Steffen Østensjø Kørte

Guidance & Control Strategies for UUVs

Master Thesis,
June, 2011

Norwegian University of Science and Technology
Department of Marine Technology

NTNU – Trondheim
Norwegian University of
Science and Technology

Abstract

Focus on safety and effectiveness in the oil and gas industry has increased the need for advanced control system for underwater vehicles. For remotely operated vehicles (ROVs) effectiveness when conducting deep sea operation is extremely critical both concerning safety and cost. For AUVs conducting survey missions, energy and collision avoidance are critical factors.

A control system for ROV Minerva is being developed through the AUR-Lab at the Institute of Marine Technology at NTNU. Several guidance features have been investigated, with respect to different mission the ROV should be able to do. A guidance system has been developed, where basic functions such as a DP system with station keeping and trajectory tracking was the focus of the project thesis of the author. More advanced functionality with respect to path following has been developed, where focus have been set on different strategies to apply when the ROV is following a path. Simulations based on different strategies for following a lawnmower pattern have been simulated and evaluated with respect to time, energy and control objective. Full scale experiments with the ROV Minerva show that especially the forward speed versus arc radius is an important issue. The results show that the ROV is able to follow a path made out of straight lines and circular arcs, and particularly lawnmower patterns.

For an underwater vehicle to be able to operate autonomously, stationary and dynamic obstacles have to be taken into account. A collision avoidance system based on local collision avoidance algorithms has been implemented, where focus has been on reactivity with respect to unmapped obstacles. The collision avoidance system has been through initial full scale tests with ROV Minerva, and the result are promising.

Preface

The work in this thesis has mainly been done during the timespan between January 2011 and June 2011, but builds upon the work done during the Fall of 2010 in my project thesis. The thesis has been written based on work done during participation in AUR-Lab at the Institute of Marine Technology at the Norwegian University of Science and Technology in Trondheim, Norway. My advisor has been Professor Asgeir J. Sørensen, and my co-advisors hav been Dr Marting Ludvigsen, Fredrik Dukan and Daniel de Almeida Fernandes.

All the work has been done with the intention of full scale implementation for a control system for ROV Minerva, which is a NTNU owned remotely operated vehicle. This has made the thesis not only a theoretical presentation, but also many practical issues have had to be considered. This has made the process very challenging, but also very fruitful and educational. Especially the cruises on the Trondheimsfjord with the NTNU research vessel RV Gunnerus have been very exiting.

The contribution from this thesis is highly dependent on all the work done by fellow students in the AUR-Lab, the motivation gotten from helping each other to reach a goal has been very important. The work done by Marianne Kirkeby during the fall of 2010 on control of ROV Minerva, has been crucial for me to focus on the guidance system. It has been a pleasure to get to know the Italian students Laura Standardi and Mauro Candeloro. For most of the Spring of 2011, the only two students contributing to AUR-Lab has been Phat Truong and myself. I would like to thank Mr. Truong for keeping long days, both in the office and during cruises, entertaining.

The cruises have been most enjoyable, mostly due to the nice atmosphere on the RV Gunnerus. The crew of RV Gunnerus has been great, and has arranged everything such that we have been able to focus entirely on the work with ROV Minerva.

I would like to thank Dr Martin Ludvigsen for sharing his experience of all aspects concerning the ROV Minerva. I would like to thank Daniel de Almeida Fernandes for our discussions and work on guidance strategies, as a sparring partner. I would like to thank Fredrik Dukan for always having an open door, and helping me with everything. He has also been in charge of the integration and the complete control system in general, which has been much appreciated.

Finally I would like to thank my advisor, Professor Asgeir J Sørensen. He has during the last year been equally enthusiastic each time we met, which has given great inspiration to achieve good results. He is always accommodating, and his door is always open.

Trondheim, June 14, 2011

Contents

Abstract **I**

Preface **III**

1 Introduction **1**
1.1 Background and Motivation . . . 1
1.1.1 Underwater Vehicles . . . 1
1.1.2 Unmanned Underwater Vehicles . . . 1
1.1.3 ROV Minerva . . . 3
1.1.4 Marine Control Systems . . . 5
1.2 Previous Work . . . 6
1.2.1 Control Systems for UUVs . . . 6
1.2.2 ROV Minerva . . . 7
1.2.3 Guidance . . . 7
1.2.4 Collision Avoidance . . . 7
1.3 Contributions . . . 8
1.3.1 Publication . . . 8
1.4 Outline of Thesis . . . 8

