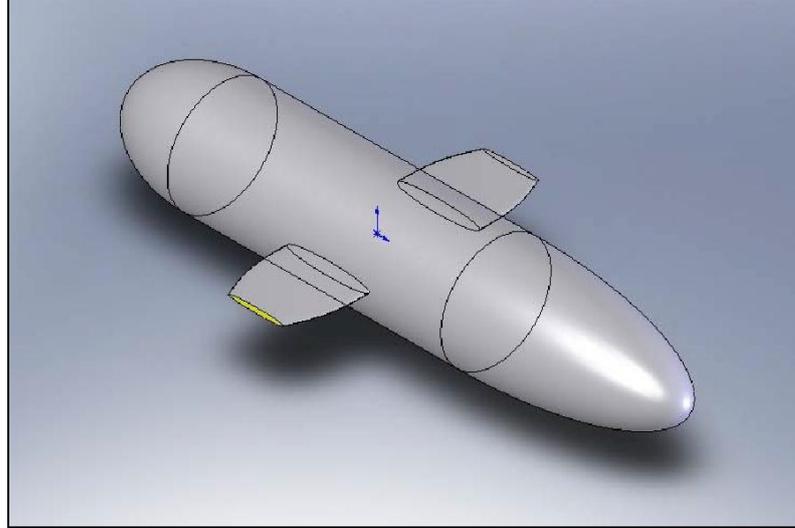


Figure 2.6 Hull with wings [10]

Blevins [8] gives an empirical formula for the added mass of a rolling circle with fins as following:

$$K_{\dot{p}} = - \int_0^{l_f} \frac{2}{\pi} \rho a^4 dx \quad (2.27)$$

The added mass effect also creates an added Coriolis and centripetal contribution. Added mass Coriolis and centripetal forces matrix  $C_A$  can be parameterized such that it is skew-symmetrical, as follows [3]:

$$C_A(v_{eb}^b) = \begin{bmatrix} 0 & 0 & 0 & 0 & -a_3 & a_2 \\ 0 & 0 & 0 & a_3 & 0 & -a_1 \\ 0 & 0 & 0 & -a_2 & a_1 & 0 \\ 0 & -a_3 & a_2 & 0 & -b_3 & b_2 \\ a_3 & 0 & -a_1 & b_3 & 0 & -b_1 \\ -a_2 & a_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix} \quad (2.28)$$

where

$$a_1 = X_{\dot{u}} u_{eb}^b + X_{\dot{v}} v_{eb}^b + X_{\dot{w}} w_{eb}^b + X_{\dot{p}} p_{eb}^b + X_{\dot{q}} q_{eb}^b + X_{\dot{r}} r_{eb}^b \quad (2.29)$$

$$a_2 = X_{\dot{v}}u_{eb}^b + Y_{\dot{v}}v_{eb}^b + Y_{\dot{w}}w_{eb}^b + Y_{\dot{p}}p_{eb}^b + Y_{\dot{q}}q_{eb}^b + Y_{\dot{r}}r_{eb}^b \quad (2.30)$$

$$a_3 = X_{\dot{w}}u_{eb}^b + Y_{\dot{w}}v_{eb}^b + Z_{\dot{w}}w_{eb}^b + Z_{\dot{p}}p_{eb}^b + Z_{\dot{q}}q_{eb}^b + Z_{\dot{r}}r_{eb}^b \quad (2.31)$$

$$b_1 = X_{\dot{p}}u_{eb}^b + Y_{\dot{p}}v_{eb}^b + Z_{\dot{p}}w_{eb}^b + K_{\dot{p}}p_{eb}^b + K_{\dot{q}}q_{eb}^b + K_{\dot{r}}r_{eb}^b \quad (2.32)$$

$$b_2 = X_{\dot{q}}u_{eb}^b + Y_{\dot{q}}v_{eb}^b + Z_{\dot{q}}w_{eb}^b + K_{\dot{q}}p_{eb}^b + M_{\dot{q}}q_{eb}^b + M_{\dot{r}}r_{eb}^b \quad (2.33)$$

$$b_3 = X_{\dot{r}}u_{eb}^b + Y_{\dot{r}}v_{eb}^b + Z_{\dot{r}}w_{eb}^b + K_{\dot{r}}p_{eb}^b + M_{\dot{r}}q_{eb}^b + N_{\dot{r}}r_{eb}^b \quad (2.34)$$

## 2.4.2 Hydrodynamic Damping

There are four main sources of hydrodynamic damping:

- **Potential damping:** Potential damping, also known as the radiation-induced potential damping is caused by the energy carried away by generated surface waves when the body is forced to oscillate with the wave excitation frequency. However for underwater vehicles operating at deep sea, contribution of this effect is negligible compared to other hydrodynamic effects.
- **Skin friction:** Skin friction can be studied in two parts: Linear skin friction due to laminar boundary layer theory, and quadratic (nonlinear) skin friction due to turbulent boundary layer theory. Linear skin friction is important when the vehicle has a low frequency motion, whereas the quadratic skin friction generates a high-frequency contribution. Even though skin friction is an important part of damping, in depth analysis is beyond the scope of this thesis. More details can be found in [9].
- **Wave drift damping:** Wave drift damping is the added dissipative term for surface ships moving ahead in surface waves. This effect doesn't have a contribution for vehicles completely submerged into water; therefore it is neglected for underwater vehicles.

- Vortex shedding: Damping due to vortex shedding is the resistance force acting on a body when moving in a fluid due to viscosity. The viscous damping due to vortex shedding can be formulated as [3]:

$$f(U) = -\frac{1}{2}\rho C_D(Rn)A|U|U \quad (2.35)$$

Here  $U$  is the velocity of the vehicle,  $A$  is the projected cross-sectional area,  $C_D(Rn)$  is the drag-coefficient based on the representative area and  $\rho$  is the water density. ( $\nu = 1.05 \times 10^{-6}$  for sea water with 20°C and salinity of 3.5%) Drag coefficient depends on the *Reynolds number* which is a function of velocity, characteristic length of the body ( $D$ ), and kinematic viscosity coefficient ( $\nu$ ).

$$Rn = \frac{UD}{\nu} \quad (2.36)$$

Appendix A gives the drag coefficients of different geometric object as a function of Reynolds numbers.

Considering top-bottom and port-starboard symmetry of our underwater vehicle, the following matrix can be used to represent 6 DOF damping equations:

$$D(\mathbf{v}_{eb}^b) = \begin{bmatrix} X_{|u|u}|u_{eb}^b|u_{eb}^b & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{|v|v}|v_{eb}^b|v_{eb}^b & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{|w|w}|w_{eb}^b|w_{eb}^b & 0 & 0 & 0 \\ 0 & 0 & 0 & M_{|w|w}|w_{eb}^b|w_{eb}^b & 0 & 0 \\ 0 & 0 & Z_{|q|q}|q_{eb}^b|q_{eb}^b & 0 & 0 & 0 \\ 0 & Y_{|r|r}|r_{eb}^b|r_{eb}^b & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & N_{|v|v}|v_{eb}^b|v_{eb}^b & 0 \\ 0 & 0 & 0 & M_{|w|w}|w_{eb}^b|w_{eb}^b & 0 & 0 \\ K_{|p|p}|p_{eb}^b|p_{eb}^b & 0 & 0 & 0 & 0 & 0 \\ 0 & M_{|q|q}|q_{eb}^b|q_{eb}^b & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & N_{|r|r}|r_{eb}^b|r_{eb}^b & 0 \end{bmatrix} \quad (2.37)$$

Here, the axial drag is calculated using the viscous damping formula:

$$X_{|u|u} = -\frac{1}{2}\rho C_D(Rn)A \quad (2.38)$$

where  $C_D$  is estimated as 0.27.

Vehicle cross flow drag is due to hull and wing drag forces. Calculation of the cross flow damping terms is similar to the calculation of the added mass terms; first the coefficients belong to 2D cross-sections are found, and then they are integrated over the length of the body. Cross flow drag terms can be calculated by [7]:

$$Y_{|v|v} = Z_{|w|w} = -\frac{1}{2}\rho C_{Dc}(Rn) \int_0^L 2b(x)dx - 3\left(\frac{1}{2}\rho S_{fin}C_{Df}\right) \quad (2.39)$$

$$M_{|w|w} = N_{|v|v} = -\frac{1}{2}\rho C_{Dc}(Rn) \int_0^L 2b(x)xdx - 3l_{fin}\left(\frac{1}{2}\rho S_{fin}C_{Df}\right) \quad (2.40)$$

$$Y_{|r|r} = Z_{|q|q} = -\frac{1}{2}\rho C_{Dc}(Rn) \int_0^L 2b(x)x|x|dx - 3l_{fin}^2\left(\frac{1}{2}\rho S_{fin}C_{Df}\right) \quad (2.41)$$

$$M_{|q|q} = N_{|r|r} = -\frac{1}{2}\rho C_{Dc}(Rn) \int_0^L 2b(x)x^3dx - 3l_{fin}^3\left(\frac{1}{2}\rho S_{fin}C_{Df}\right) \quad (2.42)$$

Here  $S_{fin}$  and  $l_{fin}$  are the area and length of the wings respectively,  $C_{Dc}$  is the cross flow drag coefficient of the hull and  $C_{Df}$  is the cross flow drag coefficient of the wings. In the calculations  $C_{Dc}$  is estimated as 1.1 and  $C_{Df}$  is estimated as 0.32.

Similar to the rolling added mass term, the rolling drag  $K_{|p|p}$  is due to the stabilization wings, and can be calculated by [7]:

$$K_{|p|p} = -3\left(\frac{1}{2}\rho S_{fin}C_{df}\right)r_{mean}^3 \quad (2.43)$$

where  $r_{mean}$  is the mean wing width from the vehicle centerline.

## 2.5 Hydrostatic Effects

The gravitational force  $f_G$ , acting on the center of gravity  $r_G = [x_G \ y_G \ z_G]^T$ , and the buoyant force  $f_B$ , acting on the center of buoyancy  $r_B = [x_B \ y_B \ z_B]^T$  are hydrostatic effects, i.e., they are not a function of a relative movement between body and fluid. However they are functions of the orientation of the body in the fluid. According to SNAME notation, the weight of a submerged body is defined as  $W = mg$ , while the buoyancy is defined as  $B = \rho g \nabla$ . Here  $m$  is the mass of the vehicle,  $\nabla$  is the volume of fluid displaced by the vehicle,  $g$  is the gravitational acceleration and  $\rho$  is the fluid density. Utilizing the earth-to-body coordinate transformation matrix  $J_1^{-1}(\Psi_{eb})$ , the weight and the buoyancy forces can be expressed in the body-fixed coordinate system as [3]:

$$f_G(\Psi_{eb}) = J_1^{-1}(\Psi_{eb}) \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix}, \quad f_b(\Psi_{eb}) = -J_1^{-1}(\Psi_{eb}) \begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix} \quad (2.44)$$

Consequently gravity and buoyancy vector can be formed using following forces and moments:

$$g(\Psi_{eb}) = - \begin{bmatrix} f_G(\Psi_{eb}) + f_b(\Psi_{eb}) \\ r_G \times f_G(\Psi_{eb}) + r_B \times f_b(\Psi_{eb}) \end{bmatrix} \quad (2.45)$$

Equation 2.45 can be expanded to obtain a 6 DOF representation for gravity and buoyancy:

$$g(\Psi_{eb}) = \begin{bmatrix} -(W - B)\sin\theta_{eb} \\ (W - B)\cos\theta_{eb}\sin\phi_{eb} \\ (W - B)\cos\theta_{eb}\cos\phi_{eb} \\ (y_G W - y_B B)\cos\theta_{eb}\cos\phi_{eb} - (z_G W - z_B B)\cos\theta_{eb}\sin\phi_{eb} \\ -(z_G W - z_B B)\sin\theta_{eb} - (x_G W - x_B B)\cos\theta_{eb}\cos\phi_{eb} \\ (x_G W - x_B B)\cos\theta_{eb}\sin\phi_{eb} + (y_G W - y_B B)\sin\theta_{eb} \end{bmatrix} \quad (2.46)$$

## 2.6 Thrusters' Dynamics

The relationship between the force/moment acting on the vehicle  $\tau_v$  and the control input of the thrusters  $u_v$  depends on some structural variables such as: the tunnel cross-sectional area, the tunnel length, the volumetric flow rate between input and output of the thrusters, the propeller diameter and the density of the water, therefore it is highly nonlinear. In depth analysis of thruster's behavior is out of scope of this thesis but can be found in [11] in detail. A common simplification is to consider a linear relationship, yielding [6]:

$$\tau_v = B_v u_v \quad (2.47)$$

where  $B_v$  is a constant matrix known as the Thruster Control Matrix (TCM). ULISAR underwater vehicle has 4 thrusters; two vertical thrusters on front and rear and two forward thrusters on left and right as seen on Figure 2.7.

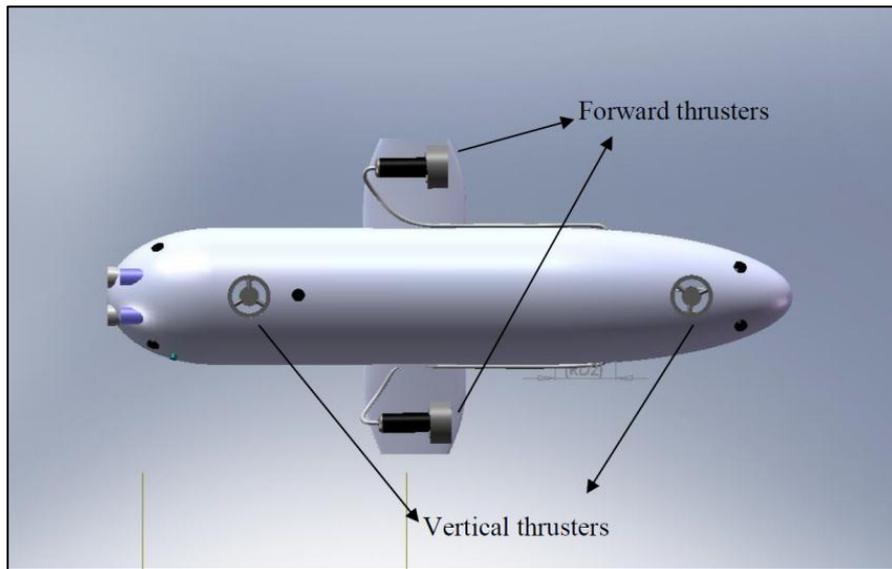


Figure 2.7 Thruster configuration – Bottom view [10]

In this configuration TCM can be expressed as:

$$B_v = \begin{bmatrix} B_{11} & B_{12} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & B_{33} & B_{34} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & B_{53} & B_{54} \\ B_{61} & B_{62} & 0 & 0 \end{bmatrix} \quad (2.48)$$

## 2.7 Equations of Motion

The kinematic equations, rigid-body dynamics, hydrodynamic and hydrostatic effects, and thrusters' dynamics can be put together to form the nonlinear equations of motion. The body-fixed vector representation of the nonlinear equations of motion can be written as:

$$M\dot{v}_{eb}^b + C(v_{eb}^b)v_{eb}^b + D(v_{eb}^b)v_{eb}^b + g(\Psi_{eb}) = \tau_v \quad (2.49)$$

$$\dot{\eta}_{eb}^e = J(\eta_{eb})v_{eb}^b \quad (2.50)$$

where

$$M = M_{RB} + M_A \quad (2.51)$$

$$C(v_{eb}^b) = C_{RB}(v_{eb}^b) + C_A(v_{eb}^b) \quad (2.52)$$

$$\eta_{eb}^e = \begin{bmatrix} r_{eb}^e \\ \Psi_{eb} \end{bmatrix}, \quad v_{eb}^b = \begin{bmatrix} v_{eb}^b \\ \omega_{eb}^b \end{bmatrix} \quad (2.53)$$

Considering that our vehicle has near top-bottom and port-starboard symmetry, and the origin of the body-fixed coordinate frame is chosen as the center of gravity ( $r_G = [0 \ 0 \ 0]^T$ ), and center of buoyancy is above center of gravity ( $r_B = [0 \ 0 \ z_B]^T$ ), the component matrices can be simplified into:

$$M = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 & 0 & 0 & -Y_{\dot{r}} \\ 0 & 0 & m - Z_{\dot{w}} & 0 & -Z_{\dot{q}} & 0 \\ 0 & 0 & 0 & I_x - K_{\dot{p}} & 0 & 0 \\ 0 & 0 & -M_{\dot{w}} & 0 & I_y - M_{\dot{q}} & 0 \\ 0 & -N_{\dot{v}} & 0 & 0 & 0 & I_z - N_{\dot{r}} \end{bmatrix} \quad (2.54)$$

$$C(v_{eb}^b) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & mw_{eb}^b - Z_{\dot{w}}w_{eb}^b & -mv_{eb}^b + Y_{\dot{v}}v_{eb}^b \\ -mw_{eb}^b + Z_{\dot{w}}w_{eb}^b & 0 & mu_{eb}^b - X_{\dot{u}}u_{eb}^b \\ mv_{eb}^b - Y_{\dot{v}}v_{eb}^b & -mu_{eb}^b + X_{\dot{u}}u_{eb}^b & 0 \\ 0 & mw_{eb}^b - Z_{\dot{w}}w_{eb}^b & -mv_{eb}^b + Y_{\dot{v}}v_{eb}^b \\ -mw_{eb}^b + Z_{\dot{w}}w_{eb}^b & 0 & mu_{eb}^b - X_{\dot{u}}u_{eb}^b \\ mv_{eb}^b - Y_{\dot{v}}v_{eb}^b & -mu_{eb}^b + X_{\dot{u}}u_{eb}^b & 0 \\ 0 & I_z r_{eb}^b - N_{\dot{r}}r_{eb}^b & -I_y q_{eb}^b + M_{\dot{q}}q_{eb}^b \\ -I_z r_{eb}^b + N_{\dot{r}}r_{eb}^b & 0 & I_x p_{eb}^b - K_{\dot{p}}p_{eb}^b \\ I_y q_{eb}^b - M_{\dot{q}}q_{eb}^b & -I_x p_{eb}^b + K_{\dot{p}}p_{eb}^b & 0 \end{bmatrix} \quad (2.55)$$

$$D(v_{eb}^b) = \begin{bmatrix} X_{|u|u}|u_{eb}^b|u_{eb}^b & 0 & 0 \\ 0 & Y_{|v|v}|v_{eb}^b|v_{eb}^b & 0 \\ 0 & 0 & Z_{|w|w}|w_{eb}^b|w_{eb}^b \\ 0 & 0 & 0 \\ 0 & 0 & Z_{|q|q}|q_{eb}^b|q_{eb}^b \\ 0 & Y_{|r|r}|r_{eb}^b|r_{eb}^b & 0 \\ 0 & 0 & 0 \\ 0 & 0 & N_{|v|v}|v_{eb}^b|v_{eb}^b \\ 0 & M_{|w|w}|w_{eb}^b|w_{eb}^b & 0 \\ K_{|p|p}|p_{eb}^b|p_{eb}^b & 0 & 0 \\ 0 & M_{|q|q}|q_{eb}^b|q_{eb}^b & 0 \\ 0 & 0 & N_{|r|r}|r_{eb}^b|r_{eb}^b \end{bmatrix} \quad (2.56)$$

$$g(\Psi_{eb}) = \begin{bmatrix} -(W - B)\sin\theta_{eb} \\ (W - B)\cos\theta_{eb}\sin\phi_{eb} \\ (W - B)\cos\theta_{eb}\cos\phi_{eb} \\ z_B B \cos\theta_{eb}\sin\phi_{eb} \\ z_B B \sin\theta_{eb} \\ 0 \end{bmatrix} \quad (2.57)$$

When the equations of motion are examined certain deductions can be made. One of them is the effect of the wings. It is clear that due to added mass and damping contributions, wings help the passive stabilization of the vehicle in the roll axis. Besides, when the gravity and buoyancy matrix is examined, it can be seen that if  $z_B < 0$  (the center of buoyancy is above the center of gravity), gravitational force opposes to roll and pitch motions and helps the horizontal stabilization of the vehicle.

## CHAPTER 3

### CONTROLLER DESIGN

#### 3.1 State-space representation

Before we start the controller design, first we need to obtain a state-space representation for the equations of motion. 12-dimension state-space is chosen to represent the position and orientation of the vehicle resolved in earth-frame and the linear and angular velocities resolved in body-frame. The state  $x(t)$  is defined as:

$$x(t) = \begin{pmatrix} \eta_{eb}^e(t) \\ v_{eb}^b(t) \end{pmatrix} \quad (3.1)$$

Then the nonlinear Equations (2.49) and (2.50) can be written in state-space form as:

$$\dot{x} = f(x, u) \quad (3.2)$$

where

$$f(x, u) = \begin{pmatrix} J(\eta_{eb})v_{eb}^b \\ M^{-1}(\tau_v - C(v_{eb}^b)v_{eb}^b - D(v_{eb}^b)v_{eb}^b - g(\Psi_{eb})) \end{pmatrix} \quad (3.3)$$

Here the variable  $t$  is not shown for simplicity.

#### 3.2 Linearization

To apply linear control techniques afterwards, first  $f(x, u)$  must be linearized about an operating point  $(x_0, u_0)$ . Equation (3.2) can be expanded into a Taylor series about this operating point as follows [12]:

$$\begin{aligned} \dot{x} = & f(x_0, u_0) + \left[ \frac{\partial f}{\partial x}(x - x_0) + \frac{\partial f}{\partial u}(u - u_0) \right] + \frac{1}{2!} \left[ \frac{\partial^2 f}{\partial x^2}(x - x_0)^2 + \right. \\ & \left. 2 \frac{\partial^2}{\partial x \partial u}(x - x_0)(u - u_0) + \frac{\partial^2 f}{\partial u^2}(u - u_0)^2 \right] + \dots \end{aligned} \quad (3.4)$$

where the partial derivatives are evaluated at  $x = x_0$  and  $u = u_0$ . Near the operating point, i.e.,  $x - x_0$  and  $u - u_0$  is small, therefore the higher order terms may be neglected [12]. Thus the linear mathematical model in the neighborhood of the operating point can be expressed as:

$$\dot{x} = f(x_0, u_0) + \left. \frac{\partial f}{\partial x} \right|_{x=x_0, u=u_0} (x - x_0) + \left. \frac{\partial f}{\partial u} \right|_{x=x_0, u=u_0} (u - u_0) \quad (3.5)$$

To express Equation (3.5) in matrix form, we define the augmented state vector  $\bar{x}$  as:

$$\bar{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \quad (3.6)$$

With the augmented state vector in Equation (3.6), the linearized model can be written in matrix form as follows:

$$\dot{\bar{x}} = A\bar{x} + Bu \quad (3.7)$$

where the state matrix  $A$ , and input matrix  $B$  are as follows:

$$A = \begin{bmatrix} \left. \frac{\partial f}{\partial x} \right|_{x=x_0, u=u_0} & f(x_0, u_0) \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} \left. \frac{\partial f}{\partial u} \right|_{x=x_0, u=u_0} \\ 0 \end{bmatrix} \quad (3.8)$$

Again for simplicity, from now on,  $x$  will be used to represent the augmented state vector instead of  $\bar{x}$ .

### 3.3 Linear Quadratic Regulator (LQR)

Linear quadratic regulator problem is an important class of optimal control problems. For linear regulator problems the optimal control law can be found as a linear time-varying function of the system states. Furthermore under certain conditions the

optimal control law becomes time-invariant [13]. Those conditions will be discussed later on.

First we describe the problem by the linear state-equations as in Equation (3.7), and define the performance measure to be minimized as:

$$J = \frac{1}{2}[x(t_f) - r(t_f)]^T H[x(t_f) - r(t_f)] + \frac{1}{2} \int_{t_0}^{t_f} \{[x(t) - r(t)]^T Q(t)[x(t) - r(t)] + u^T(t)R(t)u(t)\} dt \quad (3.9)$$

where  $r(t)$  is the reference value of the state vector. Here the final time  $t_f$  is fixed,  $x(t_f)$  is free, the states and controls are not bounded,  $H$  and  $Q$  are real symmetric positive semi-definite matrices, and  $R$  is real symmetric and positive definite [13]. The physical interpretation to this performance measure is; we want the state vector to track the reference without an excessive expenditure of control effort. A diagram of the plant and controller is shown in Figure 3.1.

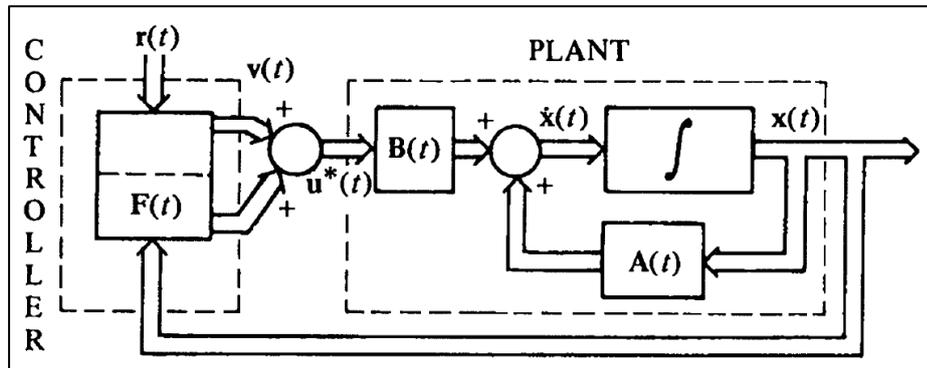


Figure 3.1 Plant and optimal feedback controller for linear tracking problems [13]

The optimal control law for this problem can be calculated by:

$$u^*(t) = -R^{-1}(t)B^T(t)K(t)x(t) - R^{-1}(t)B^T(t)s(t) \triangleq F(t)x(t) + v(t) \quad (3.10)$$

where  $K(t)$  and  $s(t)$  are solutions of the following Riccati differential equations:

$$\dot{K}(t) = -K(t)A(t) - A^T(t)K(t) - Q(t) + K(t)B(t)R^{-1}(t)B^T(t)K(t) \quad (3.11)$$

$$\dot{s}(t) = -[A^T(t) - K(t)B(t)R^{-1}(t)B^T(t)]s(t) + Q(t)r(t) \quad (3.12)$$

The derivations of Equations (3.10)-(3.12) can be found in [13]. The boundary conditions are:

$$K(t_f) = H \quad (3.13)$$

$$s(t_f) = -Hr(t_f) \quad (3.14)$$

Using the boundary conditions in the Equations (3.13) and (3.14), Equations (3.11) and (3.12) can be integrated from  $t_f$  to  $t_0$ , to find the values for  $K(t)$  and  $s(t)$ .

In our problem definition,  $A$ ,  $B$ ,  $Q$ ,  $R$ , and  $r$  are time-invariant and  $H = 0$ . Kalman [14] has shown that if the plant is completely controllable,  $H = 0$ , and  $A$ ,  $B$ ,  $Q$ , and  $R$  are constant matrices,  $K(t) \rightarrow K$  (a constant matrix) as  $t_f \rightarrow \infty$ . Kirk [13] also stated that, from a practical point of view, it may be feasible to use fixed control law even for processes of finite duration. Therefore in this study, the fixed optimal control law is used for the controllable subspace of the linearized system model.

In the implementation of the controller, first the system is linearized around its rated forward speed together with small perturbations on the remaining states. Then  $Q$  and  $R$  matrices are formed as diagonal and positive definite matrices. In the  $Q$  matrix, the entries which correspond to the forward speed, heave speed, yaw angle and pitch angle states are selected higher compared to other. Also, the entries of the  $R$  matrix are selected carefully not to exceed the limits of the thrusters.