2 Preliminaries in Motion Control **11**
2.1 Kinematics . . . 11
2.1.1 Reference Frames . . . 11
2.1.2 Transformation Between NED and Body . . . 13
2.2 Guidance Terminology . . . 13
2.2.1 Dynamic Positioning System . . . 13
2.2.2 Actuation . . . 14
2.3 Motion Control Scenarios . . . 14
2.3.1 Maneuvering . . . 15
2.3.2 Alternative Classification of Motion Control Scenarios . . . 16
2.3.3 UUV Scenarios . . . 16

3 Guidance Concepts **19**
3.1 Guidance Objectives . . . 19
3.1.1 Target Tracking . . . 19
3.1.2 Trajectory Tracking . . . 20
3.2 Path Planning . . . 24
3.3 Path representation . . . 24
3.3.1 Steering for Straight Lines and Circular Arcs . . . 25
3.3.2 Parametrized Paths . . . 29
3.4 Path Following . . . 30

3.4.1 Steering on a Straight Line . . . 30
3.4.2 Steering on a Circle . . . 31
3.4.3 Line of Sight Path Following . . . 32
3.5 Cross-Track Error Regulation for a Fully Actuated ROV . . . 34
3.6 Strategies for Path Following . . . 35
3.6.1 Four Approaches to a Corner . . . 35
3.6.2 Velocity Reference Model for Steps in Position . . . 36
3.6.3 Propeller Characteristics . . . 36

4 Guidance System for ROV Minerva 39
4.1 The Guidance System . . . 39
4.1.1 Dynamic Positioning Mode . . . 40
4.1.2 Path Following Mode . . . 41
4.2 Implementation . . . 42
4.2.1 Human Machine Interface . . . 43
4.3 Simulations of Guidance Strategies . . . 44
4.3.1 Simulation of ① . . . 44
4.3.2 Simulation of ② . . . 46
4.3.3 Simulation of ③ . . . 49
4.3.4 Simulation of ④ . . . 51
4.3.5 Summary of Guidance Strategies . . . 54
4.4 Full Scale Experiments with ROV Minerva . . . 55
4.4.1 Line of Sight Path Following . . . 55
4.4.2 Square Tests . . . 56
4.4.3 Lawnmower Patterns . . . 61
4.4.4 An Alternative Approach to Lawnmower Path Following . . . 65
4.4.5 Mapping of Periphylla periphylla Population in Verrabotn . . . 67
4.5 Discussion . . . 70

5 Collision Avoidance 71
5.1 Obstacles . . . 71
5.2 Underwater Collision Avoidance . . . 71
5.2.1 Obstacle Mapping . . . 71
5.2.2 Supervisory System . . . 72
5.3 Local Collision Avoidance Algorithms . . . 74
5.3.1 The Dynamic Window Approach . . . 74
5.3.2 The Artificial Potential Field Approach . . . 75
5.3.3 The Vector Field Histogram Approach . . . 79
5.4 Global Collision Avoidance Algorithms . . . 82
5.4.1 Shortest Path Algorithms . . . 82
5.4.2 Rapidly-Exploring Random Trees . . . 83
5.4.3 Hybrid Collision Avoidance Algorithm . . . 83

6 Collision Avoidance System for ROV Minerva 85
6.1 The Collision Avoidance System . . . 85
6.1.1 Map Generator and Obstacle representation . . . 86
6.1.2 The Vector Field Histogram Algorithm . . . 86
6.1.3 Reference Models . . . 87
6.1.4 Implementation . . . 88
6.2 Simulation-based Experiments . . . 88
6.2.1 Step Inputs in Heading vs Smooth Heading Reference . . . 88

6.2.2 Selecting the Maximum Sectors Parameter . . . 91
6.2.3 Trap Situation . . . 93
6.2.4 Travelling Between Two Obstacles . . . 94
6.2.5 Narrow Corridors and Several Obstacles . . . 96
6.3 Full Scale Experiments with ROV Minerva . . . 99
6.3.1 Scenario 1 . . . 99
6.3.2 Scenario 2 . . . 101
6.3.3 Scenario 3 . . . 103
6.4 Discussion . . . 105

7 Concluding Remarks **107**
7.1 Conclusion . . . 107
7.2 Recommendations for Further Work . . . 107
7.2.1 Future Work on ROV Minerva . . . 108
7.2.2 Future Work on Guidance . . . 108
7.2.3 Future Work on Collision Avoidance . . . 108

A Complete Results from a Selection of Full Scale Experiments **113**
A.1 Arc Radius of 2 Metres . . . 113
A.2 Lawnmower Pattern with 10 Metre Line Distance . . . 120
A.3 Collision Avoidance From Scenario 1 . . . 123