After that the rank of the controllability matrix of  $(A, B)$  is calculated and found out that  $r < n$ , where  $r$  is the rank and  $n$  is the size of the matrix  $A$ . This means that  $(A, B)$  pair is not completely controllable. To use the optimal control theory above, using the Staircase Algorithm of Rosenbrock [15], the similarity transformation matrix  $T$  is formed, which transforms the state-space system represented by  $(A, B)$  into the controllability staircase form, such that:

$$\bar{A} = TAT^T, \quad \bar{B} = TB \tag{3.15}$$

$$\bar{A} = \begin{bmatrix} A_{uc} & 0 \\ A_{21} & A_c \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} 0 \\ B_c \end{bmatrix}$$

where  $(A_c, B_c)$  is controllable and all eigenvalues of  $A_{uc}$  are uncontrollable. This transformation is also applied to  $Q$  matrix to obtain  $Q_c$  and then  $Kc$  is calculated using  $A_c$ ,  $B_c$ ,  $Q_c$  and  $R$  matrices. The resultant  $Kc$  is transformed backed to the original form for consistency:

$$\bar{K} = \begin{bmatrix} 0 & 0 \\ 0 & K_c \end{bmatrix}, \quad K = T^T \bar{K} T \tag{3.16}$$

Using the boundary conditions in Equations (3.13) and (3.14), and the assumptions made, the vector  $s$  is calculated by:

$$s = -Kr \tag{3.17}$$

At the last step the optimal control law  $u^*(t)$  is calculated as in Equation (3.10) and applied as the input of the system. Figure 3.2-9 gives the results of step response simulations using this optimal control law, showing the states together with the inputs applied:

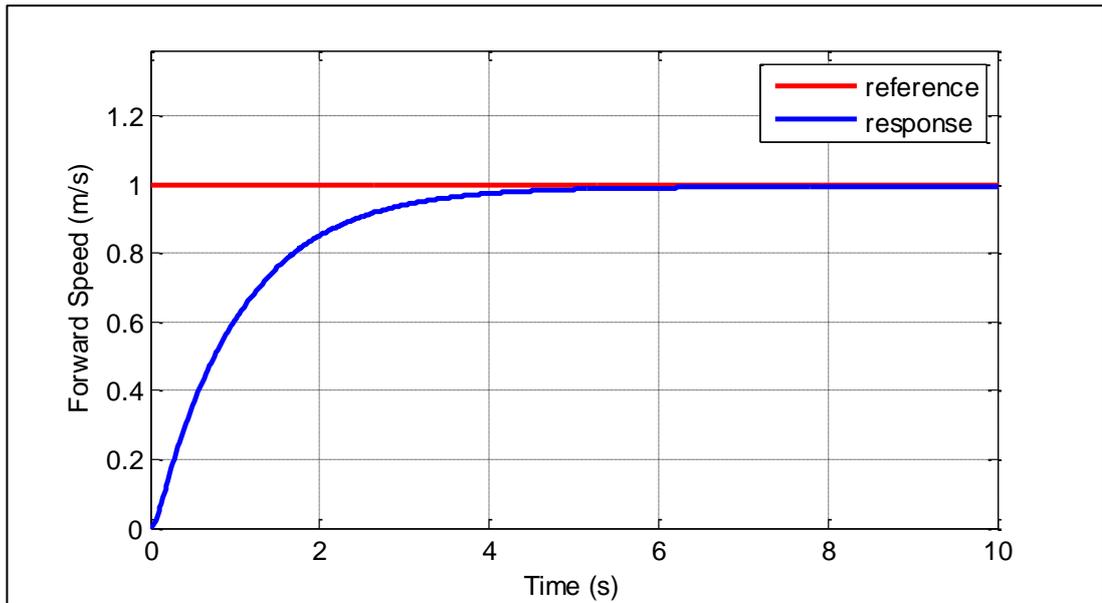


Figure 3.2 Forward speed step response

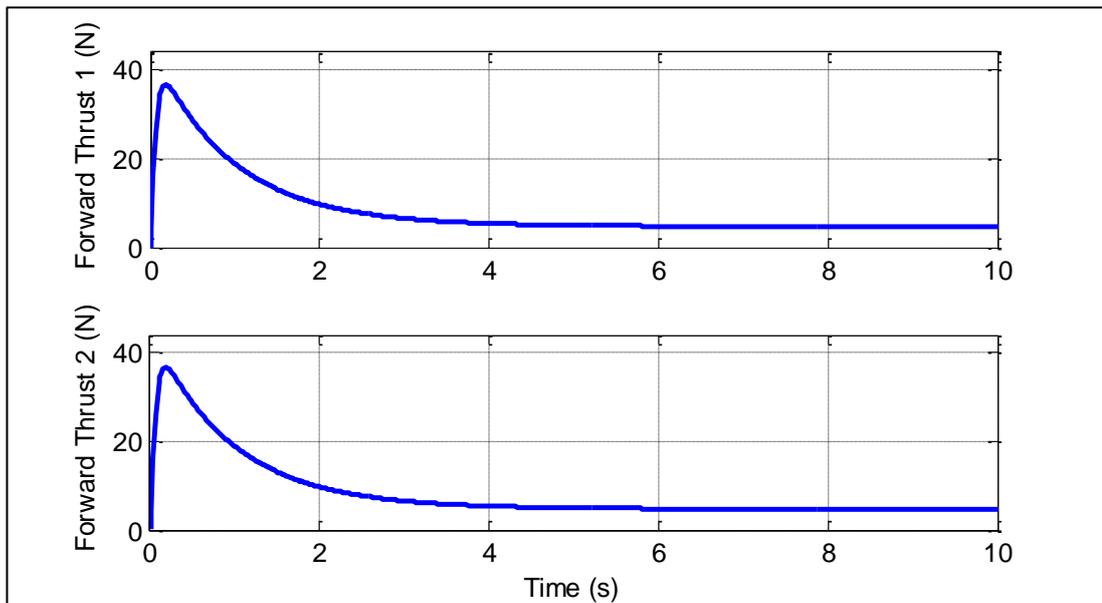


Figure 3.3 Forward thrusts for forward speed step response

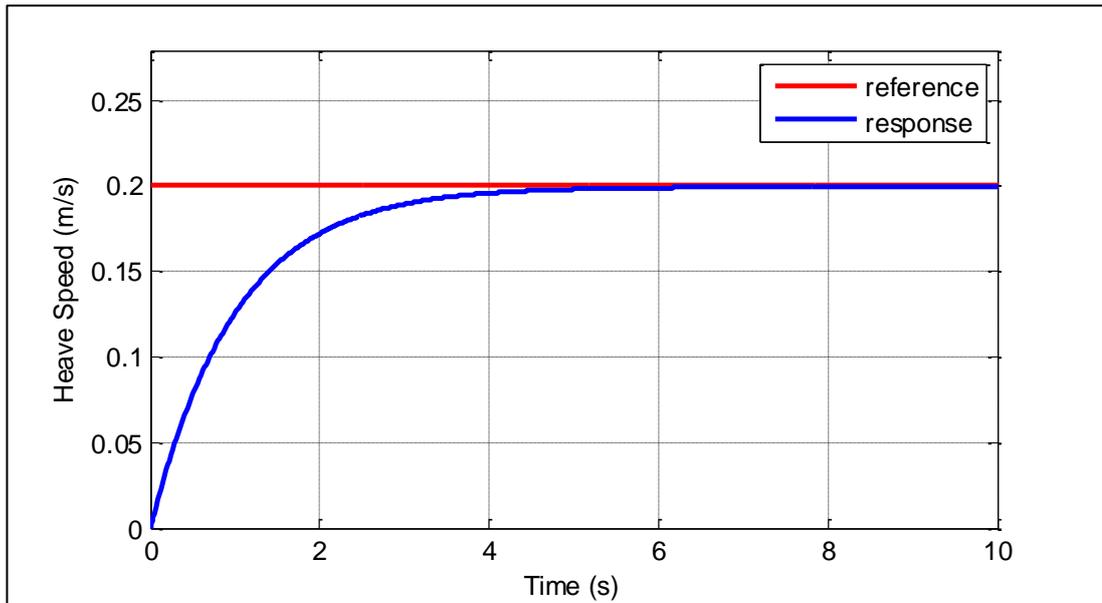


Figure 3.4 Heave speed step response

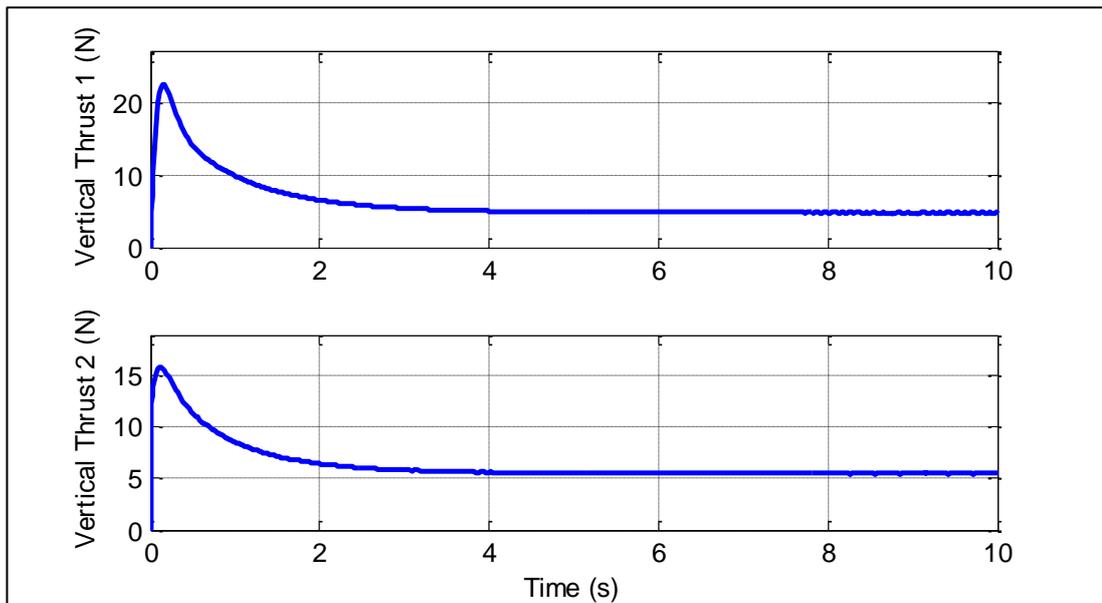


Figure 3.5 Vertical thrusts heave speed step response

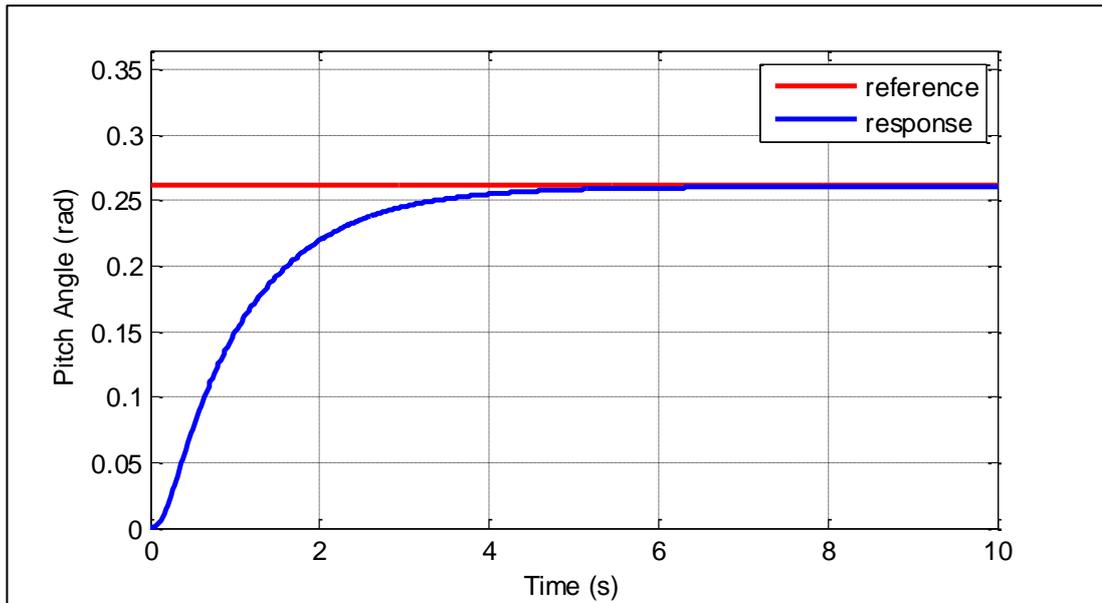


Figure 3.6 Pitch rotation step response

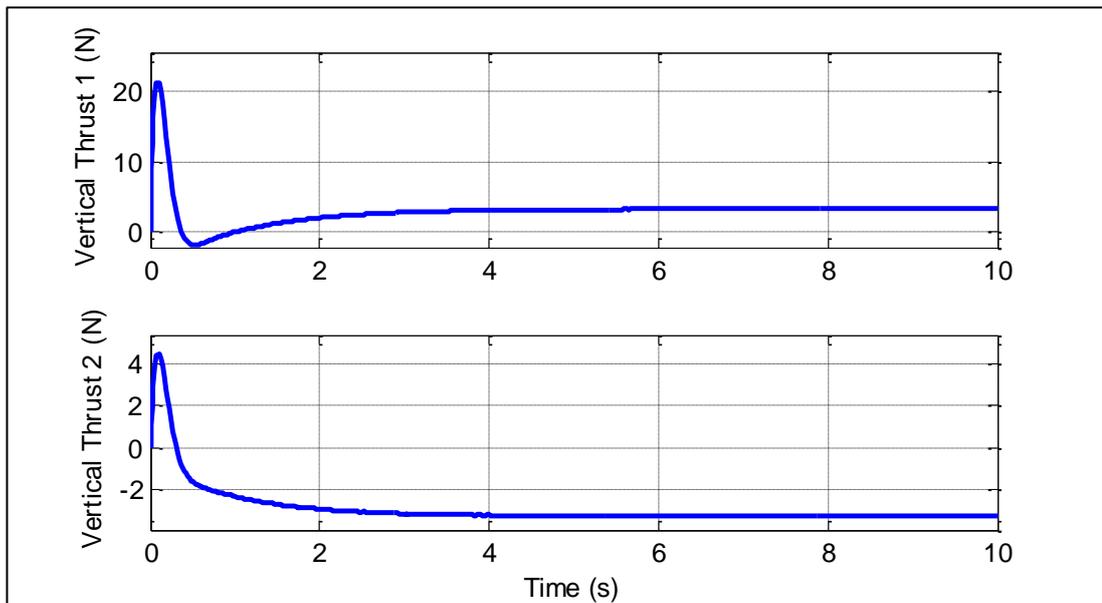


Figure 3.7 Vertical thrusts for step pitch rotation

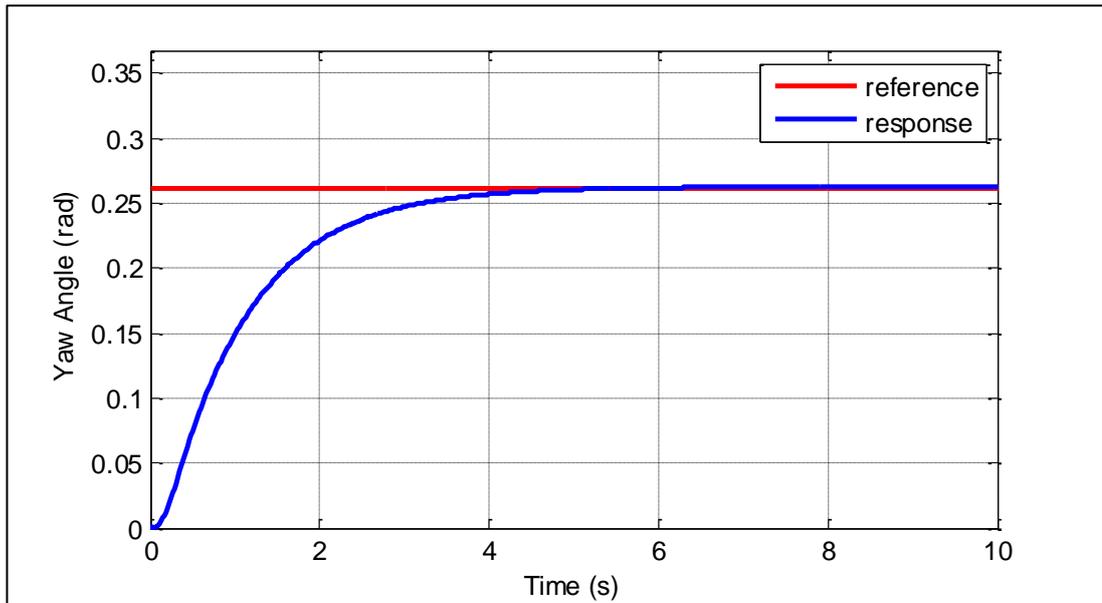


Figure 3.8 Yaw rotation step response

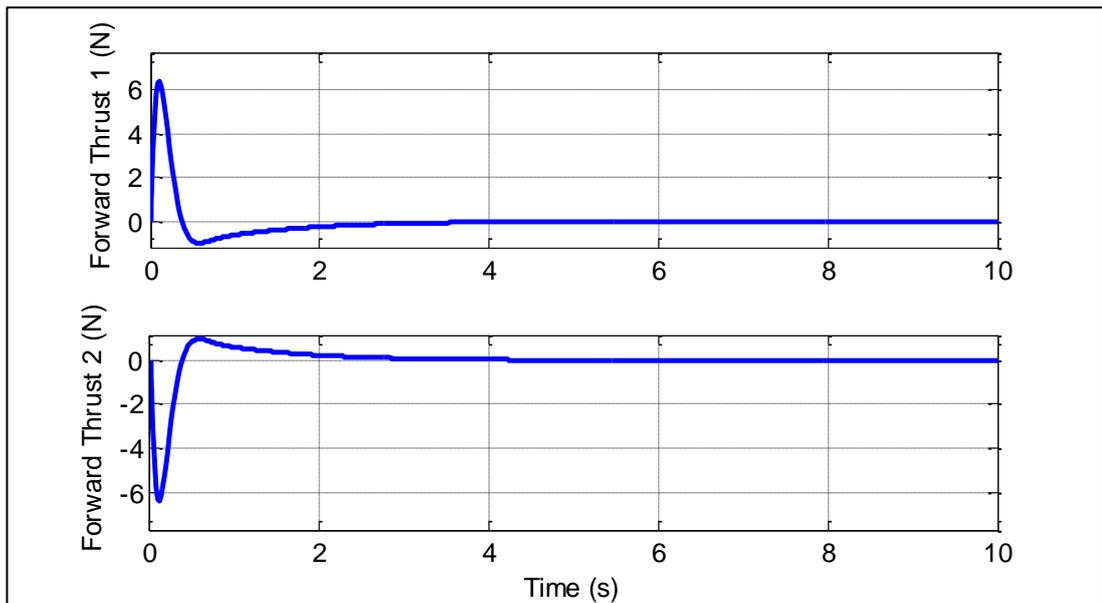


Figure 3.9 Forward thrusts for step yaw rotation

## CHAPTER 4

### GUIDANCE

#### 4.1 Introduction

The guidance system of an underwater vehicle is the key to achieve autonomous operation as well as to optimize energy, time etc. during the missions. In this study, the objective of the guidance algorithm is defined as to generate the references for the vehicle control system so that the vehicle can move through a set of waypoints in the given sequence. Some important AUV guidance laws found in the literature are [16]:

- Way Point Guidance by Line of Sight (LOS),
- Vision Based Guidance,
- Lyapunov Based Guidance,
- Guidance Using Chemical Signals,
- Proportional Navigation Guidance for AUV's,
- Guidance using Magnetometers for Cable Tracking,
- Electromagnetic Guidance.

Among the others, the way point guidance by LOS strategy is the most widely used, in fact, almost all other guidance laws in use, have some form of LOS guidance [16]. Due to its simplicity and ease of implementation, a 3D LOS guidance is chosen as the main guidance strategy also in this thesis study. After a successful implementation of the algorithm, some improvements are proposed to improve the drawbacks of this algorithm.

## 4.2 Way Point Guidance by Line of Sight (LOS)

As shown by Healey and Lienard [17], in 2D guidance can be achieved between two points  $[x_d(t_0), y_d(t_0)]$  and  $[x_d(t_f), y_d(t_f)]$  by splitting the path between those into a number of way points  $[x_d(k), y_d(k)]$  for  $k = 1, 2, \dots, N$  as shown in Figure 4.1:

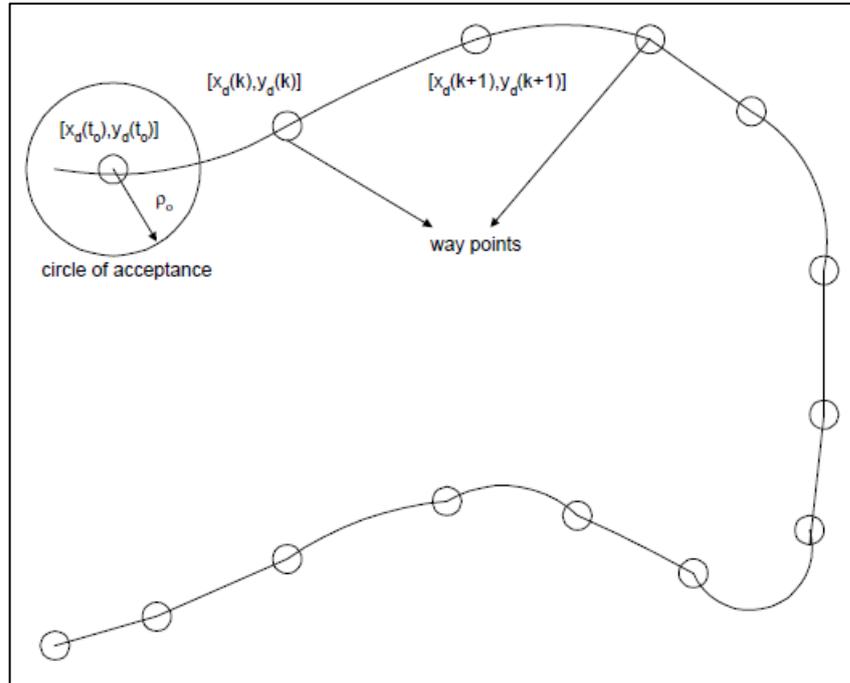


Figure 4.1 Way points and circle of acceptance [17]

In 3D implementation, guidance problem is redefined as splitting the path between two points  $[x_d(t_0), y_d(t_0), z_d(t_0)]$  and  $[x_d(t_f), y_d(t_f), z_d(t_f)]$  into a number of way points  $[x_d(k), y_d(k), z_d(k)]$  for  $k = 1, 2, \dots, N$ . In 3D, the desired heading is a combination of pitch and yaw angles. The desired pitch and yaw angles for each waypoint can be calculated using the transformation from Cartesian coordinates to spherical coordinates as follows:

$$\theta = \sin^{-1} \left[ \frac{-z_d(k) + z(t)}{\sqrt{(x_d(k) - x(t))^2 + (y_d(k) - y(t))^2 + (z_d(k) - z(t))^2}} \right] \quad (4.1)$$

$$\psi = \tan^{-1} \left[ \frac{y_d(k) - y(t)}{x_d(k) - x(t)} \right] \quad (4.2)$$

Here  $[x(t), y(t), z(t)]$  is the current location of the vehicle. In Equation (4.2) four quadrant inverse tangent function (atan2) is used to find the yaw angle in the proper quadrant. The waypoint update is made when the distance between the vehicle and the waypoint  $\rho$  is less than the radius of the sphere of acceptance  $\rho_0$ . In other words, when the current location of the vehicle  $[x(t), y(t), z(t)]$  satisfies:

$$\rho = \sqrt{(x_d(k) - x(t))^2 + (y_d(k) - y(t))^2 + (z_d(k) - z(t))^2} \leq \rho_0 \quad (4.3)$$

the next waypoint  $[x_d(k+1), y_d(k+1), z_d(k+1)]$  is selected. Here the radius of the sphere of acceptance is taken as two times the length of the vehicle, as proposed by [17]. During the course of the vehicle, desired forward speed is set to 1 m/s, however in simulations constant speed reference caused poor convergence and undesired overshoots when the vehicle approaches to the waypoint. To overcome this problem the following speed scheme is applied to the vehicle during maneuvering phases:

$$u(\rho) = \begin{cases} 1 \text{ m/s}, & \rho > 20 \text{ m} \\ 0.05\rho, & \rho \leq 20 \text{ m} \end{cases} \quad (4.4)$$

An example simulation result can be seen in Figure 4.2 from different point of views. Here the guidance algorithm is tested with 5 waypoints where the starting point is at  $[0,0,0]$ , goal point is at  $[20,20,10]$  and the remaining waypoints are distributed randomly in the limits  $[0-20,0-20,0-10]$ .  $\rho_0$  is taken as 3.2 m. The generated references are applied to the control system with a period of 10s. It can be seen that, for each waypoint convergence is achieved, i.e., none of the waypoints are missed.

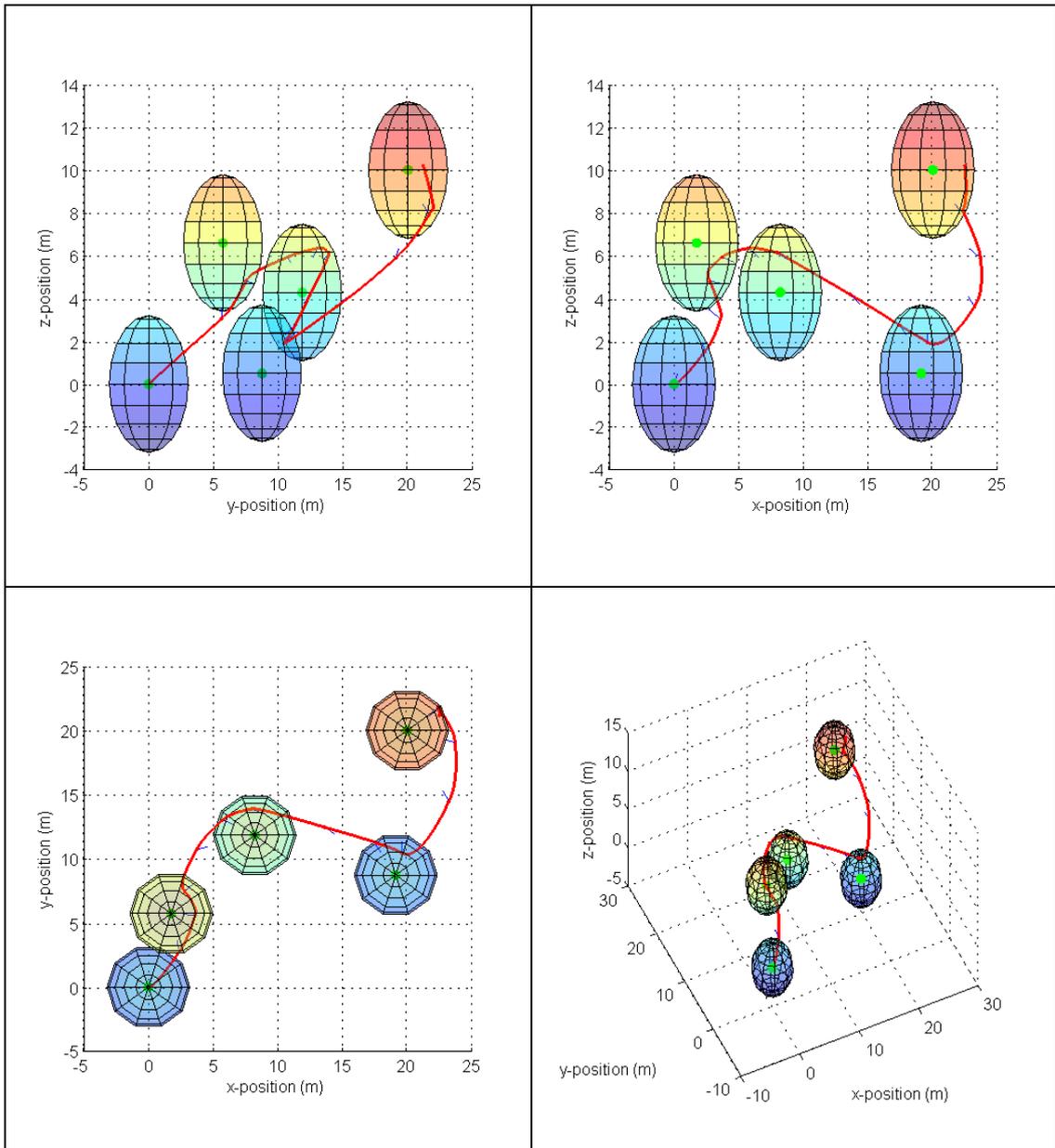


Figure 4.2 Waypoint guidance by LOS simulation in 3D

A major drawback of the waypoint guidance by LOS is that, undesirable control energy consumption be made during the change of trajectory [16]. Once the vehicle is in the sphere of acceptance of the current waypoint, a sudden change occurs in the reference heading. This causes an excessive control effort to be applied on the vehicle during the transients. In this study, the required control effort is reduced by

adding a series first order filter to the reference commands generated by the guidance algorithm. Hence a smoother turn at the waypoints is also achieved.

The choice of the radius of acceptance is important. To demonstrate the effect of the radius of acceptance in the guidance algorithm, the same waypoints are selected in a 2D environment with limits [0-50,0-50] and  $\rho_0$  is taken as first 1 m. and then 3.2 m. When  $\rho_0 = 1m$ . the vehicle circles around the waypoints and reaches the desired region after many iteration, as shown in Figure 4.3.

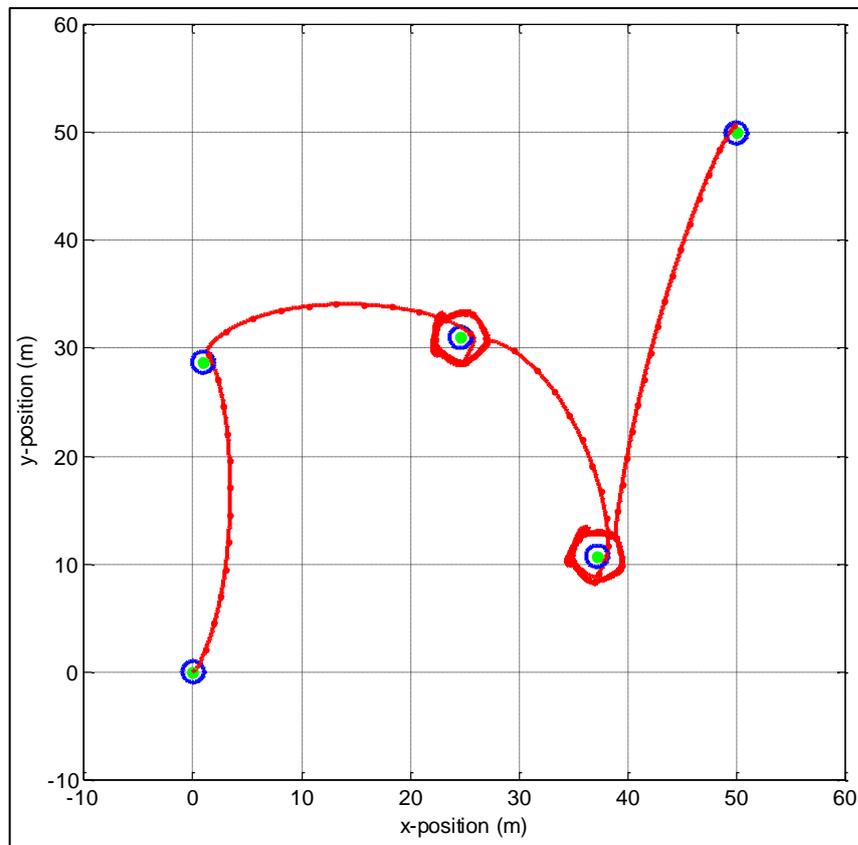


Figure 4.3 Waypoint guidance with  $\rho_0 = 1m$ .

However, when  $\rho_0 = 3.2m$ . the vehicle does not circle around the waypoint and travels on a smooth path as shown in Figure 4.4.

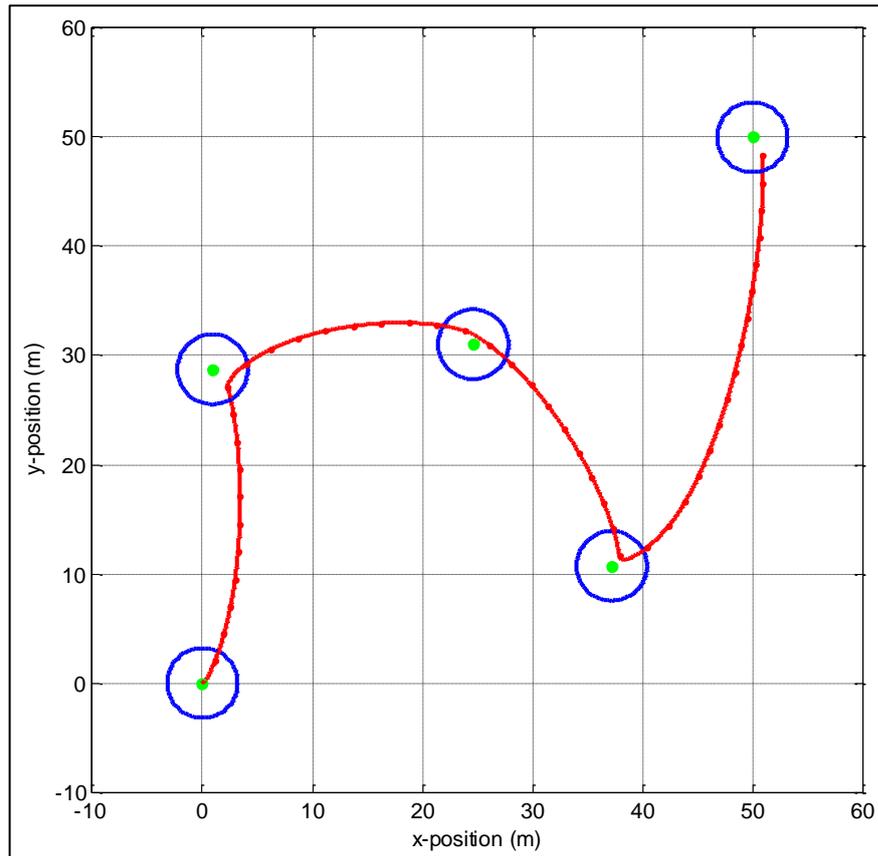


Figure 4.4 Waypoint guidance with  $\rho_0 = 3.2m$ .

In Figure 4.5 the waypoint guidance by LOS is simulated for a standard reconnaissance mission. The underwater vehicle is released into water, dived in for 50 m., surveyed the surface of the water by using given 11 waypoints, and got back to the starting point. The circle of acceptance is taken as 3.2 m., and it is verified that all of the waypoints are visited in a smooth way. The mission is completed in 16 minutes.



## CHAPTER 5

### PATH PLANNING

#### 5.1 Introduction

Path planning is a very important aspect of autonomous operation. The goal of path planning can be defined as; to plan a sequence of paths subject to some optimization criteria that allows the vehicle to accomplish its objectives by reaching the specified goal point from the starting (or current) location without colliding with an obstacle in the workspace [18]. Over the years, many researches have been conducted research on the subject of path planning of underwater vehicles, and different approaches have been developed particular to its main application area. [18] and [19] proposed genetic algorithm based planners, [20] defined a connectivity graph and applied Dijkstra's algorithm, [21] introduced a methodology based on configuration space, [22] utilized the A\* approach, [23] solved the unmanned aerial vehicle (UAV) path planning problem by using evolutionary operators, and [24] introduced D\* star algorithm for path planning in partially known environments.

The main application area of ULISAR vehicle is underwater reconnaissance. Thus the objective of our path planning is to compute a collision-free path to visit all the given desired regions, taking into account the vehicle dynamics, while minimizing the path to reduce energy and time cost. Since the UUV works in a very large scaled 3D underwater environment and the motion of the vehicle is coupled and highly nonlinear, to find a global solution to this optimization problem is difficult. To make the problem easier some approximations can be made on this problem. [18], [19] and [22] have used a grid based environment model, where the depth information is embedded as a parameter of the grid, and found the globally optimized path on this approximation. An alternative and effective method, however, is to solve the

problem in two steps; namely global path planning and local motion planning. This two-step method is similar to the behavior of a human who is intending to reach a destination point from a start point. At the first step (global path planning) he finds a sequence of streets to get there, i.e., he defines an optimal path class that minimizes the cost evaluated from a certain view point. In the second step (local motion planning), he finds a detailed motion by making local decisions among numerous possible motions in accordance with the path found in the first step [20]. In this study, an obstacle avoidance behavior is also included as a part of the second step.

The global path planning is modeled as a modified travelling salesman problem (TSP), where the starting and goal points are fixed and the shortest path visiting all the desired regions is found using the genetic algorithm. In the second step, the local motion planning problem is solved by Rapidly-exploring Random Trees (RRT) method, which is a fast searching probabilistic motion planner taking into account the vehicle constraints implicitly.

## 5.2 Path Planning Problem

The original TSP can be stated as follows: given a number of locations and the costs of traveling between them, what is the cheapest round trip route that visits each location once [25]. In our application, we have constructed the path planning problem for our underwater vehicle by modifying the original TSP problem. Let  $C$  be a continuous underwater environment with the dimensions of  $a \times b \times c$  along the three Cartesian directions, where  $z = 0$  represents the sea surface and the  $z$ -axis is directed toward the sea bottom.  $s$  is the starting location,  $g$  is goal point, and  $n_i = (x_i, y_i, z_i)$  represents the desired locations to be visited by the vehicle, where  $0 \leq x_i < a$ ,  $0 \leq y_i < b$ ,  $0 \leq z_i < c$ . Then we can define a route  $R_i$  as a sequence of starting location, goal point, and locations to be visited:  $R_i = \{s, \dots, n_i, n_{i+1}, \dots, g\}$ . We have also included underwater obstacles as unfeasible regions in the search space. With this setting the path-planning problem can be stated as follows: given a starting location  $s$ , a goal point  $g$ , a number of locations to be visited, a number of

obstacles not to collide and the costs (here distance) of traveling what is the cheapest route from the starting point to the goal point that visits each desired location once.

If there are  $N$  locations to be visited, then the number of possible routes of this modified problem is  $N!$ . This is a complex and computationally expensive problem. To give an example, given a starting point, a goal point and 10 locations to visit, 3,628,800 possible routes exist. Many algorithms have been developed to solve the TSP by minimizing computation effort, solution time or used memory. In this study, genetic algorithm is adopted to solve the TSP due to its fast convergence and minimal memory usage.

### 5.3 Genetic Algorithm

Genetic algorithm is an evolutionary method based on the Darwinian theories of natural selection and survival. In this method, a population of possible paths is maintained and the paths are iteratively transformed by genetic operators such as crossover and mutation [26]. The genetic algorithm constructed to solve the path planning is composed by the following routines:

#### a. Initialization

An initial population consisting of  $N$  individuals is generated randomly. Each individual is a candidate path represented by the sequence  $R_i$ , having a fixed start and goal position and randomly permuted locations to be visited.

#### b. Fitness Function

Given a sequence of locations, the Euclidian distance between each consecutive locations are summed up as follows:

$$J_i = \sum_{i=1}^{N-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \quad (5.1)$$

If there is an obstacle exists in the path connecting two consecutive locations, then the distance is increased to satisfy a collision free path between these locations around the obstacle. The shortest paths are evaluated as the strongest individuals.

c. *Selection*

The selection function chooses the parents of the next generation based on their fitness values. Here the roulette wheel method is adopted, where each individual occupies an area on the roulette wheel proportional to its fitness value. At every step, the algorithm uses a random number to select one of the sections with a probability equal to its area.

d. *Mutation Function*

Mutation function is used to ensure that the solutions found do not converge prematurely to a local minimum. Given a sequence of locations, a mutated individual is generated by swapping two randomly selected elements of the permutation. Since the start and goal locations are fixed, those elements are excluded from the selection of elements to be swapped, and then the rest of the locations are selected by a uniformly distributed random number generator.

*Original individual:*

<b>S</b>	4	9	3	6	1	10	5	7	2	8	<b>G</b>
----------	---	---	---	---	---	----	---	---	---	---	----------

*Mutated individual:*

<b>S</b>	4	5	3	6	1	10	9	7	2	8	<b>G</b>
----------	---	---	---	---	---	----	---	---	---	---	----------

e. *Crossover Function*

Crossover function is used to produce new individuals (offspring) from the given individuals (parents). Given two parents and a crossover point, parents are combined together from the crossover point to generate children. However if the parents are combined directly, there is a high probability of repetition of the locations in the child sequence. To overcome this problem Partially-Mapped Crossover (PMX) algorithm [27] is adopted. PMX algorithm randomly selects a crossover point. The value of the first position of  $R_2$  is copied to the first position of  $R_1$ , while the original value of the first position of  $R_1$  is transferred to the position in  $R_1$  of which the value

is the same as the new value of first position of  $R_1$ . The same is applied for second and maybe the third until the crossover is reached. In the following example the crossover point is selected as the 3<sup>rd</sup> position after the starting location:

*Parent 1:*

<b>S</b>	4	5	3	6	1	10	9	7	2	8	<b>G</b>
----------	---	---	---	---	---	----	---	---	---	---	----------

*Parent 2:*

<b>S</b>	8	10	6	7	4	9	1	3	2	5	<b>G</b>
----------	---	----	---	---	---	---	---	---	---	---	----------

*Step A.1:*

<b>S</b>	4	5	3	6	1	10	9	7	2	8	<b>G</b>
<b>S</b>	8	10	6	7	4	9	1	3	2	5	<b>G</b>

*Step A.2:*

<b>S</b>	8	5	3	6	1	10	9	7	2	4	<b>G</b>
<b>S</b>	8	10	6	7	4	9	1	3	2	5	<b>G</b>

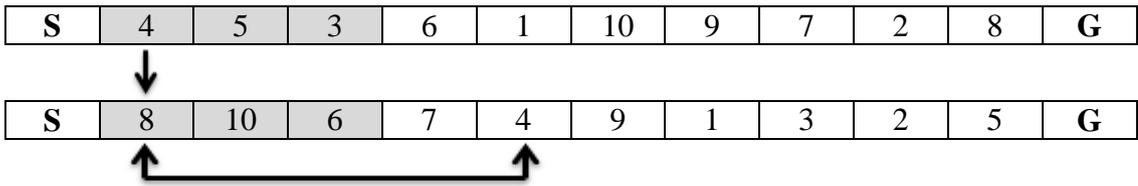
*Step A.3:*

<b>S</b>	8	10	3	6	1	5	9	7	2	4	<b>G</b>
<b>S</b>	8	10	6	7	4	9	1	3	2	5	<b>G</b>

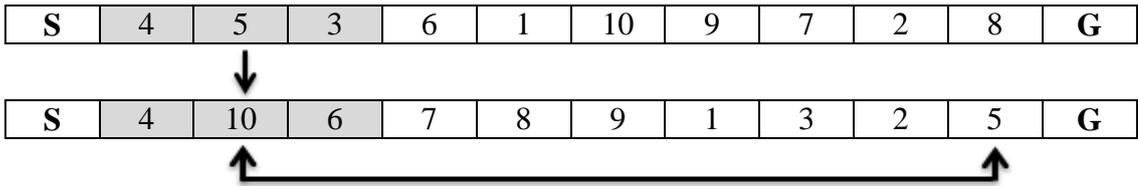
*First child:*

<b>S</b>	8	10	6	3	1	5	9	7	2	4	<b>G</b>
----------	---	----	---	---	---	---	---	---	---	---	----------

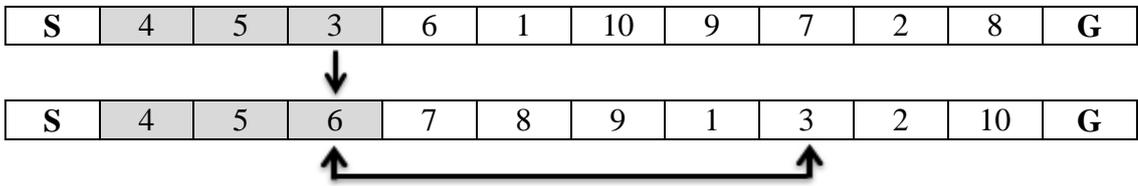
Step B.1:



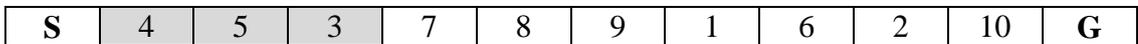
Step B.2:



Step B.3:



Second child:



f. *Reproduction*

Reproduction determines how the children will be created at each new generation. In our implementation, 2 individuals with the highest fitness value (elite individuals) have survived for the next generation. 80 percent of the next generation is produced by crossover function while the remaining 20 percent is produced by mutation function.

For the evaluation of the algorithm, the limits of the environment is selected as  $100 \times 100 \times 100$ . The positions of the starting point, the goal point and 10 locations to be visited are shown in Table 5.1.

Table 5.1 Positions of the locations used in GA simulation

	$x$	$y$	$z$
$s$	0	0	0
$n_1$	74,3686	78,9623	51,9977
$n_2$	44,9767	58,4086	14,4257
$n_3$	9,7939	74,2656	72,8072
$n_4$	55,9766	57,7846	96,2244
$n_5$	5,4709	92,1801	57,2509
$n_6$	1,5682	16,5923	77,2192
$n_7$	42,4243	28,7236	11,3031
$n_8$	55,2227	69,73471	53,1414
$n_9$	26,9202	70,2320	47,2318
$n_{10}$	98,4395	53,9436	68,3772
$g$	100	100	100

Genetic algorithm routines are constructed as described, the maximum number of iterations are set to 50, and the results for three different population sizes, 50, 100 and 200 are simulated and illustrated in Figure 5.1, Figure 5.2, and Figure 5.3 respectively.

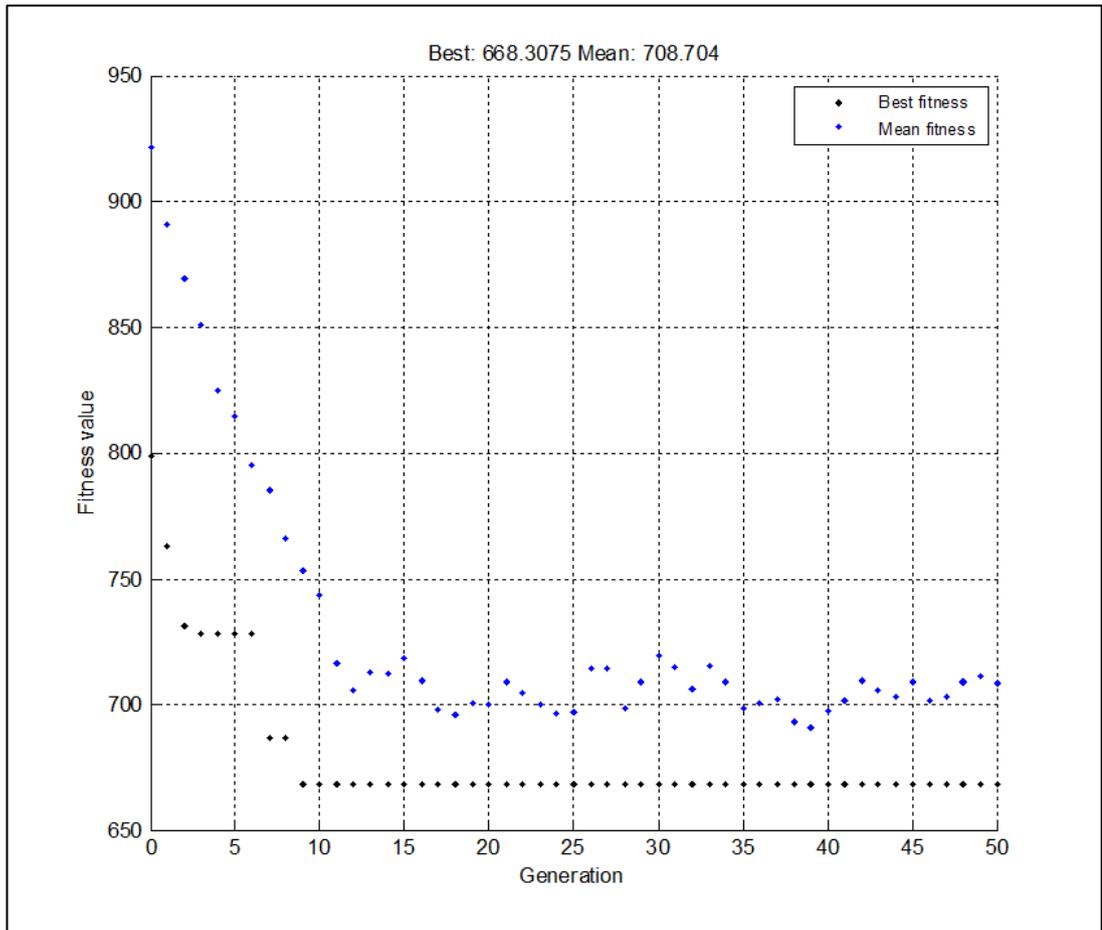


Figure 5.1 GA based TSP simulation for the population size of 50

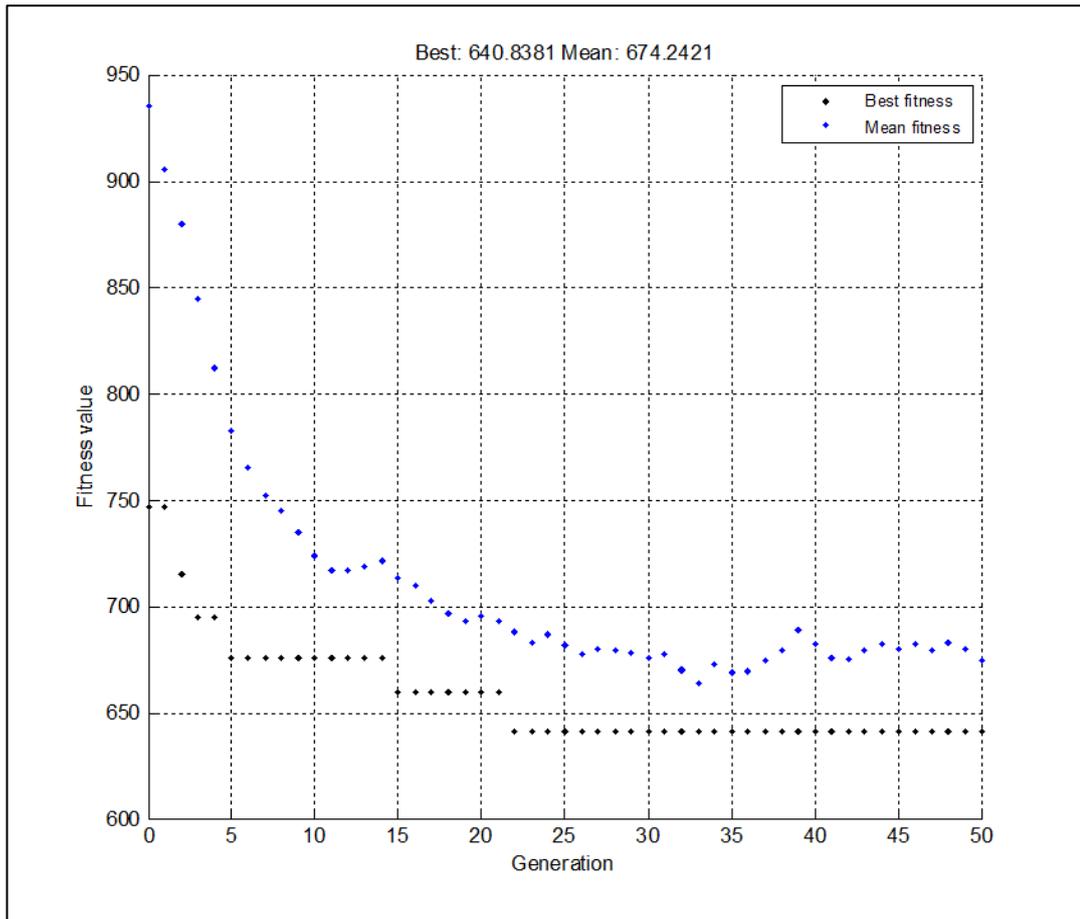


Figure 5.2 GA based TSP simulation for the population size of 100

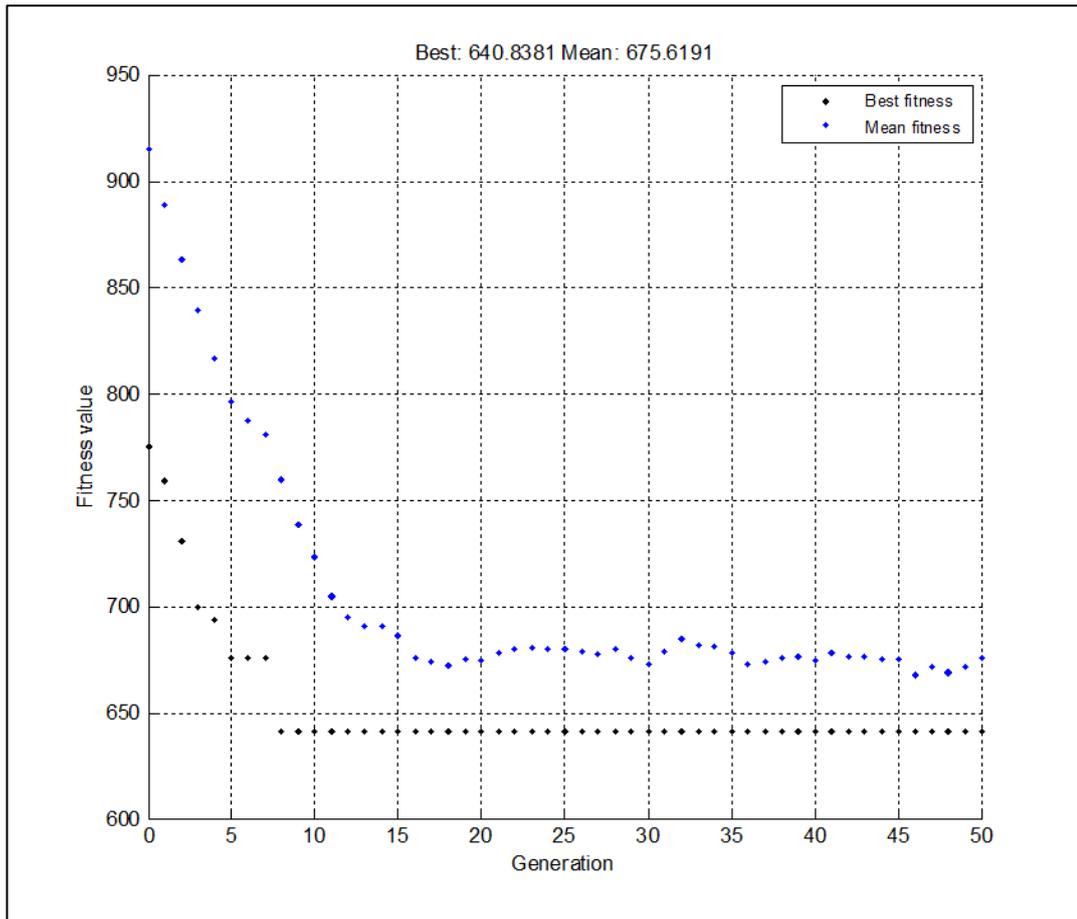


Figure 5.3 GA based TSP simulation for the population size of 200

The global minimum for this setting is 640.8381, however results show that relatively smaller population size of 50 is not able to converge to the global minimum after 50 generations, stalling in an early local minimum instead. When the simulation is repeated with population sizes of 100 and 200, the global minimum is reached where larger population converges faster as expected. To compare the time spent to produce 50 generations, it took 0.067147 seconds for the population size of 50, 0.119203 seconds for 100 and 0.147584 seconds for the population size of 200. 3D graph of this configuration and the shortest path found connecting the starting point to goal point visiting the desired locations are illustrated in Figure 5.4.

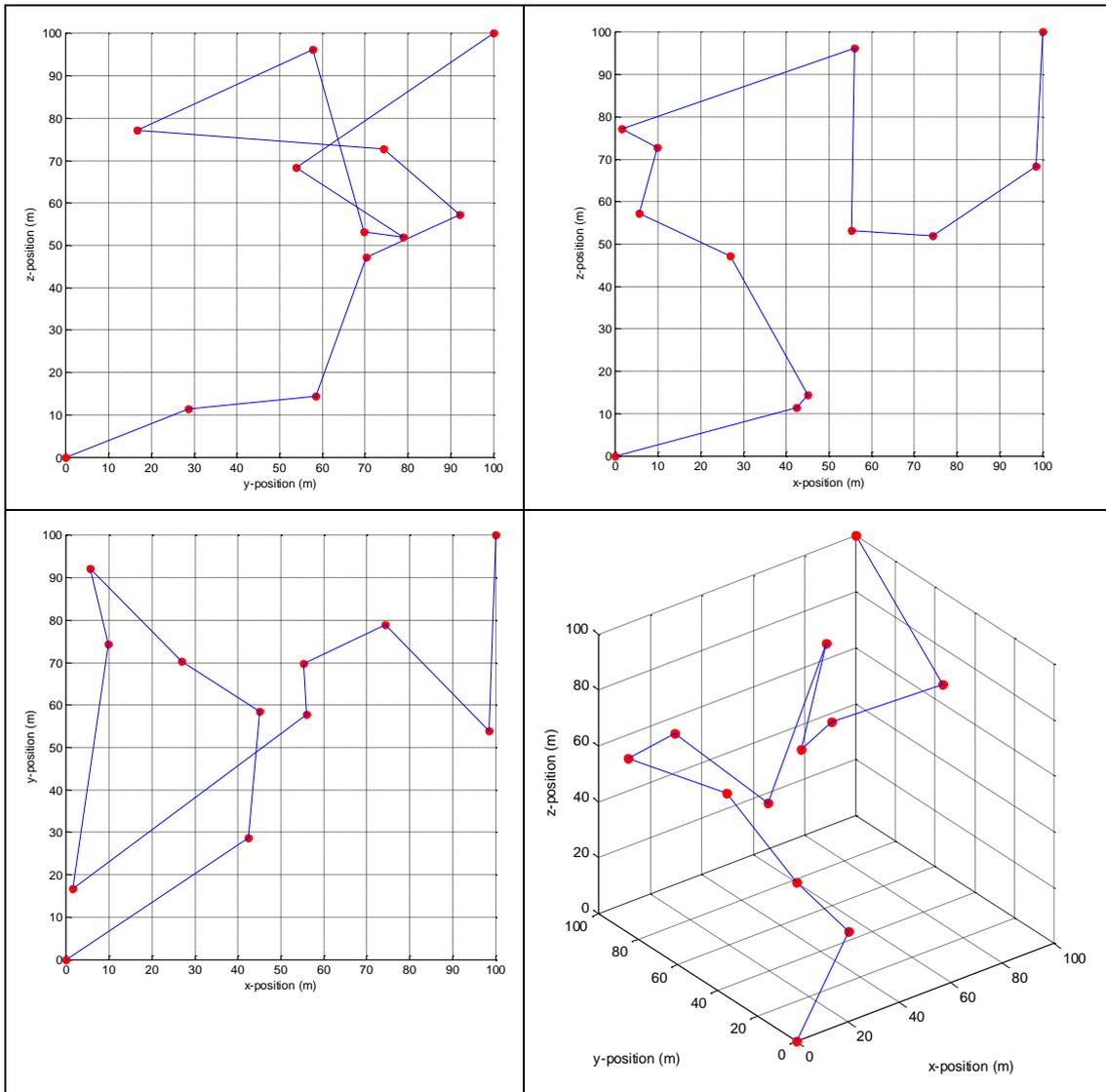


Figure 5.4 3D path generated by the GA based TSP algorithm

## 5.4 Motion Planning Problem

Motion planning gives the solution to a vehicle's motion problem which is under influences of kinematic and dynamic constraints. In general, the problem is, given a vehicle system; we must find a trajectory that starts from a start configuration and ends at a goal configuration, while avoiding obstacles and respecting the kinematic and dynamics constraints. The kinematic constraints are mainly the obstacles (or the

joint limits in manipulator applications) which limits the configuration of the vehicle. The dynamics constraints, however, govern the time-derivatives of the configuration [28].

The motion planning problem in our application can be stated as; given a starting configuration (position and velocity) and a goal position, finding a sequence of feasible control inputs required to move the vehicle among obstacles until the goal is reached. In this problem, the obstacles and the bounds on the orientation of the vehicle refer to the kinematic constraints, whereas the bounds on the acceleration and velocity refer to the dynamic constraints.

Over the years many algorithms have been developed to solve the motion planning problem. A common approach is to solve the problem in two steps; first to construct a roadmap by solving the basic path planning problem considering only the kinematic constraints and then finding a trajectory and control inputs that satisfies the dynamics and follows the path found in the first step [29]. Some of the roadmap path planning methods utilizes Dijkstra's algorithm, A\* search or other search algorithms on the maps created by visibility graphs, Voronoi diagrams, cell decomposition etc., or uses the potential functions to translate the problem into an optimization problem and finds an analytical solution. However the complexity of the roadmap planning methods in high-dimensional configuration spaces is high, therefore not practical for real-time applications. In addition, the potential field methods suffer from the presence of local minima [30]. Those disadvantages have led researchers to make a trade-off between the planning time (computational complexity) against optimality in terms of: (a) execution time of the motion, (b) strictness in observing the safety margin, and (c) closeness to the desired start and goal positions and velocities [28], and the Probabilistic Roadmap Planners (PRM) have been introduced. PRM method builds probabilistic roadmaps in the free configuration space (C-space) [31] of the vehicle by generating and interconnecting a large number of (mostly) random configurations of the robot [30]. Although PRM methods produce fast solutions to the roadmap finding problem in high DOF configuration space, it considers only the kinematic constraints like other static

planners. When the inertia and the limited controllability of the real world robots are considered, static planners cannot always find inputs for an arbitrary path in a dynamic environment and the result of the planner can be unexecutable by the robot [32].

The motion planning technique that takes into account dynamic constraints as well as kinematic constraints is known as kinodynamic planning [28]. Apart from the kinematic planners which are only concerned with motion, without regarding to the forces that cause it, kinodynamic planners incorporate dynamic constraints and plans velocity as well as position. The main distinctions of the kinodynamic planners are about the representation of the state of the vehicle and the obstacles. The dimension of the state representation of the vehicle is doubled, incorporating the dynamic quantities (velocity) as well as kinematic quantities (position). This allows the expression of dynamic constraints of the vehicle as well as introduces the concept of imminent (inevitable) collision region [32]. In this region, the state of the vehicle is such that; vehicle doesn't collide with any obstacle, but, because of its momentum, it cannot do anything to avoid collision in future. Intuitively, as the speed of the vehicle increases, imminent collision regions grow. Therefore a collision-free state space is obtained by extracting the inevitable collision regions and the static obstacle regions from the state space. Figure 5.5 shows the obstacles in state space seen by a point mass robot obeying moving in x-direction with different initial speed (Black regions denotes  $X_{obst}$ , gray denotes  $X_{ric}$ , and white regions denote  $X_{free}$ ).

Comparing the purely kinematic planners and kinodynamic planners, we have found the kinodynamic planner's suit better to our needs, particularly to produce a motion planning solution based on our highly non-linear and coupled vehicle model in a dynamic underwater environment. Consequently, we have adopted the Rapidly-Exploring Random Trees (RRT) algorithm [32] which is a randomized approach, giving a fast solution to kinodynamic planning problem in high-dimensional state spaces.