B Program Listing **127**

List of Figures

1.1 The Hugin AUV (Kongsberg Maritime AS) is a torpedo shaped AUV. 2
1.2 The Jason ROV and the VideoRay Pro 3 ROV. 2
1.3 The Nereus hybrid underwater vehicle. 3
1.4 Overview of thrust configuration. 3
1.5 RV Gunneurs is the vessel from which ROV Minerva is deployed. 4
1.6 Overview of signal flow for ROV Minerva (Figure: Mauro Candeloro). 4
1.7 The structure of a marine control system (Sørensen, 2011). 6
1.8 Overview of a marine control system. 8

2.1 The three reference frames ECI, ECEF and NED, denoted by i, e and n respectively. 12

3.1 Target tracking guidance schemes. 20
3.2 Reference model response with saturating elements. 21
3.3 3rd order reference model response with nonlinear damping. 22
3.4 Response for a step using a velocity reference model. 23
3.5 Geometric outline around a waypoint. 25
3.6 A path representation using the principle of straight lines and circular arcs. . . . 26
3.7 A path based on the steering for straight lines and circular arcs principle for following a lawnmower pattern. 27
3.8 Lawnmower pattern using arcs of different radius. 28
3.9 Lawnmower pattern where the lines between the lines also has to be covered. . 28
3.10 Path representation using parametrized paths. 29
3.11 Horizontal path using parametrized paths. 30
3.12 Path-fixed reference frame and its parameters. 31
3.13 Path-fixed reference frame when steering on a circle. 32
3.14 Line of sight path following guidance scheme. 33
3.15 Different strategies for turning a corner. 35

4.1 Signal flow of the control system for ROV Minerva. 39
4.2 The generic principle of the guidance system for ROV Minerva. 40
4.3 Screen shot of HMI for the guidance system. 43
4.4 Screen shot of HMI for the guidance system. 43
4.5 Simualtion of guidance strategy ① 45
4.6 Cross-track error and surge speed for ① 45
4.7 Thrust commands for ① 45
4.8 Simulation of guidance strategy ② w/ constant forward speed 46
4.9 Cross-track error and surge speed for ② w/ constant forward speed. 46
4.10 Thrust commands for ② w/ constant forward speed 47
4.11 Simulation of guidance strategy ② w/ reduced forward speed on arcs 48
4.12 Cross-track error and surge speed for ② w/ reduced forward speed on arcs. . . . 48
4.13 Thrust commands for ② w/ reduced forward speed on arcs. 48

4.14 Simulation of guidance strategy ③ 49
4.15 Cross-track error and surge speed for ③ 49
4.16 Thrust commands for ③ 50
4.17 Simulation of guidance strategy ④ w/ constant forward speed. 51
4.18 Cross-track error and surge speed for ④ w/ constant forward speed 51
4.19 Thrust commands for ④ w/ constant forward speed 52
4.20 Simulation of guidance strategy ④ w/ reduced forward speed on the arcs. . . . 53
4.21 Cross-track error and surge speed for ④ w/ reduced forward speed on the arcs. . 53
4.22 Thrust commands for ④ w/ reduced forward speed on the arcs. 53
4.23 Line of Sight lawnmower pattern. 55
4.24 Cross-track error for lawnmower pattern. 56
4.25 Square tests with radius of 2 metres. 57
4.26 Cross-track error for square tests with radius of 2 metres. 57
4.27 Surge velocity for square tests with radius of 2 metres. 57
4.28 Square test with radius of 2 metres w/ reduced speed on arcs. 58
4.29 Cross-track error for square test with radius of 2 metres w/ reduced speed on arcs . 58
4.30 Surge velocity for square test with radius of 2 metres w/ reduced speed on arcs . 58
4.31 Square tests with radius of 2 metres. 59
4.32 Cross-track error for square tests with radius of 3 metres. 59
4.33 Surge velocity for square tests with radius of 3 metres. 60
4.34 Square tests with radius of 5 metres. 60
4.35 Cross-track error for square tests with radius of 5 metres. 61
4.36 Surge velocity for square tests with radius of 5 metres. 61
4.37 ROV Minerva following a lawnmower pattern with line distance of 10 metres. . 61
4.38 Cross-track error for lawnmower pattern with line distance of 10 metres. 62
4.39 Velocities for lawnmower pattern with line distance of 10 metres. 62
4.40 Heading for lawnmower pattern with line distance of 10 metres. 63
4.41 ROV Minerva following a lawnmower pattern with line distance of 4 metres. . 64
4.42 Cross-track error for lawnmower pattern with line distance of 4 metres. 64
4.43 Velocities for lawnmower pattern with line distance of 4 metres. 65
4.44 Heading for lawnmower pattern with line distance of 4 metres. 65
4.45 Full scale experiment where ROV Minerva takes the turn outside the path. . . . 66
4.46 Cross-track error while taking turn outside path. 67
4.47 A Periphylla periphylla jellyfish. 67
4.48 Full scale experiment where ROV Minerva is used for mapping of jellyfish. . . 68
4.49 The position and orientation of the ROV while mapping Periphylla periphylla. . 69
4.50 Cross-track error while following a line. 69

5.1 The vehicle with its critical parameters concerning collision avoidance. 72
5.2 Critical situations for local collision avoidance algorithms. 73
5.3 An artificial potential field. 77
5.4 The behaviour of the potential field near an obstacle. 78
5.5 The behaviour of the potential field near the target. 79
5.6 An example of the Vector Field Histogram approach. 80
5.7 Polar obstacle density histogram and its smoothed counterpart. 81
5.8 Hybrid collision avoidance hierarchy. 83

6.1 The collision avoidance system placed in the guidance system. 85
6.2 The signal flow of the collision avoidance system. 86
6.3 Horizontal view of the ROV moving around an obstacle. 89
6.4 Heading for the ROV while moving around an obstacle. 89

6.5 Horizontal view of the ROV moving around an obstacle, w/ smooth heading. . . 90
6.6 Heading for the ROV while moving around an obstacle, w/ smooth heading. . . 90
6.7 Horizontal view of the ROV while moving around an obstacle, increased distance. 91
6.8 Heading for the ROV while moving around an obstacle with, increased distance. 92
6.9 The ROV in a trap situation. 92
6.10 Heading for the ROV while in a trap situation. 93
6.11 Horizontal view of the ROV moving through an opening between two obstacles. 94
6.12 The heading of the ROV while travelling between two obstacles. 95
6.13 The ROV travelling through narrow corridors. 95
6.14 The heading while travelling through narrow corridors. 96
6.15 The ROV travelling through a canyon-like environment. 97
6.16 Heading while travelling through a canyon-like environment. 98
6.17 Full scale experiment where ROV Minerva is travelling around an obstacle. . . 99
6.18 Heading while moving around an obstacle. 100
6.19 Thruster RPM while moving around an obstacle. 100
6.20 Full scale experiment where the ROV travels between closely spaced obstacles. 101
6.21 Heading while travelling between closely spaced obstacles. 102
6.22 Thruster RPM while travelling between closely spaced obstacles 102
6.23 The ROV is guided between two "islands" by the collision avoidance system. . 103
6.24 The heading while the ROV is travelling between two "islands". 104
6.25 The thrust inputs while the ROV is travelling between two "islands". 104

A.1 Full scale square test horizontal view. 113
A.2 Cross-track error for square test. 114
A.3 Velocities for square test. 114
A.4 Heading for square test. 115
A.5 Position plots for square test. 115
A.6 Thrust inputs for square test. 116
A.7 Full scale square test experiment with reduced forward speed. 117
A.8 Cross-track error for square with reduced forward speed. 117
A.9 Velocities for square test with reduced forward speed. 118
A.10 Heading for square test with reduced forward speed. 118
A.11 Position for square test with reduced forward speed. 119
A.12 Thrust inputs for square test with reduced forward speed. 119
A.13 ROV Minerva following a lawnmower pattern with line distance of 10 metres. . 120
A.14 Cross-track error for following a lawnmower pattern. 120
A.15 Velocities while following a lawnmower pattern. 121
A.16 Heading while following a lawnmower pattern. 121
A.17 Position plots while following a lawnmower pattern. 122
A.18 Thrust inputs while following a lawnmower pattern. 122
A.19 Full scale collision avoidance experiment 123
A.20 Heading while moving around an obstacle. 123
A.21 Thruster RPM while moving around an obstacle. 124
A.22 Position of the ROV while moving around an obstacle. 124
A.23 Velocities while moving around an obstacle. 125

Chapter 1

Introduction

This thesis is part of the Applied Underwater Research Laboratory (AUR-Lab) which is an initiative at the Norwegian University of Science and Technology (NTNU). A control system for the ROV Minerva is being developed, and this thesis' aim is to inspect and apply different strategies for guidance and control purposes. The work done in this thesis is an extension of the work done in the project thesis *Guidance and Control of a Remotely Operated Vehicle (ROV)* (Kørte, 2010). Emphasis has been put into improving the previous work, and inspect possibilities for using more efficient, accurate or sophisticated strategies to replace the strategies used so far.