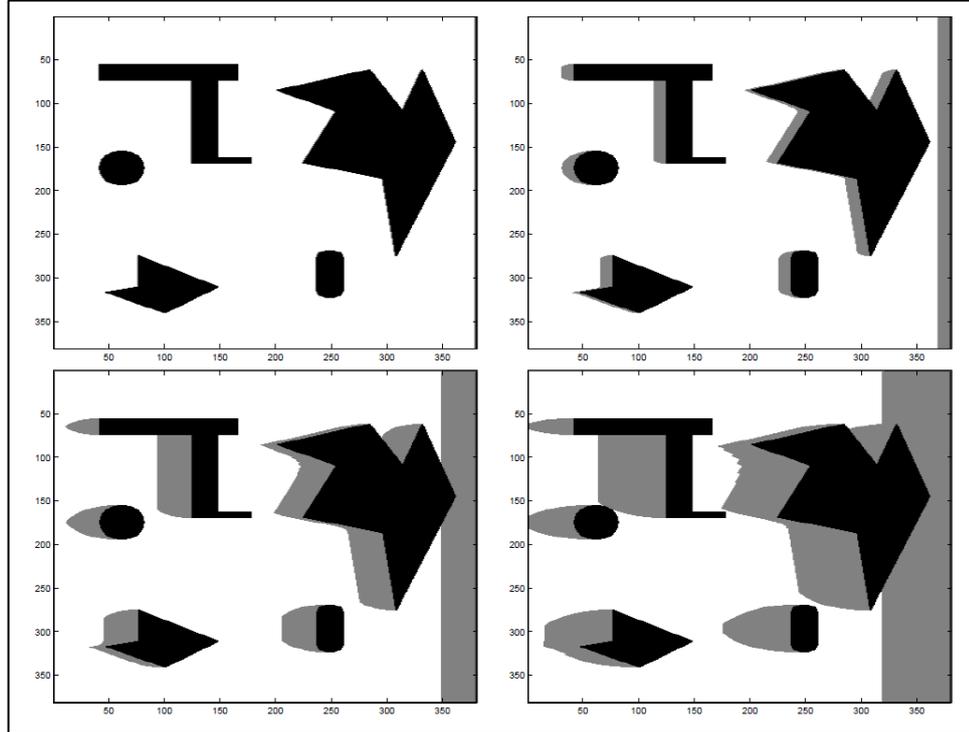


Figure 5.5 Obstacles and imminent collision regions in state space [32]

## 5.5 Rapidly-Exploring Random Trees (RRT) Algorithm

Before giving the details of the algorithm, we can briefly introduce our implementation and some modifications on the problem formulation. First of all, rather than formulating the differential constraints directly by the state space representation of our mathematical model as in Equation (2.2), we have implemented the low level LQR controller together with the plant, so that the motion planner performs a higher level control on the plant dynamics by producing reference inputs to the controller. This can be formulated as;

$$x(t + \Delta t) = g(x, r) \quad (5.2)$$

where,  $r \in R$  represents a set of allowable reference inputs for the LQR controller produced by the motion planner. Therefore Equation (5.2) yields a convenient parameterization of the allowable state transitions via the references in  $R$ .

The obstacles in the state space are implemented as described in [32]. Thus the collision-free paths are found in  $X$  rather than  $C$ , where  $X$  refers to the state space and  $C$  refers to the configuration space [31]. If we represent the states fall in the regions with obstacles as  $X_{obst}$ , the states fall in the region of inevitable collision as  $X_{ric}$ , and since  $X_{obst} \subseteq X_{ric}$ , we define  $X_{free} = X \setminus X_{ric}$ , where  $X_{free}$  represents the collision free regions.

The last modification is made on the formulation of the solution trajectory. Because of the modification made on the formulation of the dynamics of the state space described above, we can also redefine the problem of motion planning as to find a time parameterized reference function  $r: [0, T] \rightarrow R$  which results in a collision-free trajectory that starts at  $x_{init}$ , and ends at  $X_{goal}$ , where  $x_{init} \in X$  represents the initial state and  $X_{goal}$  represents the goal region.

Based on these formulations, an RRT can be constructed as follows [32]:

---

*BUILD\_RRT*( $x_{init}$ )

```

1   T.init( $x_{init}$ )
2   for  $k = 1$  to  $K$  do
3      $x_{rand} \leftarrow \text{RANDOM\_STATE}()$ ;
4     EXTEND( $T, x_{rand}$ )
5   Return  $T$ 

```

---

*EXTEND*( $T, x$ )

```

1    $x_{near} \leftarrow \text{NEAREST\_NEIGHBOR}(x, T)$ 
2   if NEW_STATE( $x, x_{near}, x_{new}, r_{new}$ ) then
3     T.add_vertex( $x_{new}$ );
4     T.add_edge( $x_{near}, x_{new}, r_{new}$ )
5     if  $x_{new} = x$  then
6       Return Reached;
7     else
8       Return Advanced;
9   Return Trapped;

```

---

The RRT is initialized by defining a starting configuration and extended until it gets sufficiently close to the goal configuration. The extension algorithm is illustrated in Figure 5.6. First a random point  $a$  in  $X$  is picked and the vertex  $b$  which is the closest vertex to the point  $a$  in the tree is found. Then the reference  $r$  is calculated which intends to lead the vehicle from  $b$  to  $a$ . When this reference is applied on the configuration  $b$  for a period of time  $\Delta t$ , let's say that the robot reaches the configuration  $c$ . If no collision occurs in moving from  $b$  to  $c$ , the configuration  $c$  is added in the tree and the reference  $r$ , generating this motion, is stored.

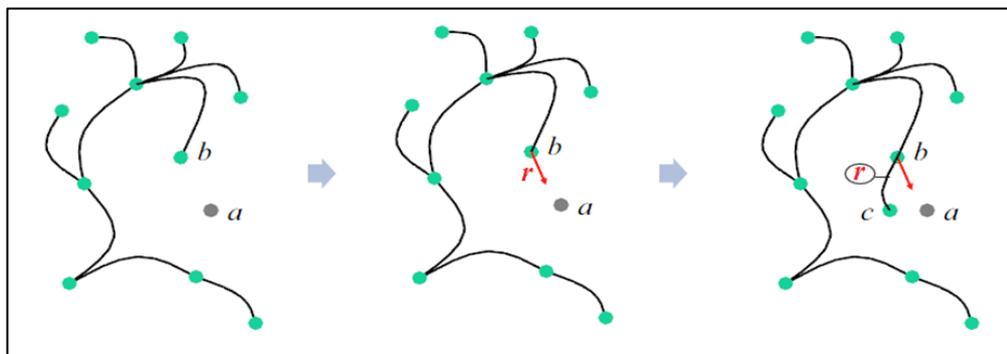


Figure 5.6 Extension of a RRT

Once the RRT reaches the goal configuration, the tree is backtracked identifying the stored references that lead from the starting configuration to the goal configuration and the algorithm outputs those references in the order from the start to the goal.

The standard implementation of the algorithm satisfies a fast and uniform exploration of the search space. However the simulation results showed that it falls short to converge to the goal configuration in a reasonable time and number of iterations.

In the simulations the search space is limited to an Euclidean space of which the dimensions is  $100m \times 100m \times 100m$  and constrained by 10 spherical obstacles with radius 10m. The starting position is  $x_{init} = [0 \ 0 \ 0]$  and the goal position is  $x_{goal} = [100 \ 100 \ 100]$ . The space is sampled by the random positions in Cartesian coordinates and no constraints are imposed on the velocities. The Euclidean distance

is used as the distance metric. The references leading the vehicle to the sampled positions is generated by the LOS guidance method and applied through the LQR controller. After 1000 iterations the tree couldn't reach the goal as seen in Figure 5.7.

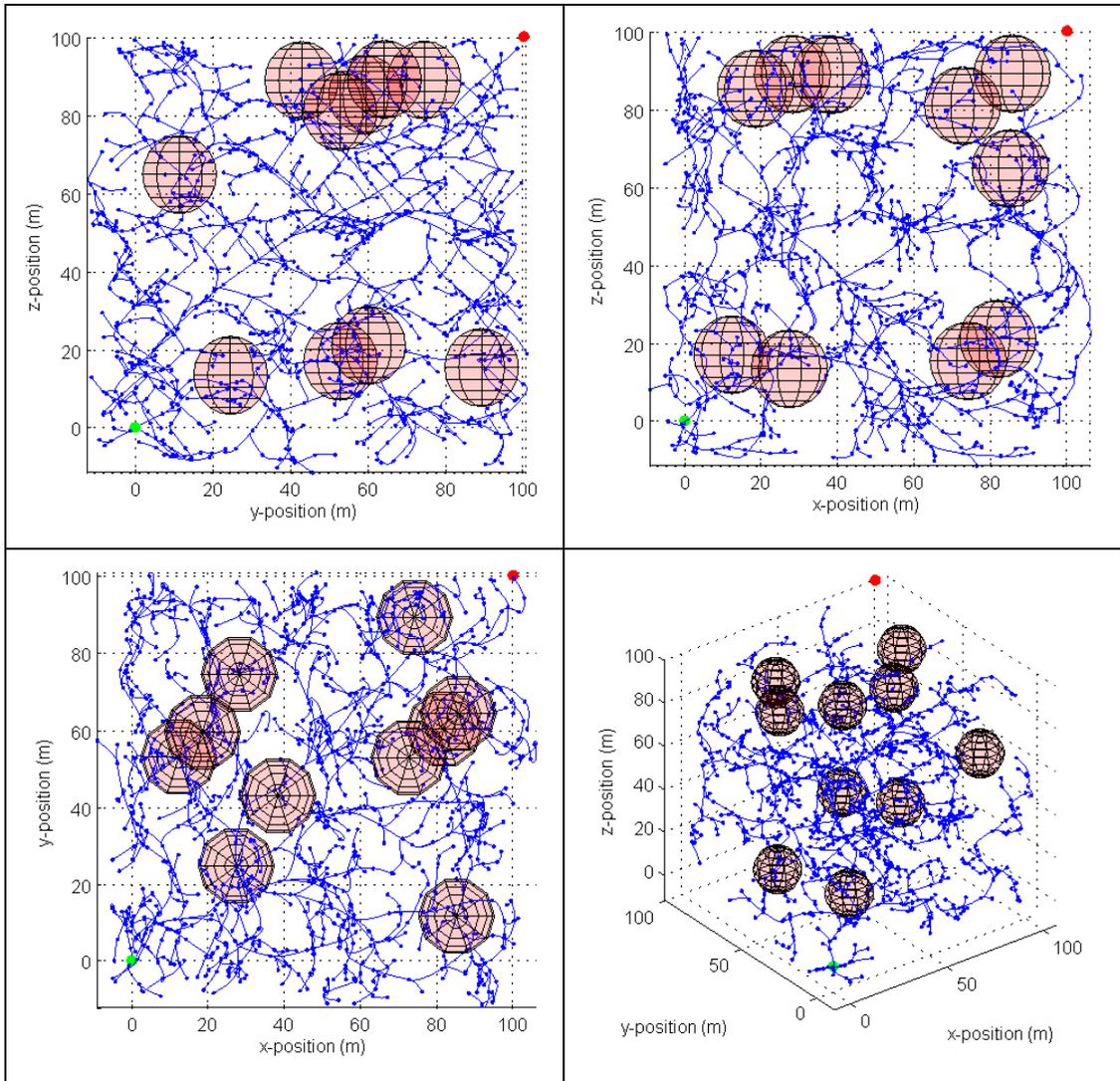


Figure 5.7 Standard RRT algorithm simulation result

To solve the fast convergence to the goal problem, while randomly exploring the search space to avoid obstacles, an informed search strategy is introduced, called the goal biasing. In this method rather than sampling always a random point in the search

space, the algorithm is enforced to sample the goal state with some probability. The simulation result showed that keeping the search space same as above, when the goal is biased with a probability of  $p = 0.5$ , the algorithm converges to the goal in 41 iterations without excessive number of unnecessary branching out. The result is illustrated in Figure 5.8.

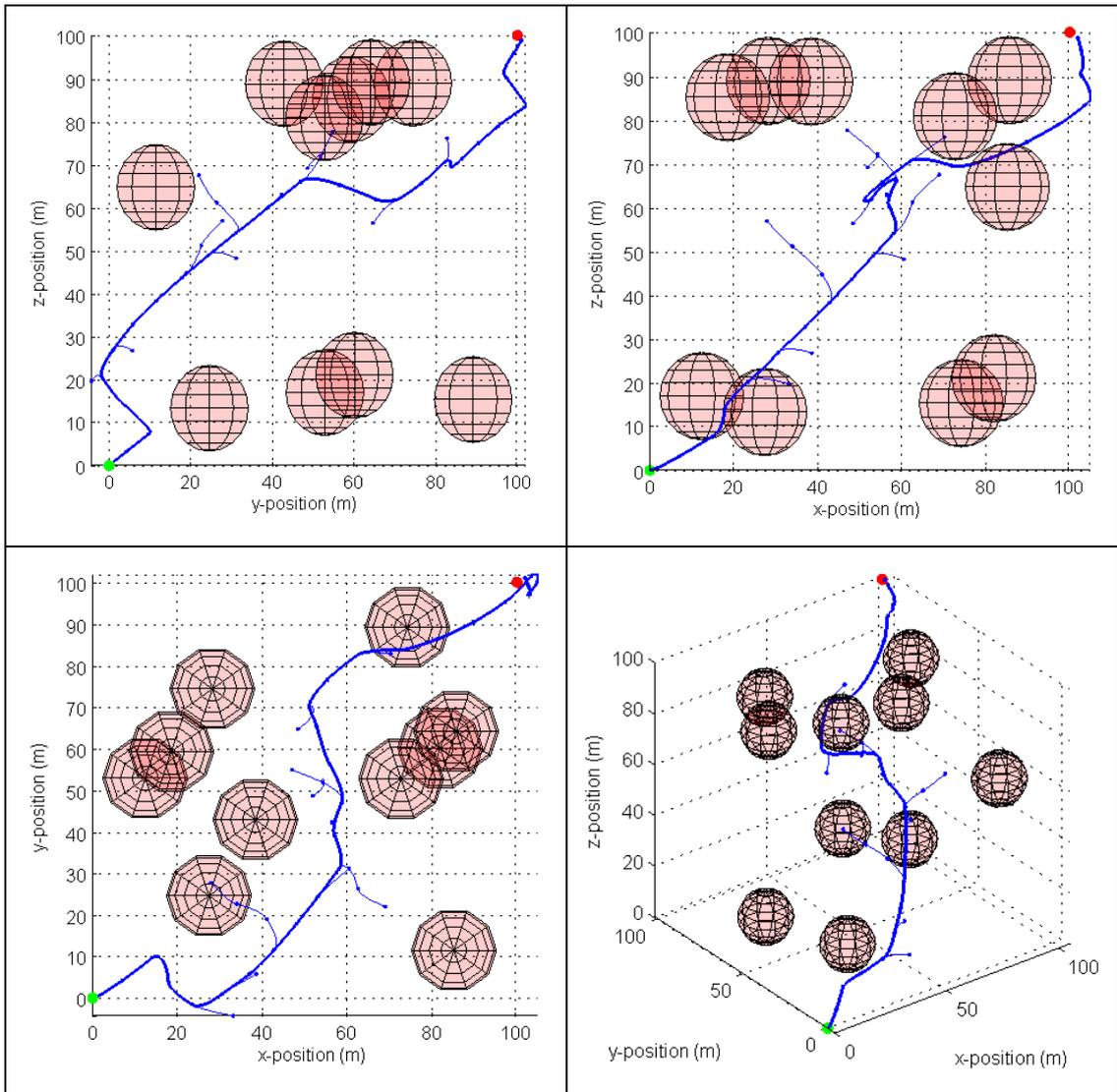


Figure 5.8 Goal-biased RRT algorithm simulation result

Intuitively in a complex environment with many obstacles the biasing should be decreased to allow the algorithm to search for alternative paths by exploring the search space. In contrast to that if there is a direct way from start to the goal or there are not many obstacles in the search space, biasing can be increased to avoid unnecessary branching out. Figure 5.9 and Figure 5.10 shows an example to that, where two different biasing factors are used for two different settings of the search space.

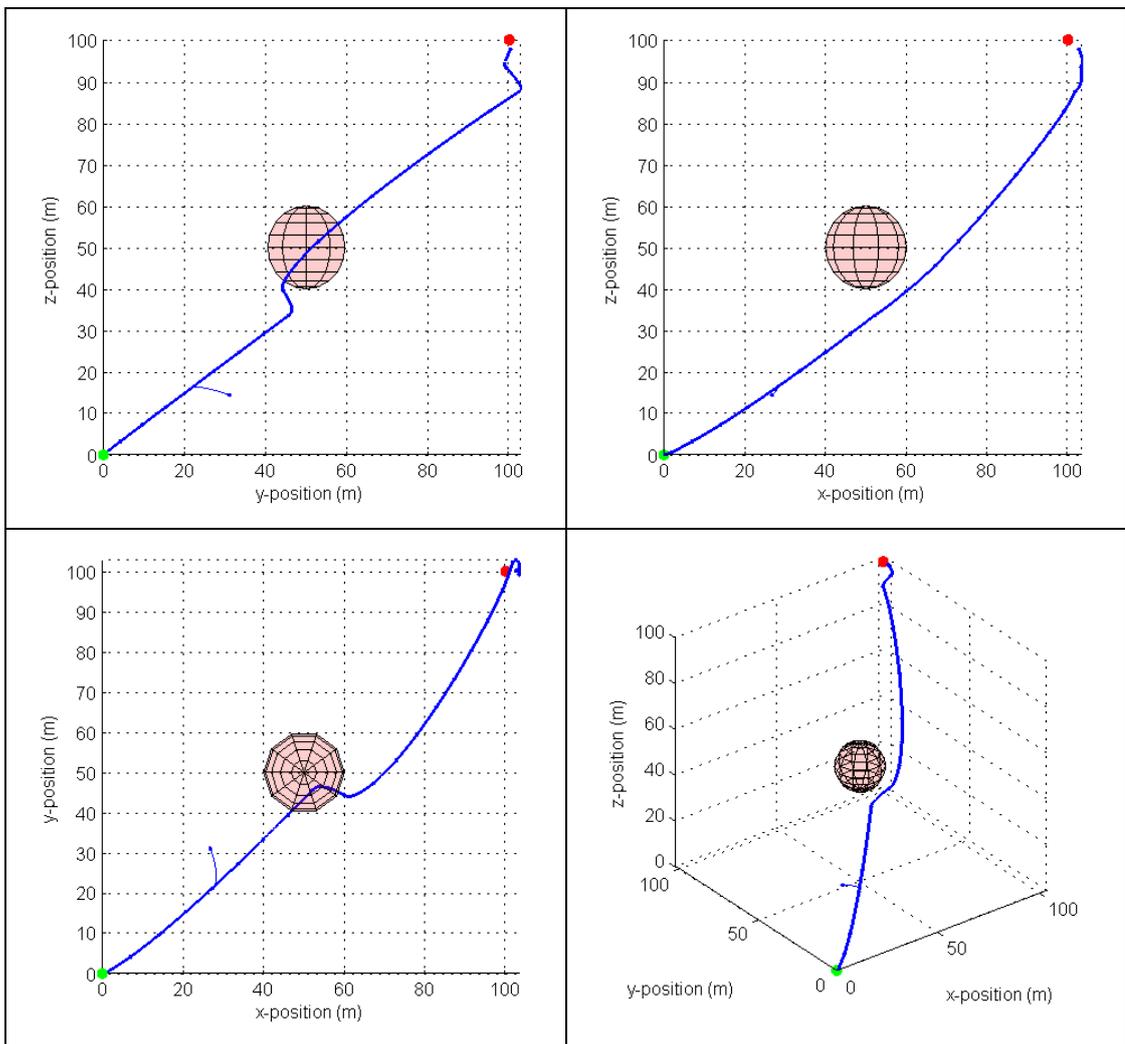


Figure 5.9 RRT with high goal-biasing in a sparse environment simulation result

Rather than using a fixed goal biasing value, the algorithm can be improved further by introducing an intelligent goal biasing scheme, utilizing the distance to the goal information. The algorithm is initialized by a constant bias and as the tree approached to the goal, the biasing is increased. Equation (5.3) shows the bias function used and Figure 5.11 illustrates the simulation result using this biasing scheme which converged to the goal in 28 iterations:

$$p(d) = \begin{cases} 1 - \frac{d}{100}, & d < 50 \\ 0.5, & d \geq 50 \end{cases} \quad (5.3)$$

Here  $d$  is the distance between the goal and the closest edge to the goal in the tree.

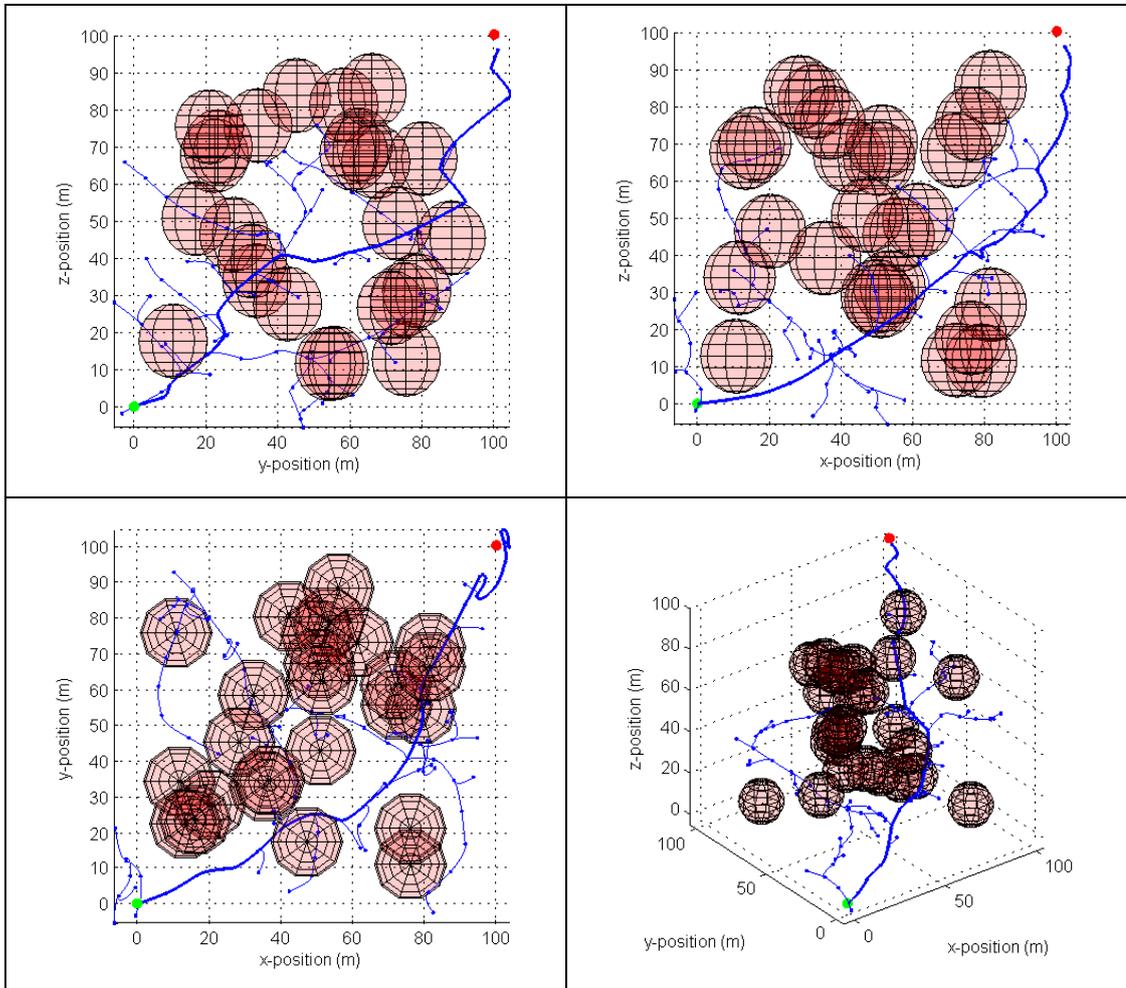


Figure 5.10 RRT with low goal-biasing in a dense environment simulation result

The last step in the implementation of the path planning is the integration of the RRT motion planner with the TSP path planner and the method is intuitive: Given a starting point, a goal point, a number of locations to visit and a number of obstacles to avoid; first the TSP is solved to find in which sequence will the locations be visited. Then using the RRT algorithm, collision free trajectories are planned between each consecutive location to visit taking into account the dynamics of the vehicle. The integrated path planner is verified by numerous simulation results, with different settings of the obstacles and search space.

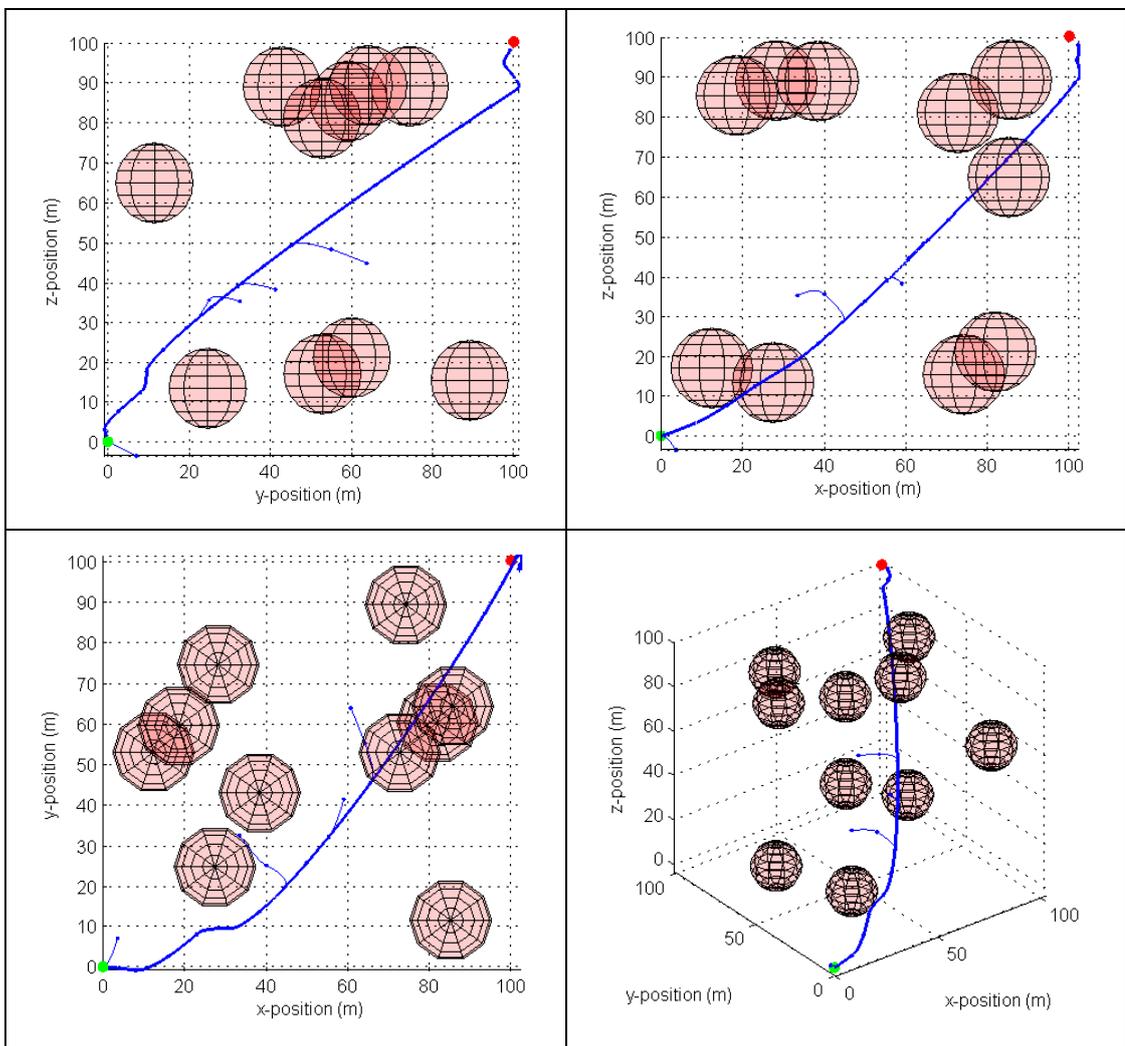


Figure 5.11 RRT with varying goal-biasing simulation result

Figure 5.12 shows an example of the path planning simulation results, where the green regions represent desired regions to visit and the red regions represents the obstacles to avoid. Figure 5.13 and Figure 5.14 shows a more dense search space where the obstacles and regions to be visited are placed on the horizontal plane.