1.1 Background and Motivation

The last 50 years an increased interest for the deep seas had been developed. This is mostly due to the oil and gas industry, and the discovery of big oil and gas reservoirs beneath the seabed. In the start oil and gas was mostly exploited at shallow depths, where divers were used to fulfil the tasks at the seabed. During the last couple of decades the industry has moved to increased depths, in addition the health issues concerned with deep sea diving has made using divers less common. This has made sophisticated underwater vehicles a crucial asset for the oil and gas industry.

1.1.1 Underwater Vehicles

An underwater vehicle is defined in Fossen (2010) as a *small vehicle that is capable of propelling itself beneath the water surface as well as on the water's surface.* The first submarine was designed in 1578 by Bourne, although the first one to build one was Van Drebble in 1620. The first underwater vehicle to be used in Naval operations was the "Turtle" built by the Bushnell brothers, David and Ezra in 1776 (Roberts, 2008). It could also be noted that one of the first efforts to design a underwater vehicle was, according to Antonelli (2006), Leonardo Da Vinci in his work *Codice Atlantico* (1480-1518).

1.1.2 Unmanned Underwater Vehicles

Initially, unmanned underwater vehicles (UUV) were torpedoes. The first self-propelled torpedo was designed by Whithead in 1868. UUVs were not taken into commercial use until the oil and gas industry recognized its utility in deep sea oil and gas operations. The term UUVs is most often used to describe both autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs).

Autonomous Underwater Vehicle (AUV)

An AUV is a free-swimming unmanned underwater vehicle with its own energy supply (Breivik & Fossen, 2008). It is often torpedo shaped and is commonly used for mapping and survey tasks for scientific, military and commercial purposes. The first AUV was built in the 1970s and was put into commercial use in the 1990s. The HUGIN series AUVs are on of the more successful AUVs in the market today where HUGIN 3000 for instance has 60 hours endurance at 4 knots speed. A typical HUGIN AUV can be seen in Figure 1.1.

Figure 1.1: The Hugin AUV (Kongsberg Maritime AS) is a torpedo shaped AUV.

(a) The Jason ROV (Woods Hole Oceanographic Institute).

(b) The VideoRay Pro 3 ROV (VideoRay LLC).

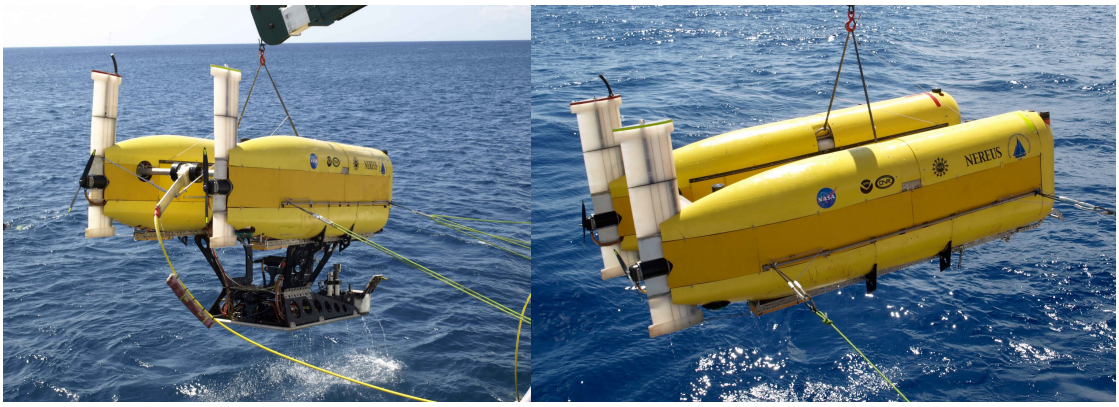
Figure 1.2: The Jason ROV is a work class ROV while the VideoRay Pro 3 ROV is a observation class ROV.

Remotely Operated Underwater Vehicle (ROV)

A ROV is an unmanned underwater vehicle that is connected to a surface vehicle through an umbilical cable that provides power and telemetry to the ROV. The umbilical cable or tether represents the main difference between AUVs and ROVs in terms of terminology. Another difference is that a ROV is often box shaped, and is equipped with different tools depending on the mission, with a number of cameras and lights such that operators can follow the ROV through real-time video is standard equipment. The first ROV was built in the 1950s, but ROVs were not taken into commercial use until the 1980s, mainly in the oil and gas industry. ROVs are widely used to carry out inspection and intervention tasks on subsea installations (Breivik & Fossen, 2008). A work class ROV is a "workhorse" equipped with manipulators and is easily equipped with widely varying tools to do intervention missions such as the Jason ROV in Figure 1.2a. A observation class ROV is normally smaller and is equipped with cameras for observation mostly such as the VideoRay Pro 3 ROV in Figure 1.2b.

Hybrid Underwater Vehicle

A new generation of unmanned underwater vehicles are being developed. These vehicles are designed such that they can operate as both AUVs and ROVs. Different approaches has been made, were Jakuba et al. (2007) describes the Nereus which can be configured as either a AUV or a ROV. Another approach is to have an underwater vehicle that can connect to an underwater docking station to power up, and also connect to an umbilical in order to be able to operate the vehicle as a ROV (Chardard & Copros, 2002).



(a) The Nereus in ROV mode (Woods Hole Oceanographic Institute).

(b) The Nereus in AUV mode (Woods Hole Oceanographic Institute).

Figure 1.3: The Nereus hybrid underwater vehicle is developed at the Woods Hole Oceanographic Institute.

1.1.3 ROV Minerva

Minerva is a SUB-fighter 7500 ROV and is designed by Sperre AS in 2003 for NTNU. The vehicle is able to operate at depths down to approximately 600 metres due to a umbilical. The ROV Minerva is being used for different purposes including biological research, development of new research technology, archaeological surveys etc. The ROV is a multi-purpose vehicle which can be equipped with different tool packages depending on the objective of the deployment. The systems needed to operate Minerva are fitted into a moveable container. The ROV is for most missions deployed from RV Gunnerus, which is a NTNU research vessel. The ROV is controlled manually by a joystick console, which also includes auto-depth and heading. A specification overview of ROV Minerva is given in Table 1.1.

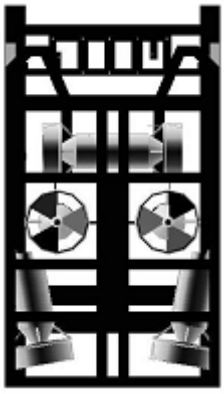


Figure 1.4: Overview of thrust configuration.

Thrust Configuration

Minerva is equipped with 5 thrusters, two longitudinal thrusters, one lateral thruster and two vertical thrusters. The maximum rotational speed of the thrusters are 1450 rounds per minute (RPM) and a capacity of 2000 W. The thrust configuration is illustrated in Figure 1.4, where it

can be seen that the longitudinal thruster are mounted with an angle. This angle is 10 degrees for the port thruster and -10 degrees for the starboard thruster. Each thruster has a $300 - 340$ N capacity in open water.

(a) The ROV Minerva (b) RV Gunnerus (foto: Fredrik Skoglund)

Figure 1.5: RV Gunneurs is the vessel from which ROV Minerva is deployed.

Sensors

Position of the ROV is obtained through the high precision acoustic reference system (HiPaP) Kongsberg HiPaP 500, which is the input to the navigation system NaviPac (Eiva) which outputs the ROVs position in UTM coordinates. Orientation and yaw-rate is measured by compass and gyro, depth by a pressure gauge while velocity and altitude is measured by a Doppler Velocity Log (DVL). Depth is also measured by sonar. Figure 1.6 shows an overview of the main components and signals used for the control system. As can be seen, NaviPac also takes in measurements from the support vessel RV Gunnerus in order to calculate the ROVs position.