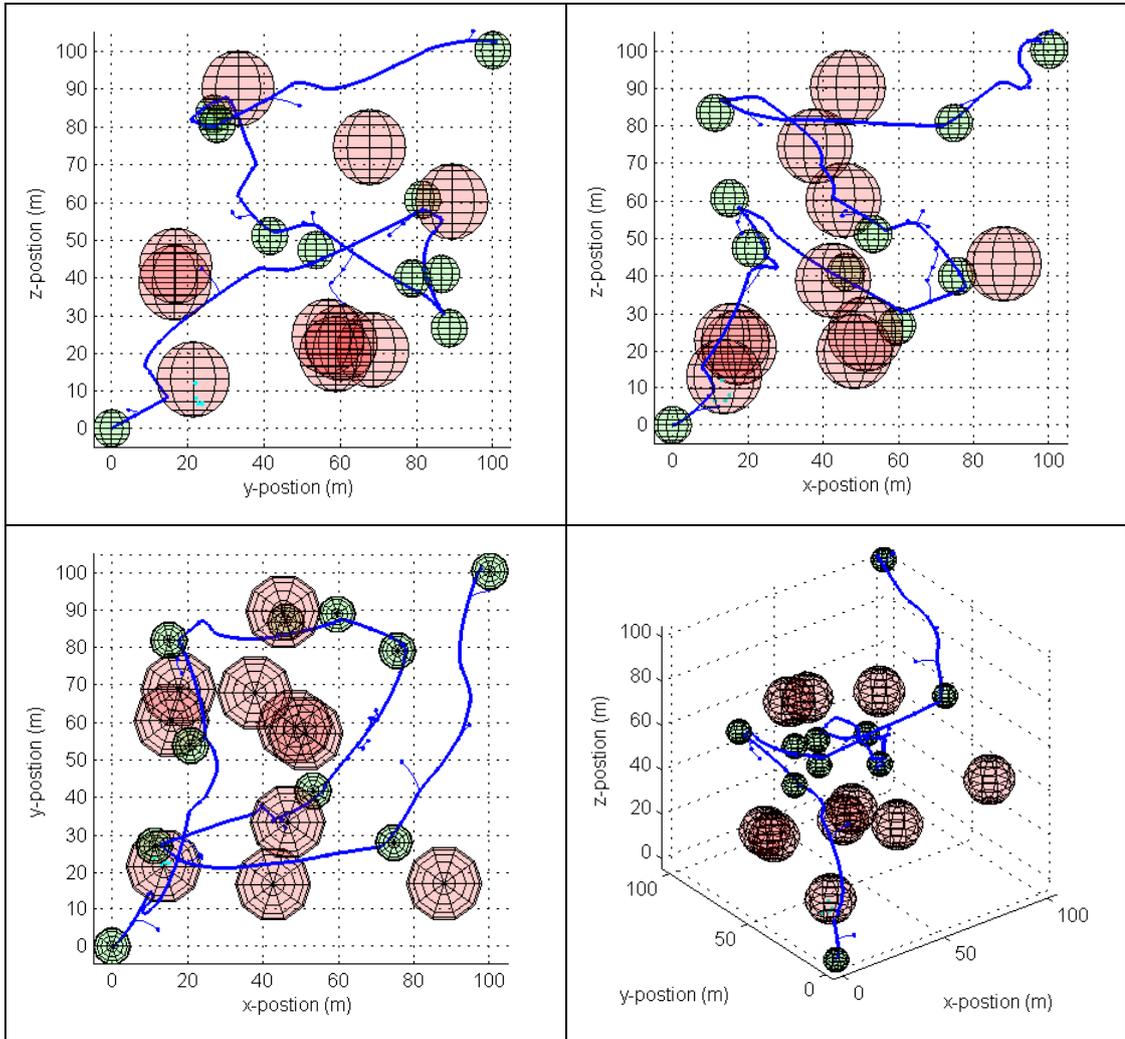


Figure 5.12 TSP/RRT integration simulation result 1

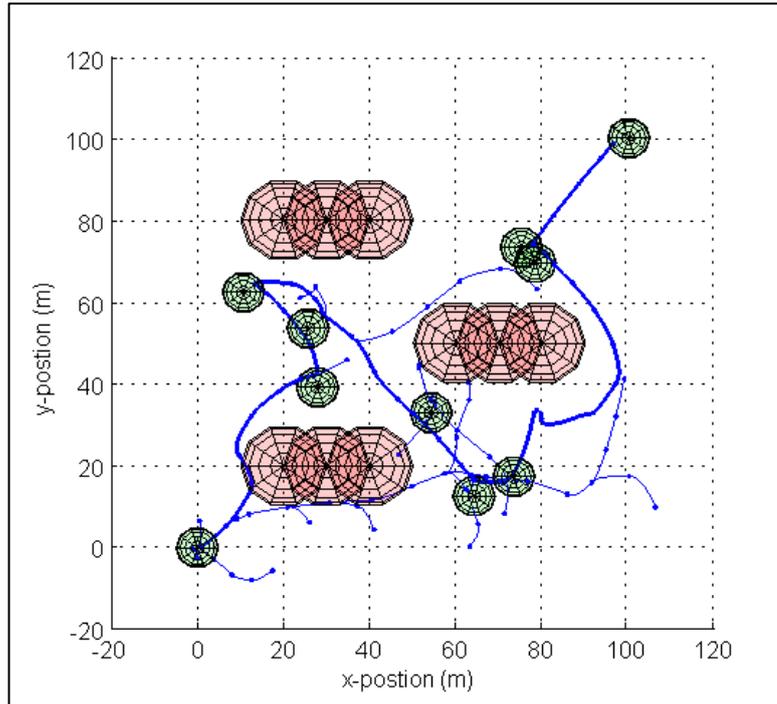


Figure 5.13 TSP/RRT integration simulation result 2

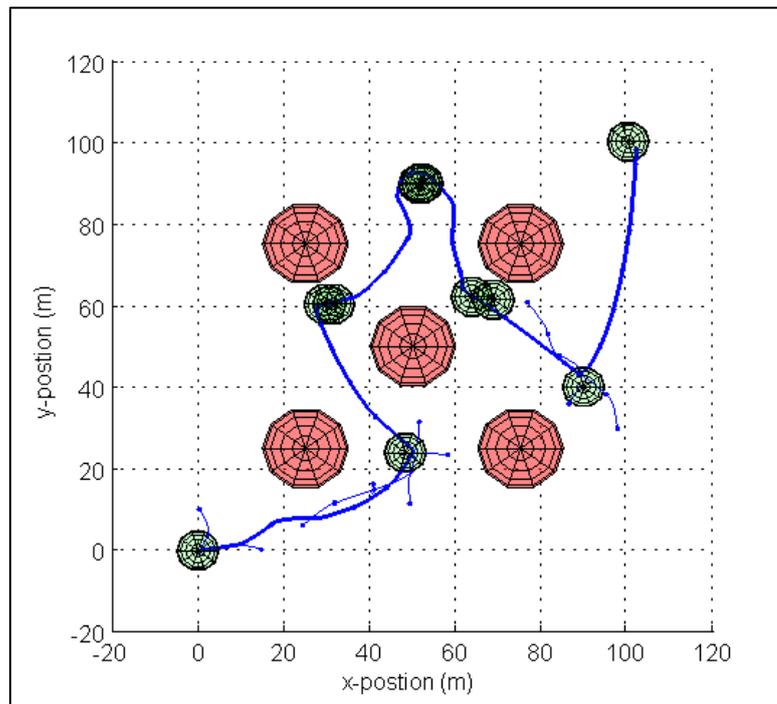


Figure 5.14 TSP/RRT integration simulation result 3

# CHAPTER 6

## NAVIGATION

### 6.1 Introduction

Navigation can be defined as the determination of the position and velocity of a moving body with respect to a known reference. In addition to that, some navigation systems provide attitude, acceleration, and angular rate information as well. A navigation system may be placed on the navigating vehicle and operate stand-alone or may require external infrastructures and components. The output of the navigation system is known as the navigation solution, which represents the coordinate frame of the navigating vehicle with respect to a reference coordinate frame resolved about the axes of a third coordinate frame. Most of the navigation techniques are based on two fundamental concepts known as position fixing and dead reckoning [5].

Position fixes may be obtained by measuring the range and/or bearing of the vehicle with respect to known reference obstacles. Bearing is the angle between the line of sight to an object and north. When the bearing angles are measured from two different known references, position of the navigating object can be found by intersecting the lines from the reference objects in the direction of each bearing measurement as seen in Figure 6.1. A third bearing measurement is needed for 3D positioning of the object.

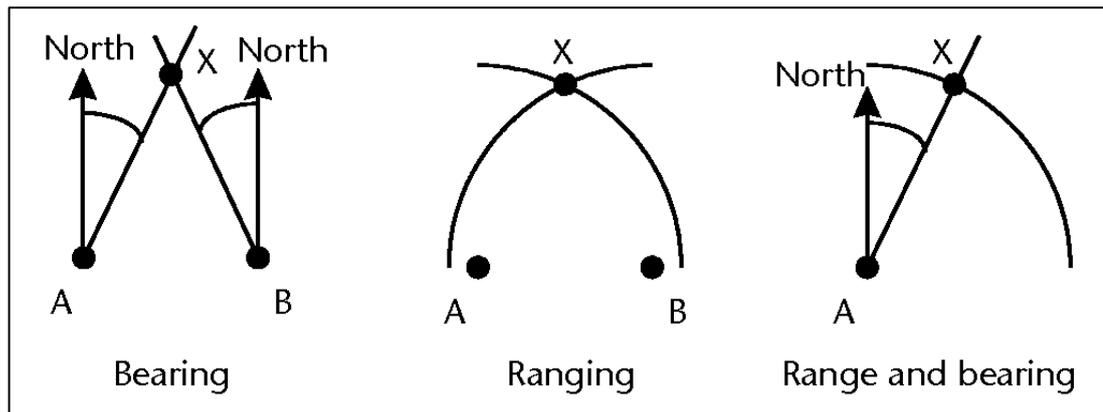


Figure 6.1 Ranging and bearing [5]

Ranging is done in a similar way to bearing; when two range measurements are taken from the reference objects, the position of the navigating object is found by intersecting the circles centered at the reference objects and having the radii equal to the range measurements, as seen in Figure 6.1. Even though the circles intersect at two points, prior position information is used to choose the correct intersection point. Again a third range measurement is needed for 3D positioning of the object.

In position fixing technique, given an angular measurement accuracy, the positioning accuracy will decrease as the distance of the object from the references increases. However, in ranging technique, given a range measurement accuracy, the positioning accuracy doesn't change with the distances. If ranging and bearing measurement can be made from the same reference, then there is no need for a second reference as seen in Figure 6.1.

Dead reckoning is the process of estimating an object's position by advancing a known position by integrating the velocity of the object over the travel time. Since the velocity is measured in the body fixed frame, the attitude information is also necessary to obtain the direction of the travel in the reference frame.

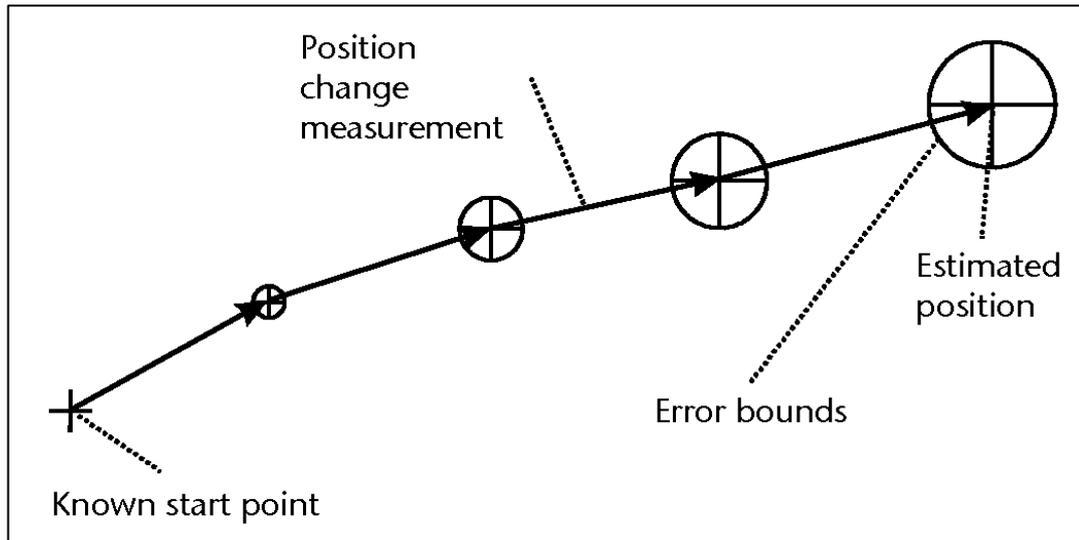


Figure 6.2 Dead-reckoning [5]

Figure 6.2 shows the navigation by dead reckoning technique. The dead reckoning solution is the sum of a sequence of position calculations including a certain amount of error. Since the position error is also integrated, the error in dead reckoning solution grows with time. If the attitude of the vehicle is not constant, decreasing the step size increases the accuracy of the position calculation.

Once initialized, dead reckoning technique provides an uninterrupted and continuous position solution but it suffers from the error growth with time. In contrast with that, accuracy of the position fixing solution doesn't degrade with time, but it may not be continuous and universally available. Considering the complementary characteristics of the two techniques, an integrated navigation system, which combines them and get the benefits of both, is rewarding, thus widely used.

## 6.2 Inertial Navigation Systems

An inertial navigation system (INS) is a three-dimensional dead-reckoning navigation system, composed of an inertial measurement unit (IMU) and a navigation processor [5]. After initialization, integrating outputs of IMU and compensating the gravitational effects, navigation processor produces the navigation

solution consisting of the position, velocity and attitude data. Figure 6.3 shows a basic schematic of an INS implementation.

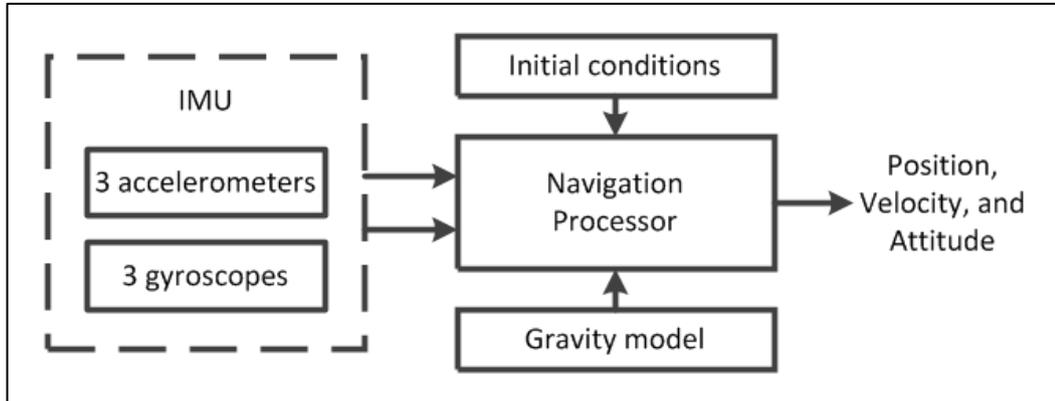


Figure 6.3 Inertial navigation system

## 6.2.1 Inertial Measurement Unit

An IMU usually comprises three mutually orthogonal accelerometers and three gyroscopes aligned with the accelerometers. Accelerometers measure the specific force applied on the vehicle, which is the acceleration due to sum of all forces except for gravity and gyroscopes measure the angular velocity of the vehicle. In this thesis, we have assumed a strap-down INS implementation, of which the accelerometers are aligned with the body coordinate frame of the vehicle, so that the attitude solution is used to transform the accelerometer output to the earth reference frame.

### 6.2.1.1 Accelerometers

Basic working principle of the accelerometers is the mass-spring-damper system. When a force is applied on a mass connected to the case by a spring, the position of the mass changes in the same direction proportional to the force applied. Most of the accelerometer designs are based on either pendulous or vibrating-beam structure. However both structures can be built either using conventional mechanical

construction or MEMS technology [5]. When the accelerometer triad is aligned with the axes of the body coordinate frame, the output is the specific force of the IMU body with respect to the inertial frame resolved along the accelerometer axes, represented by the vector  $f_{ib}^b$ .

### 6.2.1.2 Gyroscopes

There are three main types of gyroscope; namely, spinning-mass gyroscopes, optical gyroscopes and vibratory gyroscopes. Spinning-mass gyroscopes were the first examples of angular-rate sensors, which operate on the principles of conservation of angular momentum. Afterwards, the ring laser gyros (RLG) and the fiber-optic gyros (FOG) are developed, which in principle measure the change of path of a light beam due to rotation of the waveguide. Vibratory gyros are developed more recently, operating on the principle of detecting the Coriolis acceleration of the vibrating element when the gyro is rotated. MEMS gyros also use the vibratory principle [5]. The output of a gyroscope triad is the same for all types, which is the angular rate of the IMU body with respect to the inertial frame resolved along the gyro axes, represented by the vector  $\omega_{ib}^b$ .

### 6.2.1.3 Error Characteristics

The error contributions in the accelerometer and gyroscope outputs are similar and can be classify into following components: biases, scale factor errors, cross-coupling errors and random noise.

Bias is a constant error source which is independent of the measured specific force or angular rate. In addition to fixed bias (run-to-run bias), inertial sensors also exhibit in-run bias variation. Accelerometer bias is represented by  $b_a$ , whereas the gyro bias is represented by  $b_g$ . Bias is the dominant term in the overall error of inertial sensors. [5].

Scale factor error of an accelerometer is proportional to the measured specific force and scale factor error of a gyroscope is proportional to the measured angular rate on

their sensitive axes, represented by  $s_a = (s_{a,x}, s_{a,y}, s_{a,z})$  and  $s_g = (s_{g,x}, s_{g,y}, s_{g,z})$  respectively. Beside that, cross-coupling error is caused by the misalignment of the sensitive axes of inertial sensors and creates couplings between measurements of two orthogonal axes. Cross-coupling errors are represented by  $m_{a,\alpha\beta}$  for accelerometers, and  $m_{g,\alpha\beta}$  for gyroscopes, where  $\alpha$  and  $\beta$  refers to the coupled axes. Scale factor errors and cross-coupling errors for IMU containing an orthogonal accelerometer triad and an orthogonal gyro triad may be expressed in the matrix notation as follows:

$$M_a = \begin{bmatrix} s_{a,x} & m_{a,xy} & m_{a,xz} \\ m_{a,yx} & s_{a,y} & m_{a,yz} \\ m_{a,zx} & m_{a,zy} & s_{a,z} \end{bmatrix}, \quad M_g = \begin{bmatrix} s_{g,x} & m_{g,xy} & m_{g,xz} \\ m_{g,yx} & s_{g,y} & m_{g,yz} \\ m_{g,zx} & m_{g,zy} & s_{g,z} \end{bmatrix}$$

Because of the electrical or mechanical instabilities depending on their structure, the inertial sensors also exhibit random noise. The random noise on the accelerometer triad output is represented by the vector  $w_a = (w_{a,x}, w_{a,y}, w_{a,z})$ , and the random noise on the gyro triad output is represented by the vector  $w_g = (w_{g,x}, w_{g,y}, w_{g,z})$ .

Including the error contributions above, the output of the accelerometer and gyro can be expressed in the following equations [5]:

$$\tilde{f}_{ib}^b = b_a + (I_3 + M_a)f_{ib}^b + w_a \quad (6.1)$$

$$\tilde{\omega}_{ib}^b = b_g + (I_3 + M_g)\omega_{ib}^b + G_g f_{ib}^b + w_g \quad (6.2)$$

Where  $\tilde{f}_{ib}^b$  is the accelerometer-output specific force,  $\tilde{\omega}_{ib}^b$  is the gyro-output angular rate,  $f_{ib}^b$  and  $\omega_{ib}^b$  are their corresponding true values,  $G_g$  is acceleration dependent gyro bias and  $I_3$  is the identity matrix.

## 6.2.2 Inertial Navigation Processor

In an inertial navigation system, navigation processor produces the three-dimensional navigation solution, namely position, velocity and attitude solution, by integrating the IMU outputs. Before deriving the navigation equations, first the kinematic

quantities is expressed using the coordinate frames defined earlier, and then the used earth and gravity model will be expressed.

## 6.2.2.1 Kinematics

### 6.2.2.1.1 Attitude

Attitude defines the orientation of a body frame with respect to an inertial frame. If we represent the body frame as  $\alpha$  and the reference frame as  $\beta$ , attitude can be expressed as three successive rotations about the body frame axes, namely yaw ( $\psi_{\beta\alpha}$ ), pitch ( $\theta_{\beta\alpha}$ ), and roll ( $\phi_{\beta\alpha}$ ) rotations. An ideal way of representing the attitude is to use the coordinate transformation matrix (also known as the direction cosine matrix) which is a 3x3 matrix, denoted by  $C_{\alpha}^{\beta}$ . Coordinate transformation matrix is formed by first representing the roll, pitch and yaw rotations in the rotation matrix form and then multiplying each other in the reverse order to that in which they are applied. In this study, XYZ order is used:

$$\begin{aligned}
C_{\beta}^{\alpha} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_{\beta\alpha} & \sin\phi_{\beta\alpha} \\ 0 & -\sin\phi_{\beta\alpha} & \cos\phi_{\beta\alpha} \end{pmatrix} \begin{pmatrix} \cos\theta_{\beta\alpha} & 0 & -\sin\theta_{\beta\alpha} \\ 0 & 1 & 0 \\ \sin\theta_{\beta\alpha} & 0 & \cos\theta_{\beta\alpha} \end{pmatrix} \begin{pmatrix} \cos\psi_{\beta\alpha} & \sin\psi_{\beta\alpha} & 0 \\ -\sin\psi_{\beta\alpha} & \cos\psi_{\beta\alpha} & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
&= \begin{bmatrix} \cos\theta_{\beta\alpha}\cos\psi_{\beta\alpha} & \begin{pmatrix} -\cos\phi_{\beta\alpha}\sin\psi_{\beta\alpha} \\ +\sin\phi_{\beta\alpha}\sin\theta_{\beta\alpha}\cos\psi_{\beta\alpha} \end{pmatrix} & \begin{pmatrix} \sin\phi_{\beta\alpha}\sin\psi_{\beta\alpha} \\ +\cos\phi_{\beta\alpha}\sin\theta_{\beta\alpha}\cos\psi_{\beta\alpha} \end{pmatrix} \\ \cos\theta_{\beta\alpha}\sin\psi_{\beta\alpha} & \begin{pmatrix} \cos\phi_{\beta\alpha}\cos\psi_{\beta\alpha} \\ +\sin\phi_{\beta\alpha}\sin\theta_{\beta\alpha}\sin\psi_{\beta\alpha} \end{pmatrix} & \begin{pmatrix} -\sin\phi_{\beta\alpha}\cos\psi_{\beta\alpha} \\ +\cos\phi_{\beta\alpha}\sin\theta_{\beta\alpha}\sin\psi_{\beta\alpha} \end{pmatrix} \\ -\sin\theta_{\beta\alpha} & \sin\phi_{\beta\alpha}\cos\theta_{\beta\alpha} & \cos\phi_{\beta\alpha}\cos\theta_{\beta\alpha} \end{bmatrix} \quad (6.3)
\end{aligned}$$

In this manner, the successive rotations can be expressed as the multiplication of the coordinate transformation matrices:

$$C_{\gamma}^{\beta} = C_{\alpha}^{\beta} C_{\gamma}^{\alpha} \quad (6.4)$$

The reverse rotation is also simply represented by the transpose of the original coordinate transformation matrix:

$$C_{\beta}^{\alpha} = (C_{\alpha}^{\beta})^T \quad (6.5)$$

Coordinate transformation matrices are also used to transform a vector from one set of resolving axes to another:

$$x_{\beta\alpha}^{\delta} = C_{\gamma}^{\delta} x_{\beta\alpha}^{\gamma} \quad (6.6)$$

In this notation, the lower index represents the “from” coordinate frame,  $\gamma$ , and the upper index represents the “to” coordinate frame,  $\delta$ .

### 6.2.2.1.2 Angular Rate

The angular rate vector,  $\omega_{\beta\alpha}^{\gamma}$ , is defined earlier to represent the rate of rotation of the  $\alpha$ -frame axes with respect to the  $\beta$ -frame axes, resolved about the  $\gamma$ -frame axes. In navigation equations, the angular rate tensor representation is used which is the skew-symmetric matrix representation of the angular rate vector:

$$\Omega_{\beta\alpha}^{\gamma} = [\omega_{\beta\alpha}^{\gamma} \wedge] = \begin{pmatrix} 0 & -\omega_{\beta\alpha 3}^{\gamma} & \omega_{\beta\alpha 2}^{\gamma} \\ \omega_{\beta\alpha 3}^{\gamma} & 0 & -\omega_{\beta\alpha 1}^{\gamma} \\ -\omega_{\beta\alpha 2}^{\gamma} & \omega_{\beta\alpha 1}^{\gamma} & 0 \end{pmatrix} \quad (6.7)$$

If we use the small angle approximation, applied in the limit  $\delta t \rightarrow 0$ , then the time derivative of the coordinate transformation matrix can be expressed as [5]:

$$\dot{C}_{\beta}^{\alpha} = \Omega_{\alpha\beta}^{\alpha} C_{\beta}^{\alpha} \quad (6.8)$$

### 6.2.2.1.3 Position

The Cartesian position  $r_{\beta\alpha}^{\gamma} = (x_{\beta\alpha}^{\gamma}, y_{\beta\alpha}^{\gamma}, z_{\beta\alpha}^{\gamma})$  is used to represent the origin of frame  $\alpha$  with respect to the origin of frame  $\beta$ , resolved about the axes of frame  $\gamma$ , where  $x$ ,  $y$ , and  $z$  are the projections of the position in the  $x$ ,  $y$ , and  $z$  axes of the  $\gamma$  frame. As mentioned earlier, position may be resolved in a different frame by multiplying by the corresponding coordinate transformation matrix:

$$r_{\beta\alpha}^{\gamma} = C_{\beta}^{\gamma} r_{\beta\alpha}^{\beta} \quad (6.9)$$

#### 6.2.2.1.4 Velocity

Velocity is defined as the rate of change of the position of the origin of an object frame with respect to the origin and axes of a reference frame [5]. If resolved about the axes of a third frame, the velocity of frame  $\alpha$  with respect to frame  $\beta$ , resolved about the axes of frame  $\gamma$ , can be expressed as:

$$v_{\beta\alpha}^{\gamma} = C_{\beta}^{\gamma} \dot{r}_{\beta\alpha}^{\beta} \quad (6.10)$$

We can derive the time derivative of  $r_{\beta\alpha}^{\gamma}$  by differentiating both sides of the Equation (6.9) and then substituting Equation (6.10) in that:

$$\begin{aligned} \dot{r}_{\beta\alpha}^{\gamma} &= \dot{C}_{\beta}^{\gamma} r_{\beta\alpha}^{\beta} + C_{\beta}^{\gamma} \dot{r}_{\beta\alpha}^{\beta} \\ &= \dot{C}_{\beta}^{\gamma} r_{\beta\alpha}^{\beta} + v_{\beta\alpha}^{\gamma} \end{aligned} \quad (6.11)$$

Equation (6.11) implies that,  $v_{\beta\alpha}^{\gamma}$  is not equal to the time derivative of  $r_{\beta\alpha}^{\gamma}$  unless there is no angular motion of the resolving frame,  $\gamma$ , with respect to the reference frame,  $\beta$ .

#### 6.2.2.1.5 Acceleration

Acceleration is defined as the second time derivative of the position of the origin of an object frame with respect to the origin and axes of a reference frame [5]. Again, if it is resolved about the axes of a third frame, the acceleration of frame  $\alpha$  with respect to frame  $\beta$ , resolved about the axes of frame  $\gamma$ , can be expressed as:

$$a_{\beta\alpha}^{\gamma} = C_{\beta}^{\gamma} \ddot{r}_{\beta\alpha}^{\beta} \quad (6.12)$$

We can derive the time derivative of  $v_{\beta\alpha}^{\gamma}$  by differentiating both sides of the Equation (6.10) and then substituting Equation (6.12) in that:

$$\begin{aligned}
\dot{v}_{\beta\alpha}^{\gamma} &= \dot{C}_{\beta}^{\gamma} \dot{r}_{\beta\alpha}^{\beta} + C_{\beta}^{\gamma} \ddot{r}_{\beta\alpha}^{\beta} \\
&= \dot{C}_{\beta}^{\gamma} \dot{r}_{\beta\alpha}^{\beta} + a_{\beta\alpha}^{\gamma}
\end{aligned}
\tag{6.13}$$

Equation (6.13) implies that,  $a_{\beta\alpha}^{\gamma}$  is not equal to the time derivative of  $v_{\beta\alpha}^{\gamma}$  unless there is no angular motion of the resolving frame,  $\gamma$ , with respect to the reference frame,  $\beta$ . If we differentiate both sides of Equation (6.11) and substitute Equation (6.13) in that, we get the expression for the second time derivative of  $r_{\beta\alpha}^{\gamma}$  as:

$$\begin{aligned}
\ddot{r}_{\beta\alpha}^{\gamma} &= \ddot{C}_{\beta}^{\gamma} r_{\beta\alpha}^{\beta} + \dot{C}_{\beta}^{\gamma} \dot{r}_{\beta\alpha}^{\beta} + \dot{v}_{\beta\alpha}^{\gamma} \\
&= \ddot{C}_{\beta}^{\gamma} r_{\beta\alpha}^{\beta} + 2\dot{C}_{\beta}^{\gamma} \dot{r}_{\beta\alpha}^{\beta} + a_{\beta\alpha}^{\gamma}
\end{aligned}
\tag{6.14}$$

Using Equation (6.8), Equation (6.9) and Equation (6.11), Equation (6.14) can be expressed as:

$$\ddot{r}_{\beta\alpha}^{\gamma} = -\left(\Omega_{\beta\gamma}^{\gamma} \Omega_{\beta\gamma}^{\gamma} + \dot{\Omega}_{\beta\gamma}^{\gamma}\right) r_{\beta\alpha}^{\gamma} - 2\Omega_{\beta\gamma}^{\gamma} \dot{r}_{\beta\alpha}^{\gamma} + a_{\beta\alpha}^{\gamma}
\tag{6.15}$$

In the Equation (6.15) the first term on the right hand side represents the centrifugal acceleration while the second term represents the Coriolis acceleration.

## 6.2.2.2 Earth Model

In our application, we need the position solution with respect to the Earth's surface. To obtain this solution, we define a reference surface with respect to the center and the axes of the earth and we also define a set of coordinates with respect to this surface, which is called the latitude, longitude and height. We also need the Earth's rotation to transform the inertially referenced IMU measurements to Earth referenced coordinate frame.

### 6.2.2.2.1 The Ellipsoid Model of the Earth's Surface

An approximation to the surface of the Earth is an ellipsoid fitted in the main sea level. The ellipsoid is commonly defined in terms of the equatorial radius and the eccentricity of the ellipsoid,  $e$ . The eccentricity is defined by

$$e = \sqrt{1 - \frac{R_p^2}{R_0^2}} \quad (6.16)$$

where  $R_0$  is the length of semi-major axis which is the distance from the center of the ellipsoid to any point on the equator and  $R_p$  is the length of semi-minor axis which is the distance from the center to either pole. Semi-major, semi-minor axes, equator and the north pole can be seen in Figure 6.4.