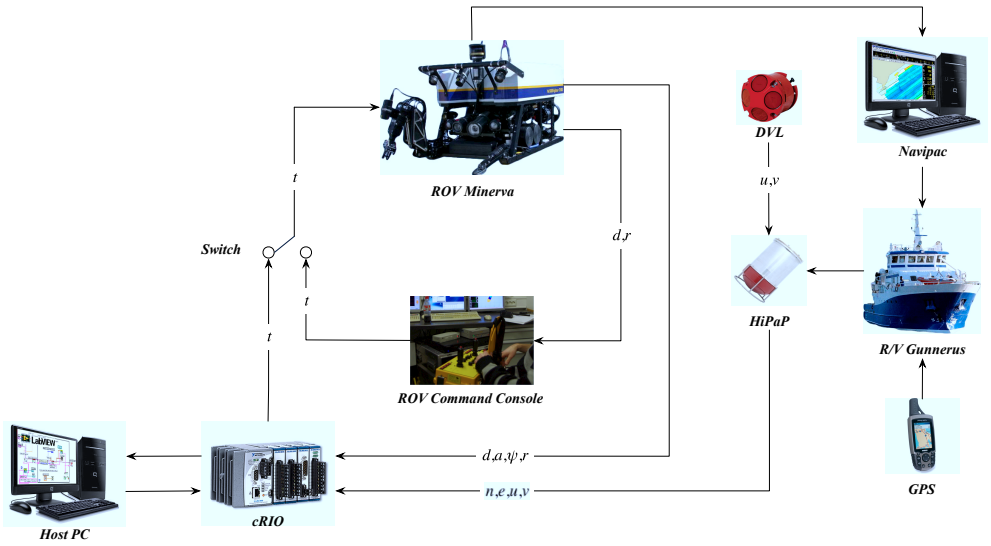


Figure 1.6: Overview of signal flow for ROV Minerva (Figure: Mauro Candeloro).

Dimensions	LWH 144x82x81 cm
Weight (air)	485 kg
Payload	20 kg approx.
Max depth	700 m
Power input	230 VAC, single phase 10 kW
Thrusters	Horizontal: 2 x 2000W Vertical: 2 x 2000 W Lateral: 1 x 2000 W
Speed	Horizontal: 2.0 knot Vertical: 1.2 knot Lateral: 1.3 knot Turn rate: 60/s
Camera 1 & 2	PAL colour CCD 460 TV lines, 0.1 lux
Camera 3	PAL colour Zoom 460 TV lines, 1 lux
Camera 4	3CCD Zoom High resolution PAL 530 TV lines, 15 lux
Sonar	Kongsberg Simrad MS 1000 (675 kHz) Beam width: 1.4 x 22 Fan (nominal) Range: 0.5-100 m (typical) Scan angle: 360 continuous Sonar-MS1000 (2011)
Manipulators	One 5-function hydraulic arm (HLK-HD5) One 1-function electric
Light	4 x 250 W halogen lights (4 channel light dimmer)
Sensors	100 bar pressure gauge fluxgate compass CRS03 silicon rate sensor Silicon-Sensing-Systems (2011) Teledyne RDI Workhorse Doppler velocity log (DVL) DVL (2011) MRU6 from Kongsberg seatex [1] MRU6 (2011) leakage detectors

Table 1.1: ROV Minerva Specificatioins

1.1.4 Marine Control Systems

The structure of a marine control system is illustrated in Figure 1.7. The upper square represents the part of the system which should be executed off-line, or more importantly it is to time consuming to be calculated real-time. This part involves the mission planning, and if needed re-planning of the mission based on some alarm triggered by a monitoring system. The local optimization is where the guidance system is executed, this part have a higher response time,