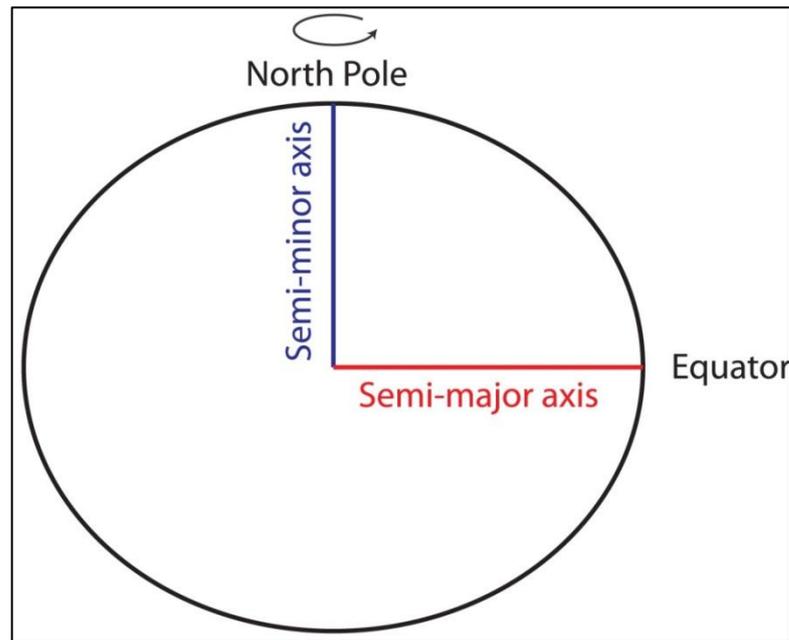


Figure 6.4 The ellipsoid model of the Earth's surface

World Geodetic System 1984 [33] gives the measures the length of semi-minor and semi-major axes and the eccentricity of the Earth as:

$$R_0 = 6,378,137.0 \text{ m,}$$

$$R_p = 6,356,752.3141 \text{ m,}$$

$$e = 0.0818191908425$$

### 6.2.2.2 Curvilinear position

Curvilinear position is defined as the position of an object with respect to the Earth's surface and described by three orthogonal coordinates, aligned with the axes of the local navigation frame, namely the *height* ( $h$ ), the *latitude* ( $L$ ), and the *longitude* ( $\lambda$ ). Height is the distance from the object to the surface along the normal to that surface, latitude is the north-south axis coordinate of the point on the surface where that normal intersects and finally longitude is the coordinate of that point in the east-west axis, as seen in Figure 6.5.

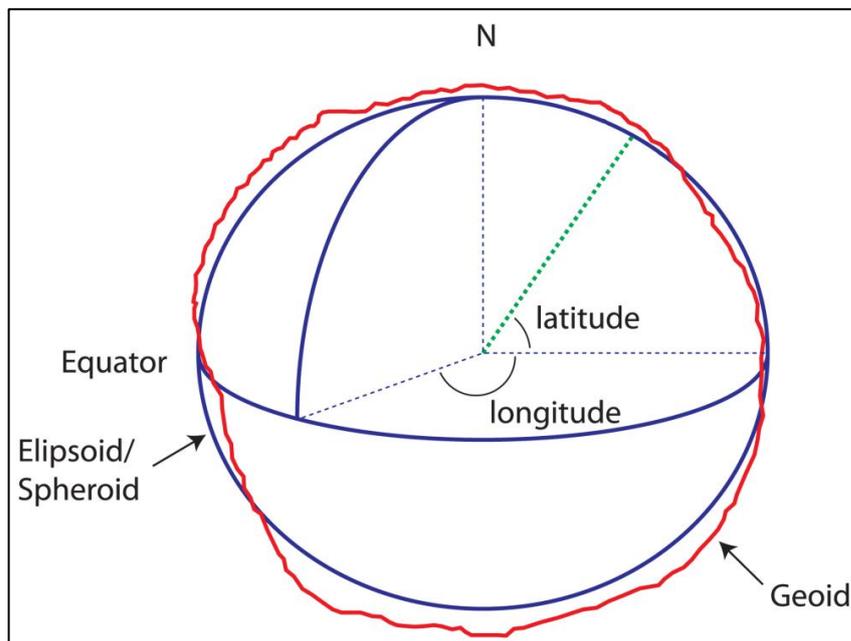


Figure 6.5 Curvilinear position components

To transform the Cartesian ECEF position to the curvilinear position, we first define the transverse radius of curvature  $R_E$ , which is the radius of curvature for east-west motion, and given by:

$$R_E(L) = \frac{R_0}{\sqrt{1 - e^2 \sin^2 L}} \quad (6.17)$$

Then the Cartesian ECEF position transforms to the curvilinear position by following equations:

$$\begin{aligned}\sin L_b &= \frac{z_{eb}^e}{(1 - e^2)R_E(L_b) + h_b} \\ \tan \lambda_b &= \frac{y_{eb}^e}{x_{eb}^e} \\ h_b &= \frac{\sqrt{(x_{eb}^e)^2 + (y_{eb}^e)^2}}{\cos L_b} - R_E(L_b)\end{aligned}\tag{6.18}$$

### 6.2.2.2.3 Earth Rotation

The ECEF rotates with respect to the ECI, clockwise about the common z-axis. The Earth-rotation vector resolved in these axes is given by

$$\omega_{ie}^i = \omega_{ie}^e = \begin{pmatrix} 0 \\ 0 \\ \omega_{ie} \end{pmatrix}\tag{6.19}$$

where, according to WGS 84, the value of the Earth's angular rate is:

$$\omega_{ie} = 7.292115 \times 10^{-5} \text{ rad s}^{-1}.$$

### 6.2.2.2.4 Specific Force, Gravitation, and Gravity

*Specific force* is the sum of all forces acting on a body with respect to an inertial frame except for the gravitation. Accelerometers measure the specific force in the body frame, thus the measured specific force is  $f_{ib}^\gamma$ , where  $\gamma$  is any resolving frame. *Gravitation* is the fundamental mass attraction force and it does not incorporate any centripetal components. Acceleration due to gravitation force can be represented by  $\gamma_{ib}^\gamma$ . Specific force varies with acceleration and the acceleration due to the gravitation force as:

$$f_{ib}^Y = a_{ib}^Y - \gamma_{ib}^Y \quad (6.20)$$

The specific force sensed by a stationary object with respect to the inertial frame is the reaction to the acceleration due to *gravity*, which can be expressed as:

$$g_b^Y = -f_{ib}^Y|_{a_{eb}^Y=0, v_{eb}^Y=0} \quad (6.21)$$

Substituting Equation (6.20) into Equation (6.21) we get the acceleration due to gravity as:

$$g_b^Y = \gamma_{ib}^Y - \Omega_{ie}^Y \Omega_{ie}^Y r_{eb}^Y \quad (6.22)$$

where the second term on the right hand side of Equation (6.22) is the centrifugal acceleration acting on the body due to the rotation of the earth.

In this study, World Geodetic System 1984 [33] representation of Earth's gravity is used to calculate the gravitational force acting on the vehicle as a function of the ECEF position of the vehicle.

### 6.2.2.3 Inertial Navigation Equations

The inertial navigation equations consist of four steps: attitude update, transformation of the specific force from the body frame to the earth frame, velocity update, and position update. To transform the specific force into acceleration the gravitation model is also included in the equations. Figure 6.6 shows the block diagram of the inertial navigation equations.

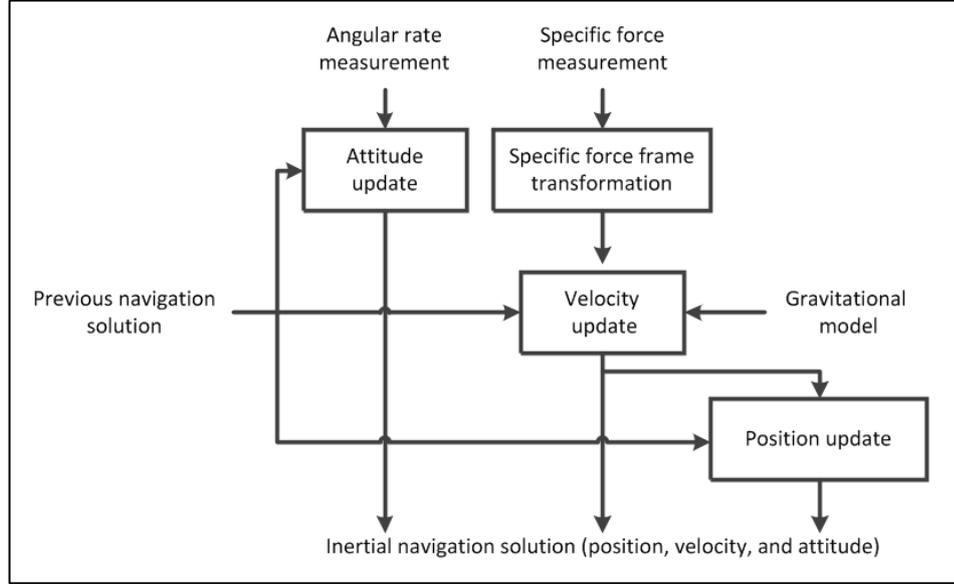


Figure 6.6 Inertial navigation equations

The following equations are used to update the attitude, velocity and position from  $t$  to  $t + \tau_i$ , where  $\tau_i$  is the sampling interval of the inertial sensors, which is chosen as 0.01s in this study. We have used the ECEF frame as the reference frame and the resolving axes since the satellite navigation solutions are also referenced and resolved in this frame. Thus, the earth rotation is also included to express the inertial measurement in the rotating reference frame equations.

### 6.2.2.3.1 Attitude Update

In this step gyro-output angular rate measurement  $\omega_{ib}^b$  is used to update the attitude solution which is also the body-to-Earth frame coordinate transformation matrix  $C_b^e$ .

The attitude is the solution of the differential equation:

$$\begin{aligned} \dot{C}_b^e &= C_b^e \Omega_{eb}^b \\ &= C_b^e \Omega_{ib}^b - \Omega_{ie}^e C_b^e \end{aligned} \quad (6.23)$$

where  $\Omega_{ib}^b$  is the gyro-output angular rate tensor and  $\Omega_{ie}^e$  is the Earth's rotation tensor. Since the Earth's rotation with respect to inertial frame is only about the common  $z$ -axis,  $\Omega_{ie}^e$  can be expressed as:

$$\Omega_{ie}^e = \begin{bmatrix} 0 & -\omega_{ie}^e & 0 \\ \omega_{ie}^e & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (6.24)$$

Here,  $\omega_{ie}^e$  is taken as  $7.292115e-5$  rad/s.

### 6.2.2.3.2 Specific-Force Frame Transformation

Using the attitude solution  $C_b^e$ , the accelerometer-output specific-force measurement is transferred from the body coordinate frame to ECEF frame as:

$$f_{ib}^e = C_b^e f_{ib}^i \quad (6.25)$$

### 6.2.2.3.3 Velocity Update

Since we used the Earth frame for both reference frame and resolving frame, differentiating both sides of Equation (6.11),  $\dot{v}_{eb}^e$  can be expressed as:

$$\dot{v}_{eb}^e = \dot{r}_{eb}^e = \dot{r}_{ib}^e \quad (6.26)$$

Using Equation (6.15), and assuming the Earth's rotation rate is constant, thus  $\dot{\Omega}_{ie}^e = 0$ , we can rewrite Equation (6.26) as:

$$\dot{v}_{eb}^e = -\Omega_{ie}^e \Omega_{ie}^e r_{ib}^e - 2\Omega_{ie}^e \dot{r}_{eb}^e + a_{ib}^e \quad (6.27)$$

Substituting Equation (6.20) and Equation (6.22), in Equation (6.27), we get the final expression for the velocity update differential equation as:

$$\dot{v}_{eb}^e = f_{ib}^e + g_b^e(r_{eb}^e) - 2\Omega_{ie}^e v_{eb}^e \quad (6.28)$$

#### 6.2.2.3.4 Position Update

Again since we used the Earth frame for both reference frame and resolving frame, we can simply utilize Equation (6.11), as the position update can be calculated by the velocity solution as:

$$\dot{r}_{eb}^e = v_{eb}^e \quad (6.29)$$

#### 6.2.2.4 INS Error Propagation

INS error is defined as the difference between any INS-indicated quantity and the corresponding true quantity. The main error source in an inertial navigation solution is the error in accelerometer and gyro measurements. The navigation equations integrate the accelerometer and gyro biases, thus the position, velocity and attitude errors grow with time. If we denote the INS-indicated quantity with ( $\sim$ ), the Cartesian position and velocity errors can be expressed as [5]:

$$\delta r_{\beta\alpha}^\gamma = \tilde{r}_{\beta\alpha}^\gamma - r_{\beta\alpha}^\gamma \quad (6.30)$$

$$\delta v_{\beta\alpha}^\gamma = \tilde{v}_{\beta\alpha}^\gamma - v_{\beta\alpha}^\gamma \quad (6.31)$$

The attitude error can be calculated using the coordinate transformation matrix:

$$\delta C_\beta^\alpha = \tilde{C}_\beta^\alpha C_\alpha^\beta \quad (6.32)$$

where the small angle approximation applies, the attitude error can be expressed in terms of the coordinate transform as:

$$\left[ \delta \psi_{\beta\alpha}^\beta \wedge \right] \approx \delta C_\alpha^\beta - I_3 \quad (6.33)$$

## **6.3 External Aids**

### **6.3.1 Global Navigation Satellite System (GNSS)**

GNSS is a position fixing navigation system which provides a three-dimensional positioning solution using passive ranging method utilizing radio signals transmitted by orbiting satellites. Currently the United States NAVSTAR Global Positioning System (GPS) and the Russian GLONASS are the only operational global navigation satellite systems, whereas China's Compass navigation system and European Union's Galileo positioning systems are under development.

A satellite navigation system consists of three components: the space segment, the control or ground segment, and the user segment. The space segment consists of the satellites which are distributed according to maximum coverage and they broadcast radio signals to both the control segment and the user segment. The radio signals include the ranging codes and navigation data messages. The user equipment uses the ranging codes to determine the time at which the received signals were transmitted. The navigation data messages are used to give information about the timing parameters and the satellite orbits. Positions of the control segments are precisely known and they have synchronized clocks, so that their ranging measurements are used to determine the satellite orbits and calibrate the satellite clocks.

Figure 6.7 shows the block diagram of the user equipment. The antenna converts the incoming radio signals from satellites into electrical signals; the receiver demodulates the electrical signals and provides timing information; the ranging processor uses this timing information to determine the range from the antenna to each of the satellite whose signal is received; and the navigation processor uses those ranging measurements to estimate a position and velocity solution.

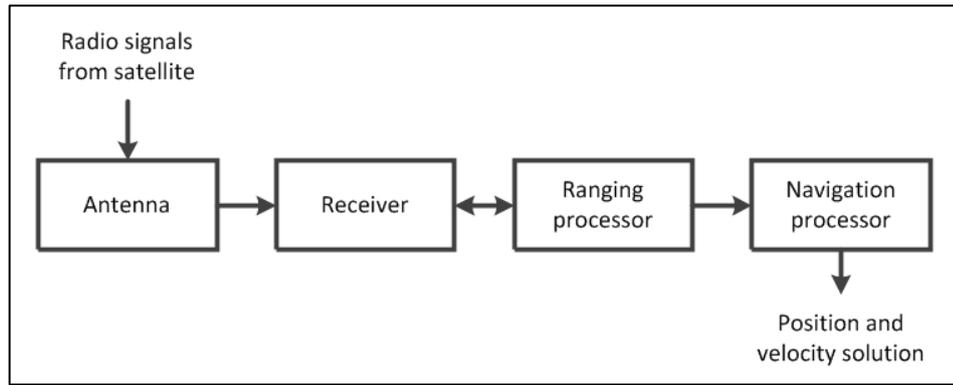


Figure 6.7 GNSS user equipment

Figure 6.8 shows the positions of the GNSS satellites  $r_{isX}^i$ , the position of the GNSS antenna  $r_{ia}^i$ , both referenced and resolved in the inertial frame, and the range measurements corresponding to each satellite as  $\tilde{\rho}_{cX}$ , called as the pseudorange. The position and velocity of the GNSS user equipment is determined by the triangulation method, using the pseudo-ranges and pseudo-range rates.

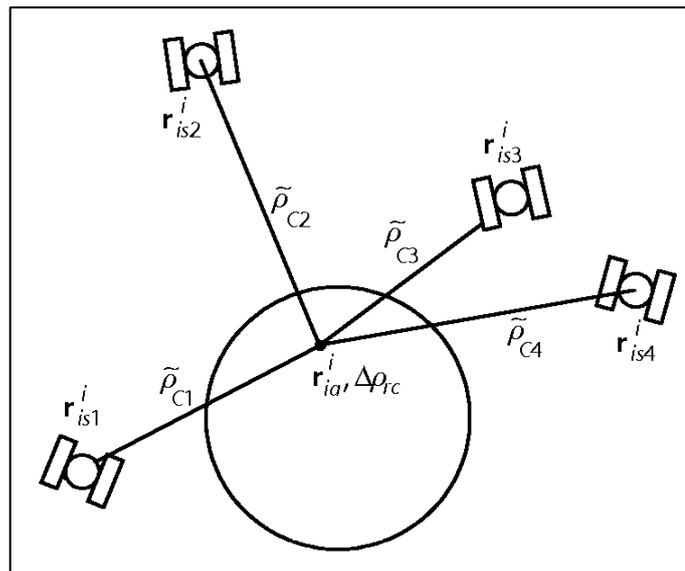


Figure 6.8 GNSS ranging [5]

The estimated states referenced and resolved in the earth frame can be shown in the vector form as:

$$\hat{x}_G = \begin{pmatrix} \hat{r}_{eaG}^e \\ \hat{v}_{eaG}^e \end{pmatrix} \quad (6.34)$$

GNSS position error changes according to the signal frequency, multipath environment, and the modulation scheme. According to [5], the position error standard deviation for a single frequency, short-range multipath model and BPSK(1) modulation setting is found to be 3.7 m. on the horizontal (radial) plane and 6.0 m. on the vertical plane. Appendix B lists the GNSS position error standard deviation for other settings.

### 6.3.2 Acoustic Navigation

Acoustic navigation systems are classified by the length of the baseline, which is the distance between the active sensing acoustic transceivers. Today, three types of acoustic navigation systems differ in the baseline are available: long baseline (LBL) of the order 100-6000m, short baseline (SBL) of the order 20-50m, and ultra-short baseline (USBL) of less than 10cm. [34]. In LBL and SBL systems, the underwater vehicle position is calculated by the triangulation of acoustic ranges within a network of surveyed transponders, whereas in USBL navigation a sonar array is employed to determine the range and bearing to the underwater vehicle [35]. Since the transceiver array is contained in a single unit, USBL system is advantageous compared to other acoustic navigation systems, in the means of ease of installation.

Figure 6.9 shows the basics of USBL navigation. The transceiver device is placed on a surface vessel, whereas the transponder device is placed on the navigation underwater vehicle. Transceiver device contains three or more transducers separated by the length of baseline.

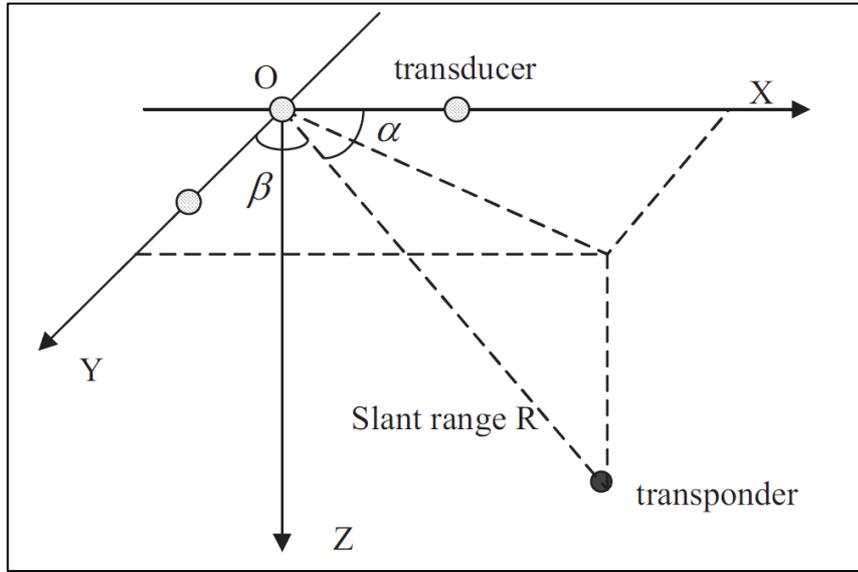


Figure 6.9 USBL ranging and bearing [36]

To determine the position of the underwater vehicle, first the transceiver emits an acoustic pulse which is replied back by transponder. The reply is received by the transducer array of the transceiver. The bearing angles  $\alpha$  and  $\beta$  are derived by the phase difference of the arriving sound wave between individual transponders and the slant range  $R$  is determined using the time of flight technique [36]. Using the range and bearing measurements, the position of the transponder located on the underwater vehicle can be expressed with respect to transceiver located on the surface vessel and resolved in the transducer array coordinate axes as following:

$$r_{ab}^a = \begin{pmatrix} R \cos \alpha \\ R \cos \beta \\ R \sqrt{1 - \cos^2 \alpha - \cos^2 \beta} \end{pmatrix} \quad (6.35)$$

According to the product survey done by Hydro International in 2008 [37], the commercial USBL systems are capable of covering ranges from 1000m to 10000m, with an angular coverage of  $90^\circ$  to  $200^\circ$  on the vertical plane and full  $360^\circ$  on the horizontal plane. The range accuracies are reported as differing from 0.1m to 1m and the angular accuracies are generally less than  $1^\circ$ .

### **6.3.3 Attitude Measurement**

The 3-axis attitude can be measured by using a three-axis fluxgate magnetometer for heading together with two tilt sensors for the roll and pitch angles [38]. The fluxgate magnetometers are widely used to sense the magnitude and direction of a magnetic field. A three-axis fluxgate magnetometer can be used to determine the heading by measuring the Earth's magnetic field which is always pointing towards the magnetic north in the Northern Hemisphere, if the magnetometer is lying on the local horizontal plane (zero roll and pitch angles). If the magnetometer is not lying on the local horizontal plane, to compensate the heading measurement, the tilt angles (roll and pitch) are needed to be determined using a tilt sensor (inclinometer). Tilt sensors sense the direction of gravity and common tilt measuring devices include accelerometers, electrolytic (fluid) based tilt sensors, and gimballed mechanical structures [38]. The accuracy of the magnetometer measurement is affected by the distortions in the Earth's magnetic field, i.e., hard and soft iron effects, and the accuracy of the tilt sensor measurement are affected by the acceleration. Hard and soft iron effects can be compensated by a calibration procedure and the tilt errors can be neglected since our underwater vehicle will not be influenced by high dynamics. When the earth's magnetic field is known precisely and the distortions are compensated, the heading error can be as low as  $1^\circ$  [38], whereas the precision of commonly used tilt sensors are on the order of  $0.1^\circ$ - $1^\circ$  [35].

### **6.4 Integrated Navigation**

Even though inertial navigation systems provide continuous and high-bandwidth output and exhibit low short-term noise, since the inertial sensor errors are integrated through the navigation equations, the accuracy of the inertial solution degrades with time significantly. Oppositely GNSS positioning accuracy does not degrade with time and limited to a few meters, but compared to inertial systems, GNSS data output rate is low, and suffers from the environment obstruction and interference, thus GNSS does not guarantee a continuous navigation solution. Considering the

complementary behavior of those two systems, an integrated navigation system which combines the positive features of both systems is suggested. However since the GNSS signals are not available underwater, an acoustic navigation system is included to link the GNSS navigation solution to the INS underwater. Since the underwater vehicle is designed for reconnaissance missions, which requires a high data output rate for video and image transfer, a cabled solution is adopted, thus the positioning solution from the GNSS and acoustic navigation will also be transferred through the cable. Beside that external attitude sensors are also added to the integrated navigation system to correct any drifting attitude errors. Figure 6.10 shows the overall structure of the integrated navigation system.

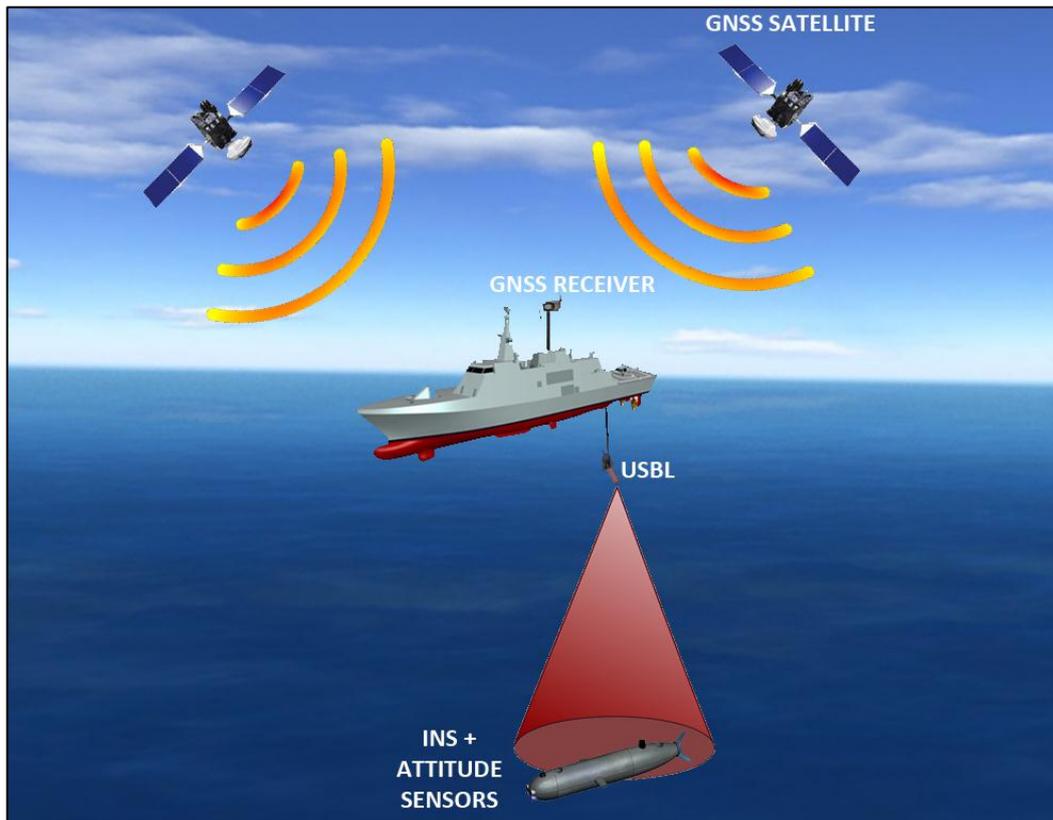


Figure 6.10 Overall integrated navigation system

### 6.4.1 Integration Architecture

In the integrated navigation system, because of its simplicity and redundancy, a loosely-coupled integration architecture is adopted. A Kalman filter based estimation algorithm compares the INS solution with the outputs of GNSS and acoustic positioning system, and external attitude measurements and estimates corrections to the inertial position, velocity and attitude solutions. The corrected INS solution forms the integrated navigation solution. This architecture also provides a continuous navigation solution even in the cases when no external aids are available. The integration architecture is illustrated in Figure 6.11.

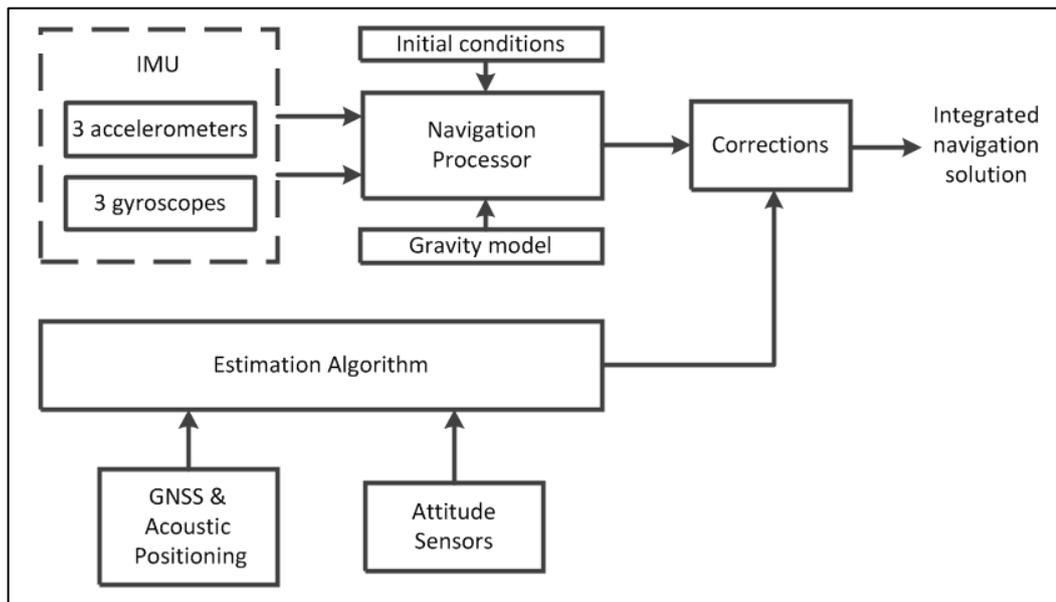


Figure 6.11 Integrated navigation system

### 6.4.2 State Selection and System Model

An error-state implementation is adopted where the integration algorithm estimates the position, velocity and attitude errors. The navigation solution is expressed in the ECEF coordinates, therefore the error states are also expressed in ECEF coordinates

as:  $\delta r_{eb}^e$ ,  $\delta v_{eb}^e$ , and  $\delta \psi_{eb}^e$  respectively representing the position, velocity and attitude error. Together with those the accelerometer bias  $b_a$ , and gyro bias  $b_g$  is also included in the estimated states, since those errors also contribute to the error growth in velocity and attitude. The state vector combining 15 error states expressed in ECEF coordinate is

$$x = \begin{pmatrix} \delta r_{eb}^e \\ \delta v_{eb}^e \\ \delta \psi_{eb}^e \\ b_a \\ b_g \end{pmatrix} \quad (6.36)$$

The system model is obtained by time differentiation of each state. The main source of the attitude error in ECEF coordinates is the angular rate error referenced to earth:

$$\delta \dot{\psi}_{eb}^e \approx \hat{C}_b^e (\tilde{\omega}_{eb}^b - \omega_{eb}^b) \quad (6.37)$$

where the accent character (^) denotes an estimated quantity, ( $\sim$ ) denotes a measured quantity and no accent character denotes true counterpart. The angular rate measured is the sum of body rotation and Earth rotation. Decomposing those components gives:

$$\begin{aligned} \delta \dot{\psi}_{eb}^e &\approx \hat{C}_b^e \delta \omega_{ib}^b - \hat{C}_b^e (\tilde{C}_b^e - C_b^e) \omega_{ie}^e \\ &= \hat{C}_b^e \delta \omega_{ib}^b - \Omega_{ie}^e \delta \psi_{eb}^e \end{aligned} \quad (6.38)$$

In terms of the estimated states, the attitude error can be expressed as:

$$\delta \dot{\psi}_{eb}^e = \hat{C}_b^e b_g - \Omega_{ie}^e \delta \psi_{eb}^e \quad (6.39)$$

The time derivative of ECEF-frame velocity is:

$$\dot{v}_{eb}^e = f_{ib}^e + g_b^e(r_{eb}^e) - 2\Omega_{ie}^e v_{eb}^e \quad (6.40)$$

The time derivative of the velocity error is:

$$\delta \dot{v}_{eb}^e = \tilde{f}_{eb}^e - f_{eb}^e + \tilde{g}_b^e(\tilde{r}_{eb}^e) - g_b^e(r_{eb}^e) - 2\Omega_{ie}^e \delta v_{eb}^e \quad (6.41)$$

The accelerometer error in the earth coordinate is due to the specific force measurement error in body coordinates. Therefore,

$$\tilde{f}_{eb}^e - f_{eb}^e = \tilde{C}_b^e \tilde{f}_{eb}^e - C_b^e f_{eb}^e \approx \hat{C}_b^e (\tilde{f}_{ib}^b - f_{ib}^b) + (\tilde{C}_b^e - C_b^e) \hat{f}_{ib}^b \quad (6.42)$$

The accelerometer bias is the only source of specific force measurement error included in the estimated states, therefore,

$$\tilde{f}_{ib}^i - f_{ib}^i = \delta f_{ib}^b \approx b_a \quad (6.43)$$

The small angle approximation on the attitude error yields:

$$\tilde{C}_b^e - C_b^e = (\delta C_b^e - I_3) C_b^e \approx [\delta \psi_{eb}^e \wedge] C_b^e \quad (6.44)$$

The gravity error depends on the position error, and variation of gravity due to change in local position is small in our application, therefore neglected. Thus the time derivative of velocity error is

$$\delta \dot{v}_{eb}^e \approx \hat{C}_b^e b_a - (\hat{C}_b^e \hat{f}_{ib}^b) \wedge \delta \psi_{eb}^e - 2\Omega_{ie}^e \delta v_{eb}^e \quad (6.45)$$

The time derivative of the position error in the ECEF representation is:

$$\delta \dot{r}_{eb}^e = \delta v_{eb}^e \quad (6.46)$$

The accelerometer and gyro measurement noises which cause a random walk contribution on the velocity and attitude error states can be modeled as white noise. The time variation of the accelerometer and gyro biases can also be modeled as white noise [5]. Therefore the INS system model can be expressed in the standard form as:

$$\dot{x} = Fx + Gw \quad (6.47)$$

$$\begin{pmatrix} \dot{\delta r}_{eb}^e \\ \dot{\delta v}_{eb}^e \\ \dot{\delta \psi}_{eb}^e \\ \dot{b}_a \\ \dot{b}_g \end{pmatrix} = \begin{bmatrix} 0_3 & I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & -2\Omega_{ie}^e & -(\hat{C}_b^e \hat{f}_{ib}^b) \wedge & \hat{C}_b^e & 0_3 \\ 0_3 & 0_3 & -\Omega_{ie}^e & 0_3 & \hat{C}_b^e \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{bmatrix} \begin{pmatrix} \delta r_{eb}^e \\ \delta v_{eb}^e \\ \delta \psi_{eb}^e \\ b_a \\ b_g \end{pmatrix} \\ + \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ \hat{C}_b^e & 0_3 & 0_3 & 0_3 \\ 0_3 & \hat{C}_b^e & 0_3 & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \begin{pmatrix} v_a \\ v_g \\ v_{ab} \\ v_{gb} \end{pmatrix}$$

where  $v_a \sim N(0, \sigma_{v_a}^2)$ ,  $v_g \sim N(0, \sigma_{v_g}^2)$ ,  $v_{ab} \sim N(0, \sigma_{v_{ab}}^2)$ , and  $v_{ag} \sim N(0, \sigma_{v_{ag}}^2)$  are the accelerometer noise, gyro noise, accelerometer bias noise and gyro bias parameters respectively.

The digital implementation of the integrated navigation system requires discrete representation of the state and error model. The discrete equivalent of the continuous differential equations can be expressed as:

$$x_{k+1} = \Phi_k x_k + w_k \quad (6.48)$$

where  $X_k$  is the state vector at time step  $k$ , and  $\Phi_k$  is the state transition matrix. The solution of Equation (6.48), can be denoted as:

$$x_{k+1} = e^{F(t_{k+1}-t_k)} x_k + \int_{t_k}^{t_{k+1}} e^{F(t_{k+1}-t_k-\tau)} G w(\tau) d\tau \quad (6.49)$$

Assuming that the system matrix  $F$  is constant over the integration time period  $\Delta t = t_{k+1} - t_k$ ,  $\Phi_k$  can be obtained by,

$$\Phi_k = e^{F\Delta t} = I + F\Delta t + \frac{(F\Delta t)^2}{2!} + \dots \quad (6.50)$$

On the other hand, the discrete time equivalent of the process noise  $Q_k$  can be obtained by using the following method [39]:

$$\text{Let } A = \begin{pmatrix} -F & G W G^T \\ 0 & F^T \end{pmatrix}, \text{ and } B = e^A = \begin{pmatrix} \dots & \Phi^{-1} Q_k \\ 0 & \Phi^T \end{pmatrix}$$

then  $Q_k$  can be obtained by multiplying the lower right part and upper right parts of matrix  $B$ , as

$$Q_k = \Phi \Phi^{-1} Q_k \quad (6.51)$$

where  $W$  is the power spectral density matrix of the process noise

### 6.4.3 Measurement Model

In the error-state implementation of the integrated navigation system, the differences between measurement outputs from the external aids and predictions of those measurements from the inertial navigation system are used to update the state vector. For example, if we represent a GNSS based measurement as  $\tilde{m}_G$ , and the prediction of that measurement obtained from the inertial navigation solution as  $\tilde{m}_I$ , then the Kalman filter measurement  $z$  can be obtained from [5]:

$$z = \tilde{m}_G - \tilde{m}_I \quad (6.52)$$

In our application, the position error measurement is obtained by comparing the INS estimated position solution with the position solution derived from the GNSS+acoustic navigation system. Also the attitude error measurement is obtained by comparing the INS estimated attitude solution with the attitude measurements of the magnetometer and inclinometer system. Therefore the measurement innovation vector in ECEF coordinates can be expressed as:

$$\delta z_k^{e-} = \begin{pmatrix} \delta r_{eb}^e \\ \delta \psi_{eb}^e \end{pmatrix} = \begin{pmatrix} \hat{r}_{eaG}^e - \hat{r}_{eb}^e - \hat{C}_b^e l_{ba}^b \\ \hat{\psi}_{eb}^e - \hat{\psi}_{ebA}^e \end{pmatrix} \quad (6.53)$$

Using Equation (6.32) Equation (6.32), INS-indicated coordinate transformation matrix can be derived as:

$$\tilde{C}_b^Y = \left( I_3 + \left[ \delta \psi_{Y\beta}^Y \wedge \right] \right) C_b^Y \quad (6.54)$$

Therefore

$$\hat{C}_b^\gamma l_{ba}^b = C_b^\gamma l_{ba}^b - [(C_b^\gamma l_{ba}^b) \wedge] \delta\psi_{\gamma\beta}^\gamma \quad (6.55)$$

For a Kalman filter based estimator, the measurement matrix is given by:

$$H_k = \left. \frac{\partial h(x)}{\partial x} \right|_{x=\hat{x}_k^-} = \left. \frac{\partial z(x)}{\partial x} \right|_{x=\hat{x}_k^-} \quad (6.56)$$

Using the measurement vector in Equation (6.53), the state vector in Equation (6.36), and using Equation (6.55), the measurement matrix in ECEF coordinates can be formed as:

$$H_k = \begin{bmatrix} -I_3 & 0_3 & [(\hat{C}_b^e l_{ba}^b) \wedge] & 0_3 & 0_3 \\ 0_3 & 0_3 & -I_3 & 0_3 & 0_3 \end{bmatrix} \quad (6.57)$$

However, since the position error is mainly estimated through the change in the velocity error and the coupling of the attitude errors into the position measurements through the lever arm is weak, the Equation (6.57) can be approximated to [5]:

$$H_k = \begin{bmatrix} -I_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & -I_3 & 0_3 & 0_3 \end{bmatrix} \quad (6.58)$$

The measurement covariance matrix  $R$ , is formed using the error information of the external aid sensors and the measurement noise covariance is assumed to be constant. Thus,

$$R = \begin{bmatrix} R_{11} & 0_3 \\ 0_3 & R_{22} \end{bmatrix} \quad (6.59)$$

Even though the GNSS position measurement is expressed in the Cartesian coordinates, the coupled transformation caused by the acoustic navigation system introduces off-diagonal elements on the upper left part of the measurement covariance matrix  $R_{11}$ .

#### 6.4.4 Estimation Algorithm

A Kalman Filter based estimation algorithm is used to produce error estimates and measurement estimates of the states and integrate the INS-indicated quantities with

the available external measurements. The Kalman Filter is an efficient recursive filter that estimates the state of a linear dynamic system from a series of noisy measurements. It uses the system model to propagate the state and error covariance in time. Then, using the measurement and the corresponding measurement error information, and comparing those with the predicted measurements, the state estimate and state error covariance is updated in an optimal way. Here a brief description of the algorithm will be given, whereas the detailed of the derivation of the equations can be found in [40].

---

*Discrete time state space model*

1  $x_{k+1} = \Phi_k x_k + w_k$

2  $y_k = H_k x_k + v_k$

where  $w_k$  and  $v_k$  are independent, zero-mean, Gaussian noise process of covariance matrices  $Q_k$  and  $R_k$ , respectively.

*Initializaiton:* For  $k = 0$ , set

3  $\hat{x}_0 = E[x_0]$

4  $P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T]$

*Computation:* For  $k = 1, 2, \dots$ , compute:

*State estimate propagation*

5  $\hat{x}_k^- = \Phi_{k-1} \hat{x}_{k-1}^+$

*Error covariance propagation*

6  $P_k^- = \Phi_{k-1} P_{k-1}^+ \Phi_{k-1}^T + Q_{k-1}$

*Kalman gain matrix*

7  $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$

*State estimate update*

8  $\hat{x}_k^+ = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-)$

*Error covariance update*

9  $P_k^+ = (I - K_k H_k) P_k^-$

---

## 6.5 Simulation Results

After the implementation of the integrated navigation algorithm, both underwater vehicle and surface vehicle motions are simulated where parameters of both vehicles are set as the same. The surface vehicle is assumed to contain a GPS-equipment together with a USBL transceiver and the navigation data collected by the surface vehicle is assumed to be transferred to the underwater vehicle without any loss. Both GNSS and USBL measurements are sampled and transferred at 1 Hz.

The underwater vehicle is assumed to contain an IMU together with heading and tilt sensors with an output rate of 100 Hz. and 1 Hz. respectively. IMU bias and random noise parameters are selected as:

$$b_a = b_g = \begin{pmatrix} 0.0001 \\ 0.0001 \\ 0.0001 \end{pmatrix} m/s^2, \quad PSD_a = PSD_g = \begin{pmatrix} 0.0001 \\ 0.0001 \\ 0.0001 \end{pmatrix} (m/s^2)^2/Hz$$

The inertial navigation equations and integrated navigations are implemented on the ECEF coordinate frame; however position solution is transformed into local navigation frame afterwards. The initial positions of both the vehicles are set as the same (38.3244° Latitude, 26.3145° Longitude, 0 m. Depth) which simulates an underwater vehicle release scenario from a surface vessel. Inertial navigation equations are initialized using the GNSS-indicated position, magnetometer- and inclinometer-indicated attitude and zero velocity. After the initialization, the underwater vehicle is released and actuated to dive into the sea for 200 seconds. The surface vessel was also moving during the course of the underwater vehicle. The true position, velocity and attitude data of the underwater vehicle in local coordinate frame can be seen in Figure 6.12, Figure 6.13, and Figure 6.14 respectively.. The INS-indicated position, velocity and attitude data of the underwater vehicle in local coordinate frame can be seen in Figure 6.15, Figure 6.16, and Figure 6.17 respectively. The GNSS-indicated data of the surface vessel in local coordinate frame can be seen in Figure 6.18. USBL-indicated bearing angles and range data from the surface vessel to underwater vehicle in body frame coordinates of USBL device is seen in Figure 6.19. True and estimated position errors in ECEF coordinate

frame can be seen in Figure 6.20 and Figure 6.21 respectively, true and estimated velocity errors in ECEF coordinate frame can be seen in Figure 6.22 and Figure 6.23 respectively, and true and estimated attitude errors can be seen in Figure 6.24 and Figure 6.25 respectively. Integrated navigation position, velocity and attitude estimation errors can be seen in Figure 6.26, Figure 6.27, and Figure 6.28 respectively. Figure 6.29 illustrates 3D trajectories generated by GNSS-indicated surface vessel position and position output of the integrated navigation on the underwater vehicle in the local navigation frame.

The results show that since the bias and random noise on the inertial sensor outputs are integrated, INS position and velocity error grows with time. Using the system model of the INS together with the external aids, the estimation algorithm can estimate the position, velocity and attitude solution errors of the inertial navigation system and the external aid contribution ensures that the error of the corrected navigation solution does not grow with time. However, since the external aids does not measure the velocity, the velocity estimation error is larger compared to other estimated states. Besides that, the different precision in the pitch and yaw estimation compared to heading estimation is due to the sensor accuracy.

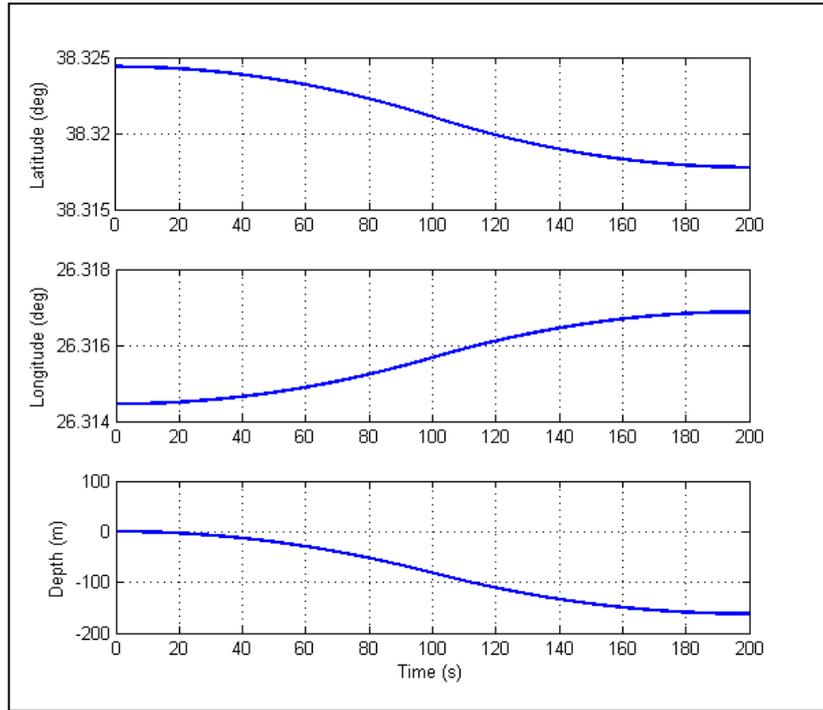


Figure 6.12 True position of underwater vehicle in local navigation coordinate frame

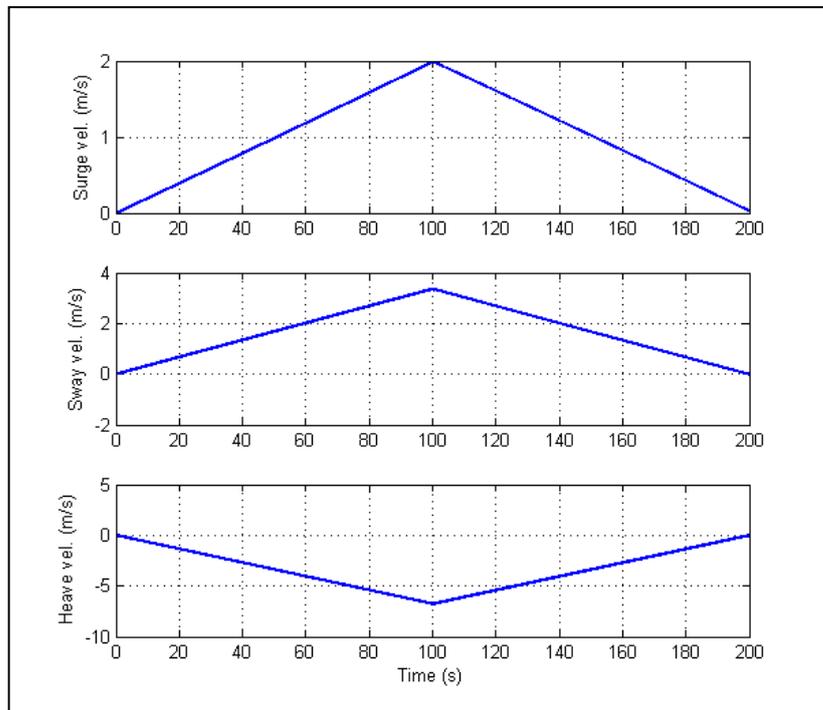


Figure 6.13 True velocity of underwater vehicle in ECEF coordinate frame

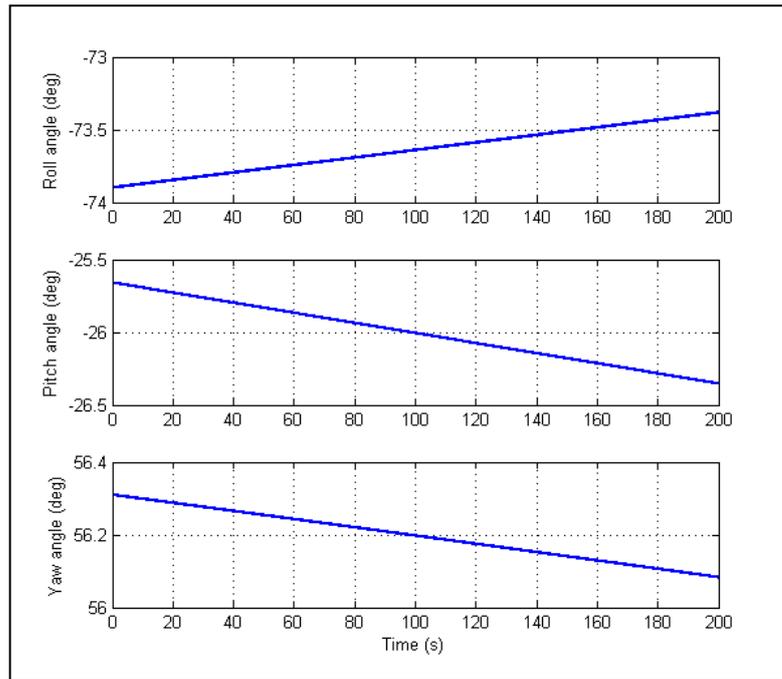


Figure 6.14 True attitude of underwater vehicle

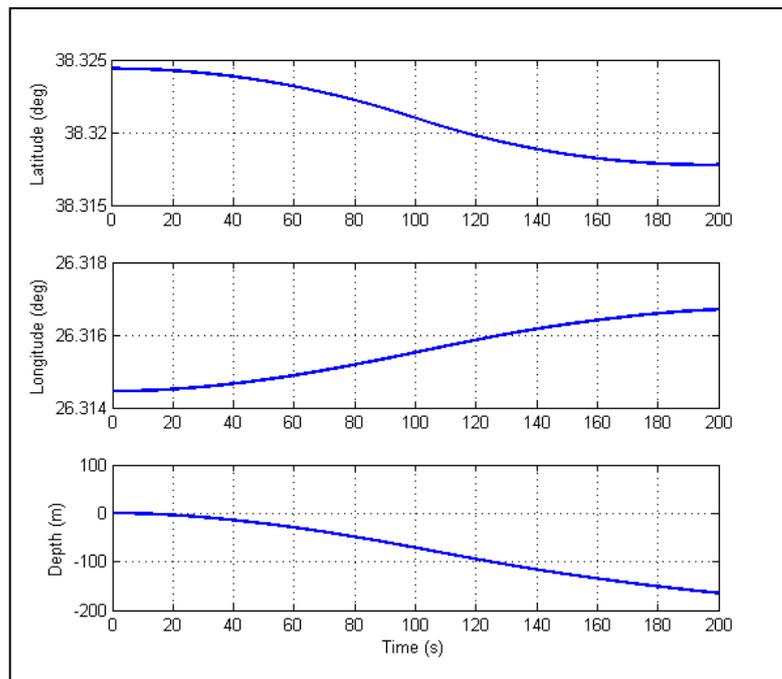


Figure 6.15 INS-indicated position of underwater vehicle in local navigation coordinate frame

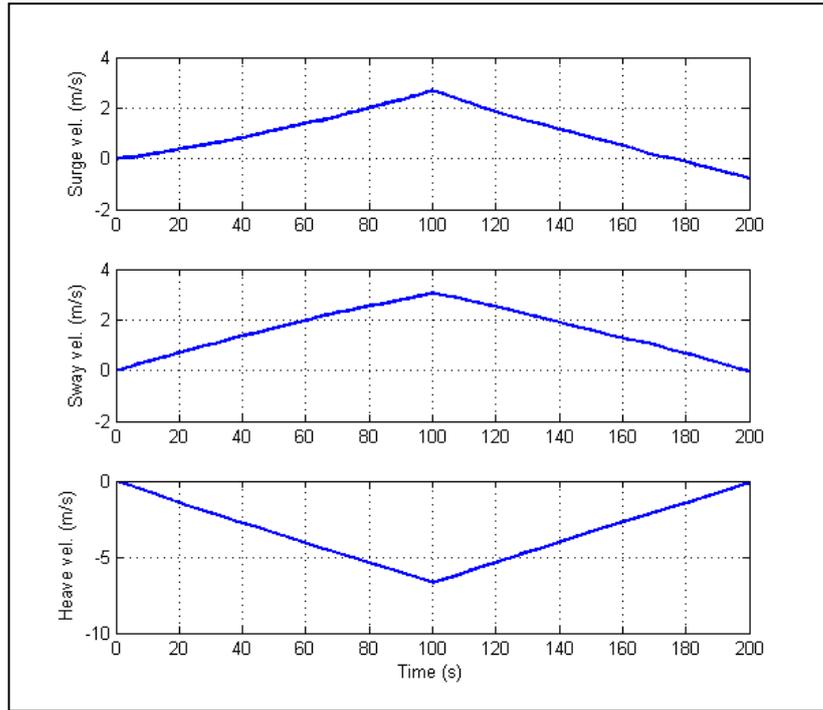


Figure 6.16 INS-indicated velocity of underwater vehicle in ECEF coordinate frame

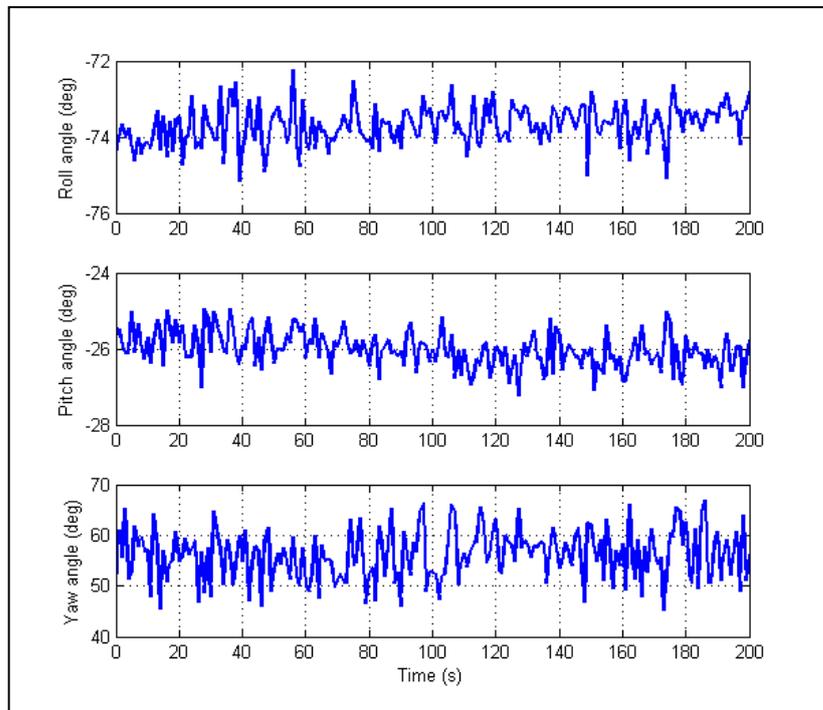


Figure 6.17 INS-indicated attitude of underwater vehicle

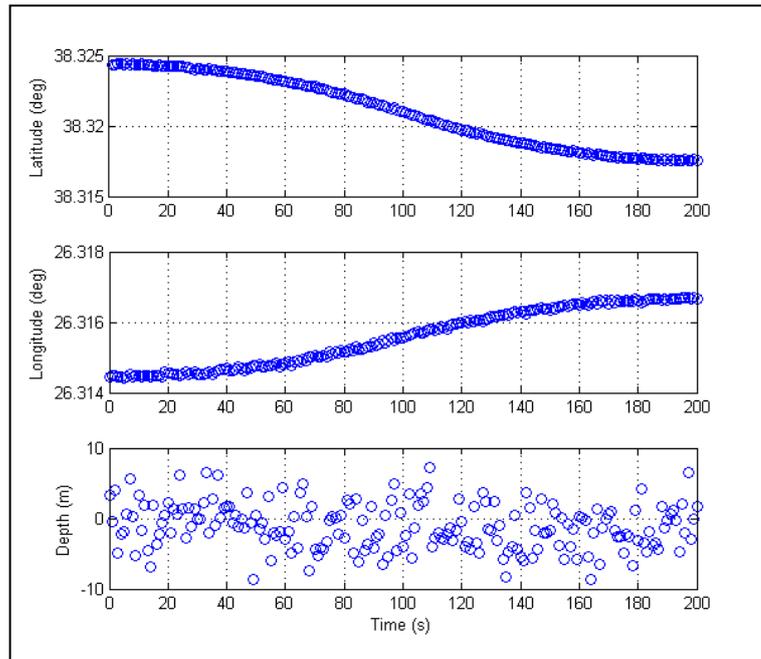


Figure 6.18 GNSS-indicated position of surface vessel in local navigation coordinate frame

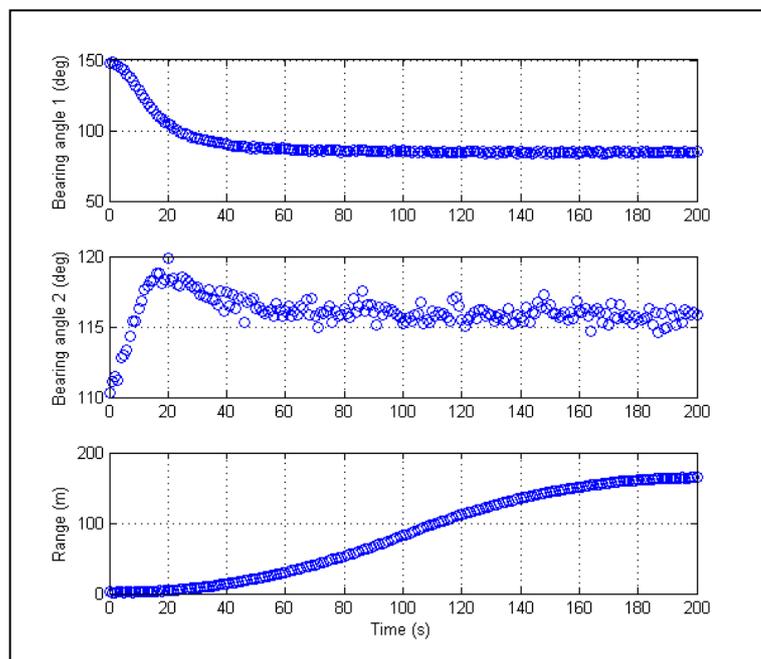


Figure 6.19 USBL-indicated range and bearing from the surface vessel to the underwater vehicle in ECEF coordinate frame

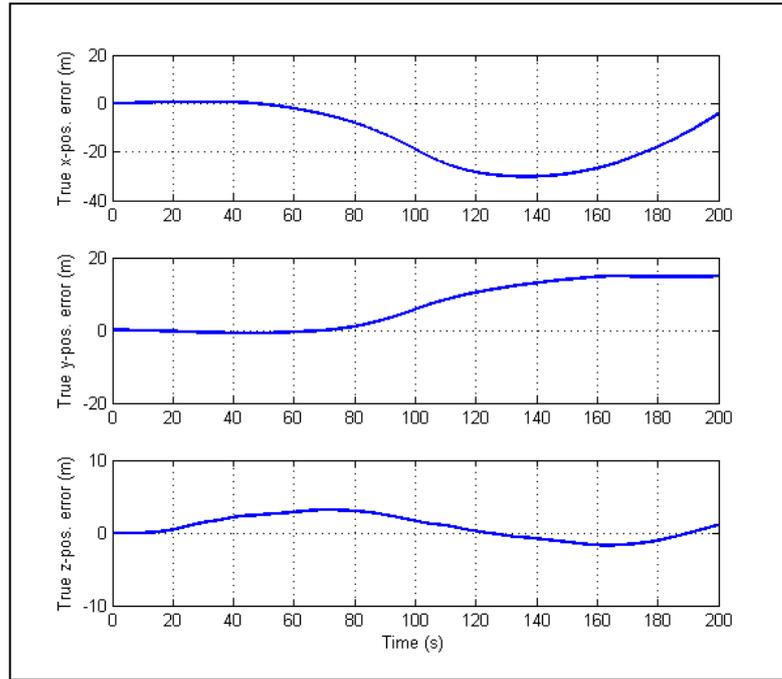


Figure 6.20 INS position error of underwater vehicle in ECEF coordinate frame

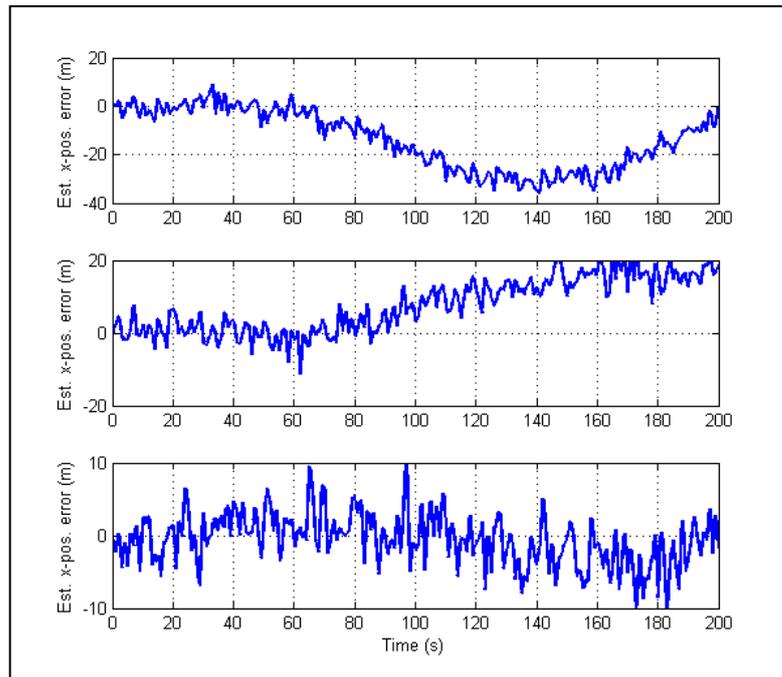


Figure 6.21 Estimated position error of underwater vehicle in ECEF coordinate frame

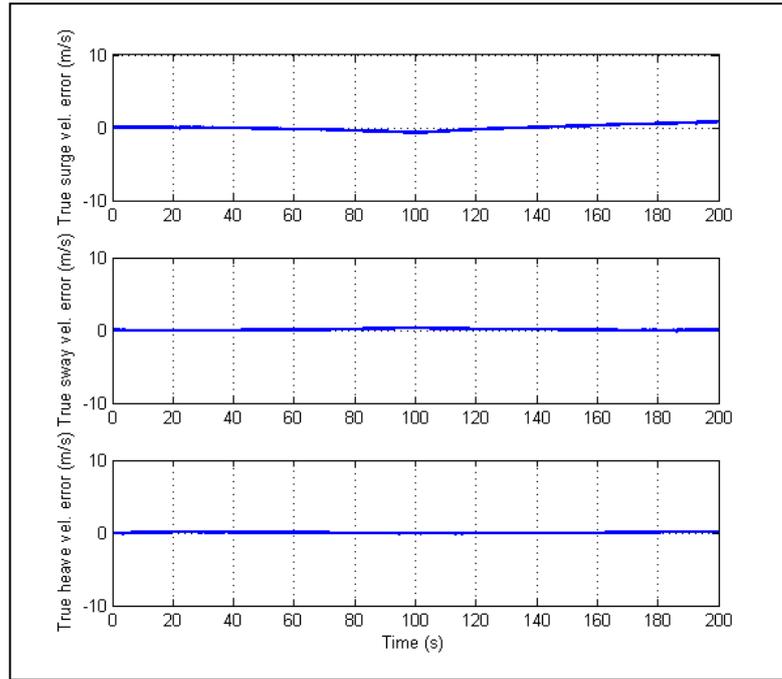


Figure 6.22 INS velocity error of underwater vehicle in ECEF coordinate frame

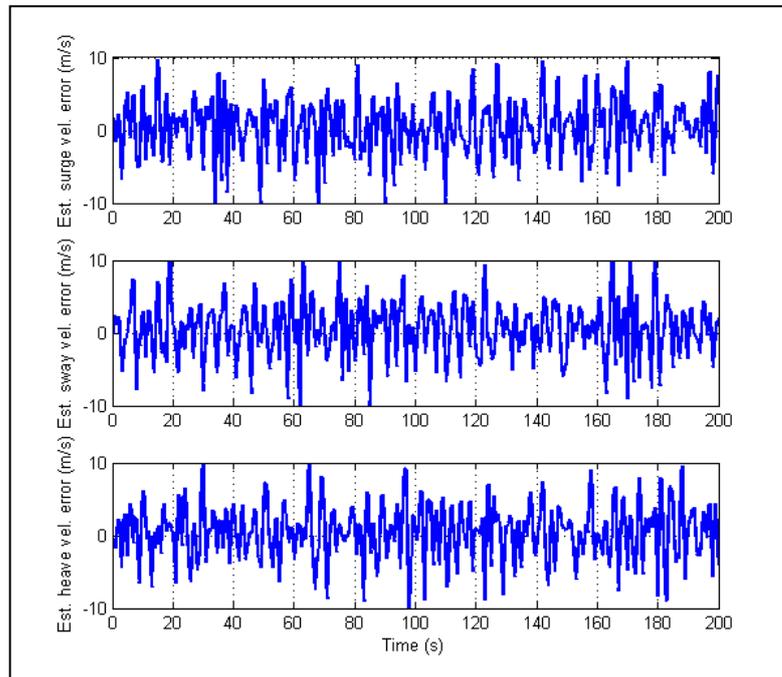


Figure 6.23 Estimated velocity error of underwater vehicle in ECEF coordinate frame

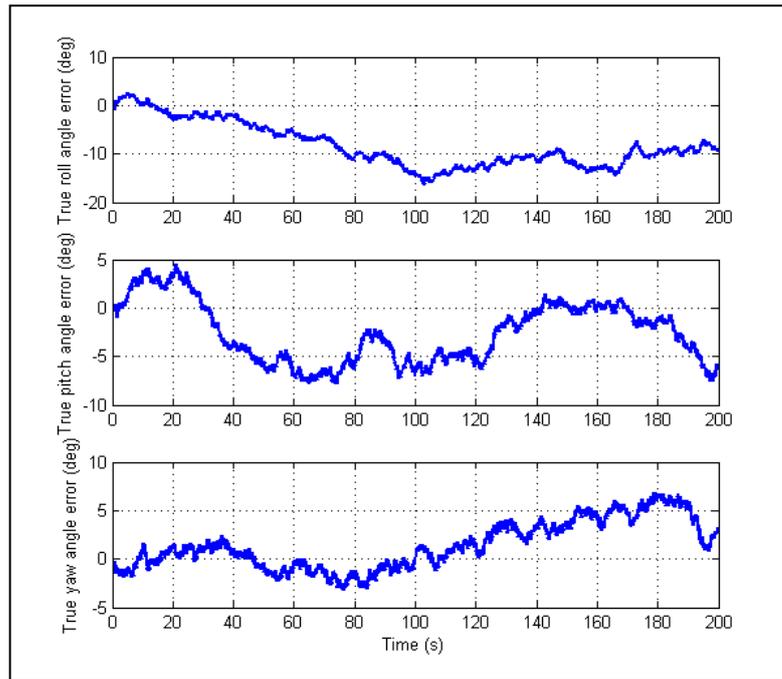


Figure 6.24 INS attitude error of underwater vehicle

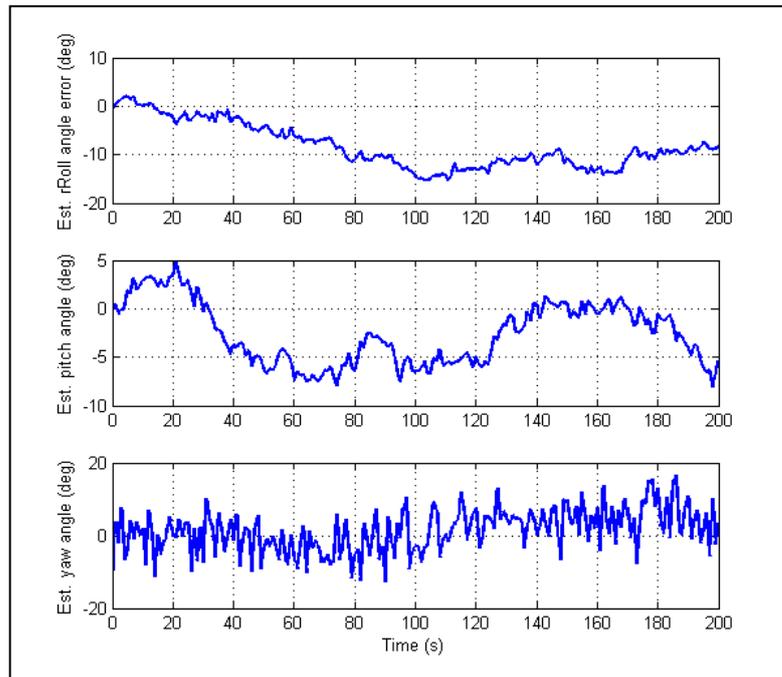


Figure 6.25 Estimated attitude error of underwater vehicle

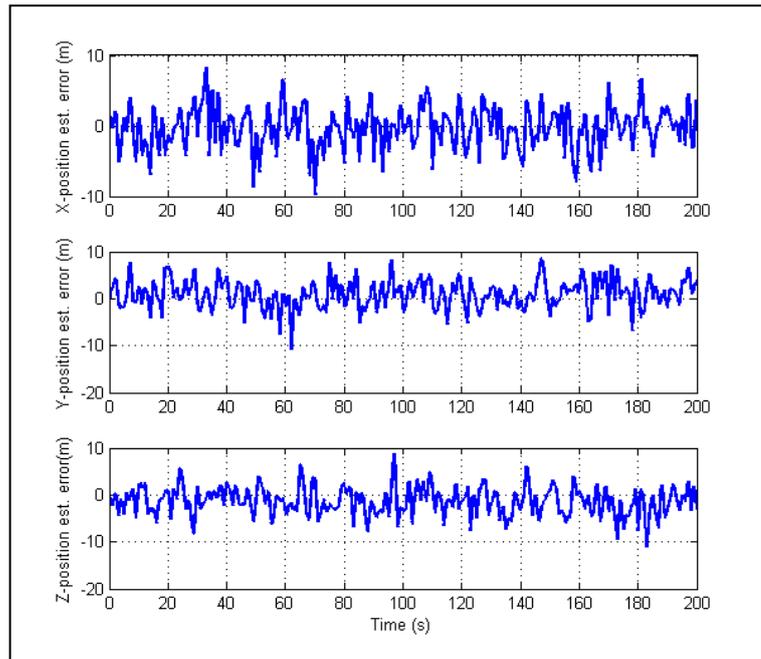


Figure 6.26 Position estimation error of integrated navigation in ECEF coordinate frame

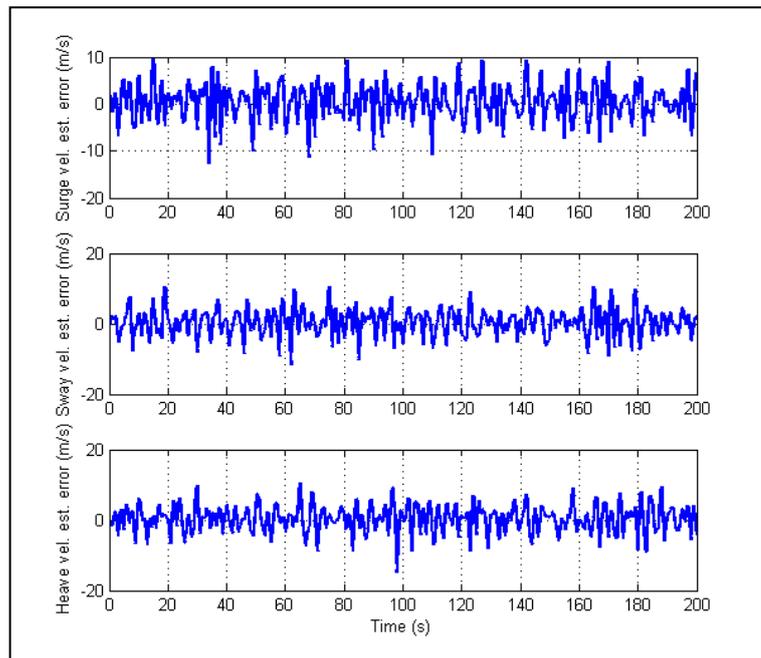


Figure 6.27 Velocity estimation error of integrated navigation in ECEF coordinate frame

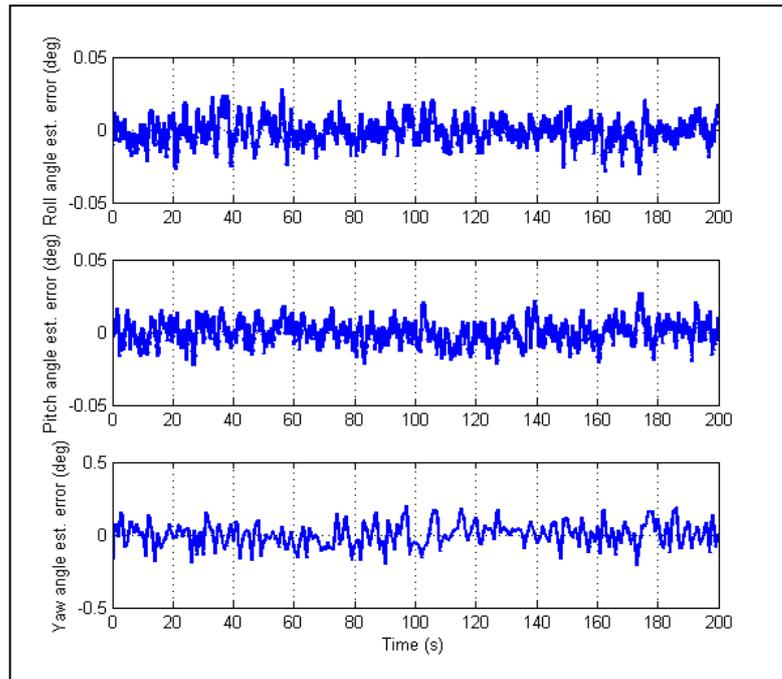


Figure 6.28 Attitude estimation error of integrated navigation

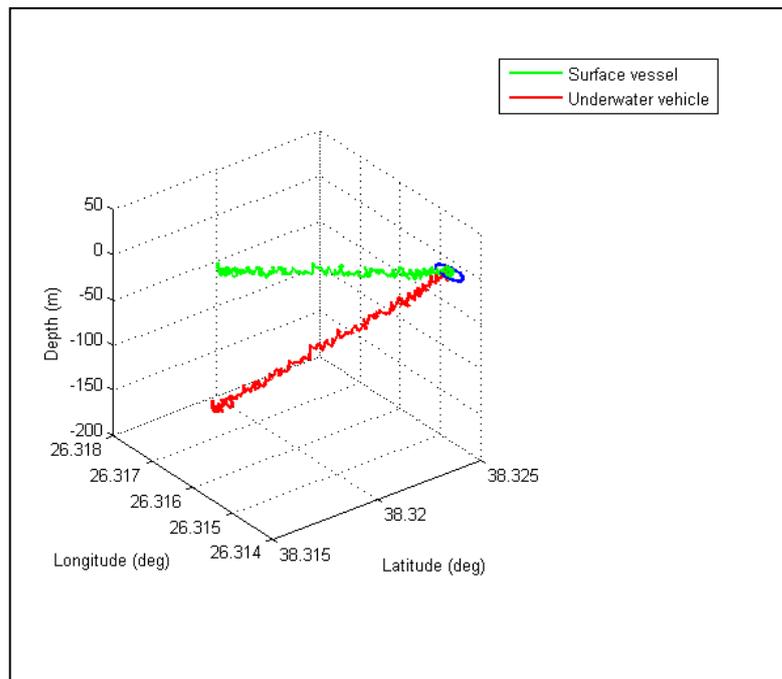


Figure 6.29 Trajectories of the surface vessel and the underwater vehicle in the local navigation frame.

## CHAPTER 7

### CONCLUSION AND FUTURE WORKS

#### 7.1 Conclusion

In this study, navigation and path planning of an UUV is focused. First rigid-body and hydrodynamic parameters of ULISAR UUV is calculated using the strip theory. Using those parameters, a detailed 6-degree-of-freedom non-linear mathematical model, including hydrodynamic and hydrostatic terms, is obtained. To apply linear control techniques, first, the model is linearized about an operating point, and then LQR optimal control method is implemented to control the forward and heave speed as well as pitch and yaw rotations. On top of low-level controller governing the basic motion, waypoint guidance by line-of-sight algorithm is implemented to drive the vehicle using the given sequence of waypoints while providing smooth turns during the waypoint update. The path planning problem is investigated in two parts, firstly, to find the optimal permutation of the given waypoints the Travelling Salesman Problem is solved using genetic algorithm, and secondly, a Rapidly-exploring Random Trees based motion planner algorithm is designed to obtain collision-free trajectories while visiting the given waypoints. At the end, some navigation techniques applicable to underwater operations are studied. An integrated navigation system is proposed, making use of the measurements of INS and attitude sensors on the underwater vehicle as well as the GNSS and acoustic navigation system installed on a surface vessel which collaborates with the UUV. The obtained mathematical model, LQR controller, way-point guidance algorithm, TSP solving genetic algorithm, RRT motion planner and integrated navigation system is verified by numerous simulation results.

## 7.2 Future Works

Since the rigid-body parameters and hydrodynamic parameters of the UUV is calculated only in a theoretical way in this study, to form a more realistic mathematical model, various experiments can be done on the real UUV, and theoretically derived parameters can be compared by the experimentally identified parameters. The environmental disturbances, such as currents, can also be included in the model.

The path planning algorithms can be studied further to provide time-minimizing or energy-minimizing trajectories while avoiding the obstacles. The integrated navigation solution can be integrated into the path planning algorithm to obtain a more complete planner. Collaborative path planners can be designed for multiple UUV missions.

More sophisticated integrated navigation algorithms can be studied, some of which uses the raw GNSS range measurements instead of the outputs of the GNSS processor, and the performance of those algorithms can be compared by simulation. Various other external aids such as underwater landmarks or depth-image matching techniques can be investigated to decrease the cost of expensive acoustic navigation systems.

Finally the controller, motion/path planner and integrated navigation algorithms can be implemented on a real UUV, and by making pool and open sea experiments, the theoretical solutions can be compared with the experimental data.

## REFERENCES

- [1] Blidberg, D. (2001). The Development of Autonomous Underwater Vehicles (AUV); A Brief Summary. *The Development of Autonomous Underwater Vehicles (AUV); A Brief Summary*. Seoul: IEEE.
- [2] Isiyel, K. (2007). *Autopilot Design and Guidance Control of ULISAR UUV (Unmanned Underwater Vehicle)*. Ankara: M.Sc.Thesis in METU Electrical and Electronics Engineering.
- [3] Leblebicioğlu, K. (2010). ULİSAR: Bir İnsansız Su Altı Aracı. *Savunma Sanayii Gündemi Dergisi*, 12, 40-46.
- [4] Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. Chichester, United Kingdom: John Wiley & Sons.
- [5] Groves, P. D. (2008). *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Boston: Artech House.
- [6] Antonelli, G. (2006). *Underwater Robots - Motion and Force Control of Vehicle-Manipulator Systems*. Berlin, Germany: Springer.
- [7] Prestero, T. (2001). *Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle*. Massachusetts Institute Of Technology and the Woods Hole Oceanographic Institution.
- [8] Blevins, R. D. (1979). *Formulas for Natural Frequency and Mode Shape*. Florida: Kreiger Publisher.
- [9] Newman, J. N. (1997). *Marine Hydrodynamics*. Massachusetts: MIT Press.
- [10] Cevheri, N. (2009). *Computer Aided Engineering of an Unmanned Underwater Vehicle*. Middle East Technical University.
- [11] Bachmayer L., W. L. (2000). An Accurate Four-Quadrant Nonlinear Dynamical Model for Marine Thrusters: Theory and Experimental Validation. *IEEE Journal of Oceanic Engineering*, 25:146–159.

- [12] Ogata, K. (2002). *Modern Control Engineering*. New Jersey: Prentice-Hall, Inc.
- [13] Kirk, D. E. (2004). *Optimal Control Theory: An Introduction*. New York: Dover Publications, Inc.
- [14] Kalman, R. E. (1960). Contributions to the Theory of Optimal Control. *Bol. Soc. Mat. Mex.*, 102-119.
- [15] Rosenbrock, M. (1970). *State-Space and Multivariable Theory*. John Wiley.
- [16] Naeem, W. ; Sutton, R. ; Ahmad, S. M. ; Burns, R. S. . (2003). A Review of Guidance Laws. *The Journal of Navigation*, 56, 15-29.
- [17] Healey, A. J.; Lienard, D. (1993). Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles. *IEEE Journal of Oceanic Engineering*, 18, 327-339.
- [18] Hong-jian, W., Xin-qian, B., Jie, Z., Fu-guang, D., & Guo-qing, X. (2004). A GA path planner based on domain knowledge for AUV. *MTTS/IEEE TECHNO-OCEAN '04*, (pp. pp. 1570- 1573).
- [19] Qiaorong, Z. (. (2006). A hierarchical global path planning approach for AUV based on genetic algorithm. *IEEE International Conference on Mechatronics and Automation* (pp. 1745-50). Zhengzhou: IEEE .
- [20] Arinaga, S.; Nakajima, S.; Okabe, H.; Ono, A.; Kanayama, Y. (1996.). A Motion Planning Method for an AUV. *Proceedings of the IEEE Symposium on Autonomous Underwater Vehicle Technology*, (pp. 477–484).
- [21] Shuping, H., Yanli, C., Yanchao, Z., & Zheping, Y. (2009). C-space geometric theory applicated in AUV three-dimensional global path planning . *International Conference on Mechatronics and Automation, 2009. ICMA 2009.* (pp. 2089 - 2094). Harbin: IEEE.
- [22] Carroll, K., McClaran, S., Nelson, E., Barnett, D., Friesen, D., & William, G. (1992). UV path planning: an A\* approach to path planning with consideration of variable vehicle speeds and multiple, overlapping, time-dependent exclusion zones. *Autonomous Underwater Vehicle Technology, 1992. AUV '92.*, (pp. pp.79-84).

- [23] Ergezer, H.; Leblebicioğlu, K. (2011). Planning unmanned aerial vehicle's path for maximum information collection using evolutionary algorithms. *Preprints of the 18<sup>th</sup> IFAC World Congress*, Milano.
- [24] Stentz, A. (1994). Optimal and efficient path planning for partially-known environments. *Proceedings of International Conference on Robotics and Automation, 1994* (pp. 3310-3317 vol.4). Pittsburgh: IEEE.
- [25] Wiener, J. N. (2009). Planning paths to multiple targets: memory involvement and planning heuristics in spatial problem solving. *Psychological Research* 77, 644-658 .
- [26] Alvarez, A. (2004). Evolutionary path planning for autonomous underwater vehicles in a variable ocean. *IEEE Journal of Oceanic Engineering*, 29, 418-429 .
- [27] Goldberg, D., & Lingle, R. (1985). Alleles, Loci and the Traveling Salesman Problem. *First International Conference on Genetic Algorithms and Their Applications* (pp. 154-159). Hillsdale, NJ: Lawrence Erlbaum Associates.
- [28] Canny J., D. B. (1988). On the complexity of kinodynamic planning. *Proceedings of the 29th Symposium on the Foundation of Computer Science* (pp. 306-316). New York: IEEE.
- [29] Latombe, J. (1991). *Robot Motion Planning*. Boston, MA: Kluwer Academic.
- [30] Kavraki, L., Svestka, P., Latombe, J.-C., & Overmars, M. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 566-580 .
- [31] Lozano-Perez, T. (1983). Spatial Planning: A Configuration Space Approach. *IEEE Transactions on Computers*, 108-120.
- [32] LaValle, S.M.; Kuffner, J.J. (1999). Randomized Kinodynamic Planning. *IEEE International Conference on Robotics and Automation*, (pp. 473-479).
- [33] Anon. (1997). *Department of Defense World Geodetic System 1984*. National Imagery and Mapping Agency (now NGA), TR8350.2, 3rd ed.: 1997.
- [34] Rigby, P.; Pizarro, O.; Williams, S.B. . (2006). Towards Geo-Referenced AUV Navigation Through Fusion of USBL and DVL Measurements. *OCEANS 2006*, (pp. 1-6).

- [35] Kinsey, J. C.; Eustice, R. M.; Whitcomb, L. L. (2006). A survey of underwater vehicle navigation: Recent advances and New Challenges. *In Proceedings of the IFAC Conference of Manoeuvring and Control of Marine Craft*. Lisbon.
- [36] Min, Y.; Junyin, H. . (2010). The calibration of the USBL transducer array for Long-range precision underwater positioning. *IEEE 10th International Conference on Signal Processing (ICSP) 2010* (pp. 2357-2360 ). IEEE.
- [37] *Hydro INTERNATIONAL*. (2008, May). Retrieved June 2012, from [http://www.hydro-international.com/files/productsurvey\\_v\\_pdfdocument\\_21.pdf](http://www.hydro-international.com/files/productsurvey_v_pdfdocument_21.pdf)
- [38] Caruso, M. (2000). Applications of magnetic sensors for low cost compass systems. *Proceedings of IEEE Position, Location and Navigation Symposium* (pp. 177-184). San Diego, CA: IEEE.
- [39] Akça, T. (2012). *An Adaptive Unscented Kalman Filter for Tightly-Coupled INS/GPS Integration*. Ankara: M.Sc.Thesis in METU Electrical and Electronics Engineering .
- [40] Haykin, S. (2002). *Kalman Filtering and Neural Networks*. New York: John Wiley & Sons, Inc.

## APPENDIX A

### DRAW COEFFICIENT

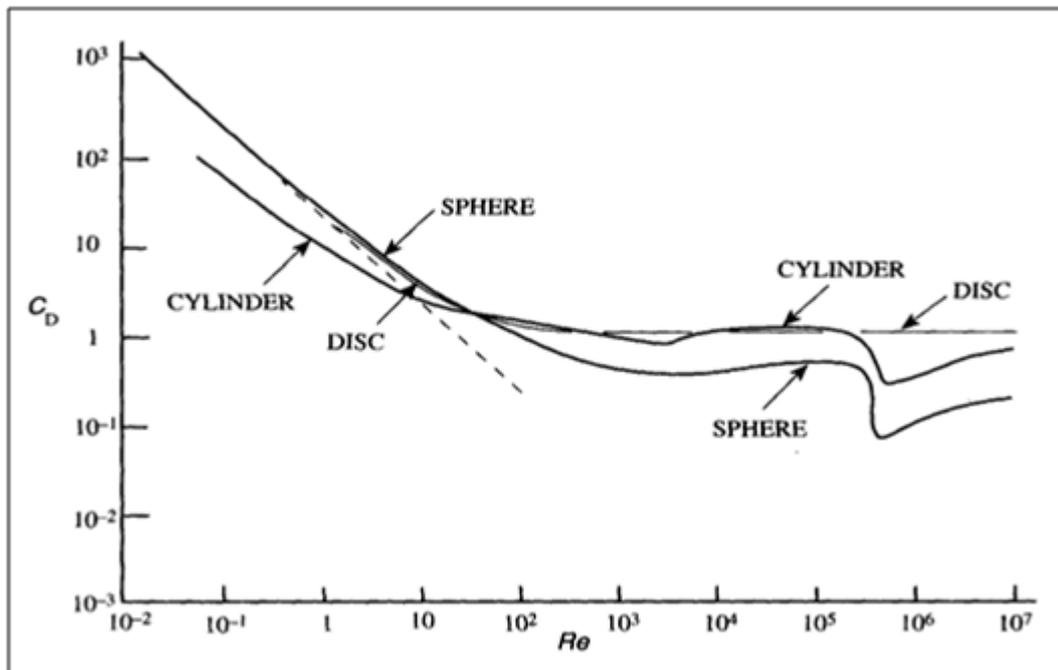


Figure A.1 The drag coefficient as a function of Reynolds number for different geometrical objects

In the drag coefficient function of sphere, the part of which the Reynolds number is between  $10^3$  and  $3 \times 10^3$  denotes the laminar flow, and after  $3 \times 10^3$ , the turbulent flow begins [9].

## APPENDIX B

### GNSS POSITION ERROR

<i>Frequencies</i>	<i>Multipath</i>	<i>Signal</i>	<i>Position Error Standard Deviation (m)</i>		
			<i>Total</i>	<i>Horizontal (Radial)</i>	<i>Vertical</i>
Single	Short-range	BPSK(1)	7.0	3.7	6.0
		BOC <sub>s</sub> (1,1)	6.9	3.7	5.9
		BPSK(10)	6.9	3.7	5.9
	Medium-range	BOC <sub>s</sub> (10,5)	6.9	3.7	5.9
		BPSK(1)	7.2	3.8	6.1
		BOC <sub>s</sub> (1,1)	7.1	3.7	6.0
		BPSK(10)	7.1	3.7	6.0
	Long-range	BOC <sub>s</sub> (10,5)	7.1	3.7	6.0
		BPSK(1)	7.4	3.9	6.3
		BOC <sub>s</sub> (1,1)	7.3	3.9	6.2
		BPSK(10)	6.9	3.7	5.9
		BOC <sub>s</sub> (10,5)	6.9	3.7	5.9
Dual	Short-range	BPSK(1)	2.2	1.2	1.9
		BOC <sub>s</sub> (1,1)	2.0	1.0	1.6
		BPSK(10)	2.0	1.0	1.6
		BOC <sub>s</sub> (10,5)	2.0	1.0	1.6
	Medium-range	BPSK(1)	2.7	1.4	2.3
		BOC <sub>s</sub> (1,1)	2.5	1.3	2.1
		BPSK(10)	2.5	1.3	2.1
		BOC <sub>s</sub> (10,5)	2.5	1.3	2.1
	Long-range	BPSK(1)	3.2	1.7	2.8
		BOC <sub>s</sub> (1,1)	3.0	1.6	2.5
		BPSK(10)	1.9	1.0	1.6
		BOC <sub>s</sub> (10,5)	1.9	1.0	1.6

Table B.1 GNSS position error standard deviation for different settings [5]

# CURRICULUM VITAE

**Full name:** Uğur Doğan GÜL

**Education:**

M.Sc., Electrical and Electronics Engineering Dept., METU (2009-2012)

B.Sc., Electrical and Electronics Engineering Dept., METU (2005-2009)

High School, American Collegiate Institute, İZMİR (2001-2005)

**Conference Presentations:**

- [1] Gül, U.D.; Leblebicioğlu, K. (2011). Otonom Sualtı Aracı Modellemesi, Denetimi ve Hareket Planlama Tasarımı. *Fırat Üniversitesi Elektrik-Elektronik-Bilgisayar Sempozyumu*, Elazığ.
- [2] Gül, U.D.; Leblebicioğlu, K. (2012). Otonom Sualtı Aracı Modellemesi, Denetimi, Hareket ve Görev Planlama Tasarımı. *Sinyal İşleme ve İletişim Uygulamaları (IEEE SİU 2012) Kurultayı*, Muğla.
- [3] Gül, U.D.; Leblebicioğlu, K. (2012). İnsansız Denizaltı Aracı Modellemesi, Denetimi, Hareket ve Görev Planlama Tasarımı. *Savunma Teknolojileri (SAVTEK 2012) Kongresi*, Ankara.
- [4] Gül, U.D.; Leblebicioğlu, K. (2012). Bir İnsansız Sualtı Aracının Denetim, Yol Planlama ve Navigasyonu. *Otomatik Kontrol Türk Milli Komitesi 2012 Ulusal Toplantısı (TOK 2012)*, Niğde.