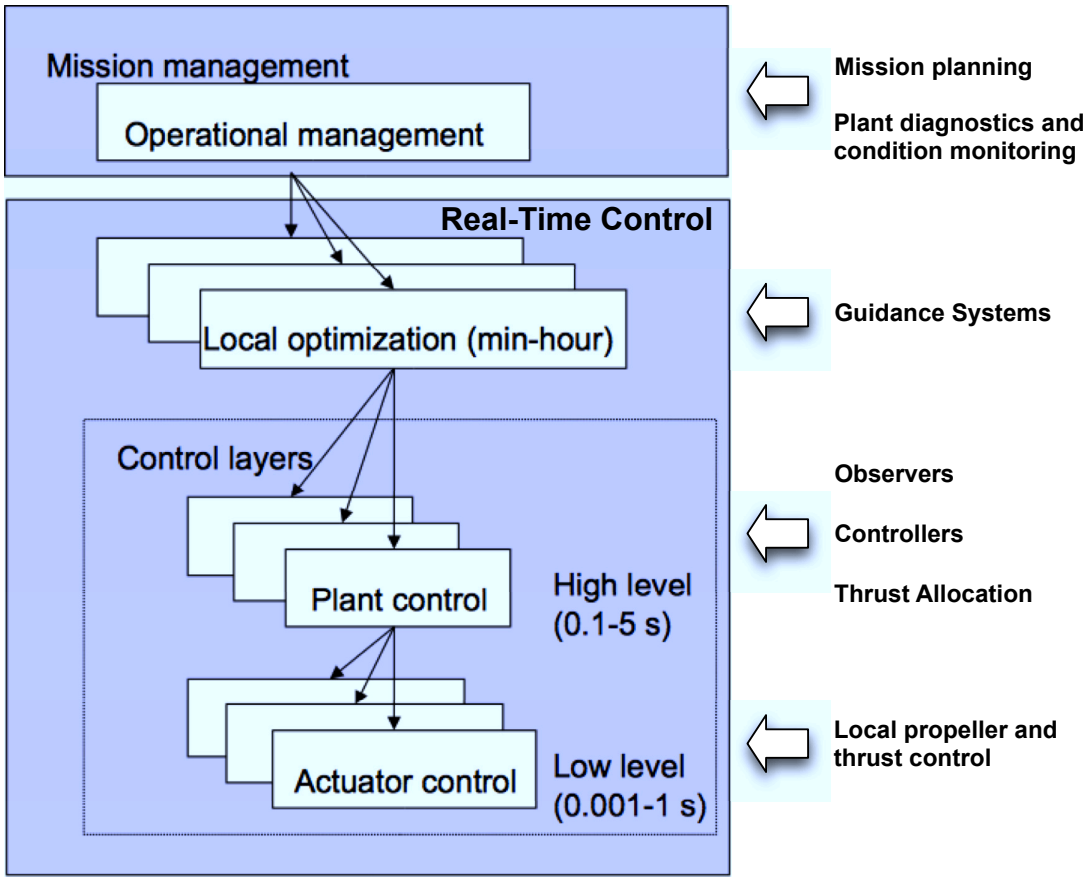


Figure 1.7: The structure of a marine control system (Sørensen, 2011).

such that it is placed high in the structure. This upper part of the structure is the focus of this thesis.

High level control consist in general of observers, controllers and thrust allocation. The low level control, where control of thrusters and propellers can be done by controlling shaft speed (rpm), pitch, torque and power or a combination of these. The controller for station keeping is typically of a PID type, while observers usually consists of a Kalman filter (Sørensen, 2011).

In order for the work done in this thesis to make any sense, it has depended on that the other layers of the control structure has been designed and implemented by other contributors. Controllers, thrust allocation and low level control was the focus of the work done by Marianne Kirkeby for her master thesis (Kirkeby, 2010a). Several observers has been implemented, where an extended Kalman filter designed by Mauro Candeloro as part of his master thesis (Candeloro, 2011) is the one mainly used for this thesis. Signal processing has been the focus of the master thesis of Laura Standardi (Standardi, 2011). As part of the AUR-Lab, also a navigation system is being developed, which has been the focus of the master thesis of Phat Truong (Truong, 2011).

1.2 Previous Work

1.2.1 Control Systems for UUVs

A lot of work has been done on control systems for AUVs. In particular Refsnes (2008) and the references therein are useful references on this topic. Also Fossen (2010) and his work on underwater vehicles has been particularly helpful for this thesis. For ROVs in particular, a

supervisory control system for the ROV Jason has been developed (Yoerger et al., 1986). A more recent approach to underwater vehicles considers hybrid underwater vehicles, where a combination of AUV and ROV and the challenges with control of these types of hybrids are highlighted. For the interested reader, Sørensen & Refsnes (2009) and the references therein should be investigated for further literature on this topic.

1.2.2 ROV Minerva

Design of control systems for ROV Minerva was previously attempted in Svendby (2007), where lack of an observer design and proper thrust allocation were influencing factors on the outcome. The AUR-Lab started its work with Minerva during the Spring of 2010, where initial experiments with stationkeeping were attempted with some success (Kirkeby, 2010b). The continued work with AUR-Lab was very fruitful, and some of the notable work done since 2010 can be further investigated in Kirkeby (2010a), Candeloro (2011) and Kørte (2010). A summary of some of the different functions and results so far is presented in Dukan et al. (2011).

Dr. Martin Ludvigsen has been working closely with ROV Minerva since it was purchased in 2003, and has been conducting several hydrodynamic experiments, both full scale and scaled model experiments. This includes a bollard pull test and a full scale thrust test (Ludvigsen & Ødegård, 2004). Much of his work with Minerva is summarized in Ludvigsen (2010).

1.2.3 Guidance

The work done on guidance is influenced by theory presented in Breivik & Fossen (2008), Breivik & Fossen (2004), Breivik (2010) and Fossen (2010). They present theory on both underwater vehicles and surface vehicles and presents experimental results on their findings. Breivik & Fossen (2008) focuses mainly on guidance laws for autonomous vehicles, while Breivik & Fossen (2004) presents the principle of steering for straight lines and circular arcs, which is applied in this thesis. The principle of Maneuvering theory is explained in Skjetne (2005) and Skjetne et al. (2004). The Maneuvering theory divides the path following objective in to two tasks, the first one purely geometrical which is the primary task, when the primary task is satisfied a dynamic assignment can be negotiated.

1.2.4 Collision Avoidance

Former NTNU students have done thorough literature review and application of several collision avoidance algorithm, particularly Loe (2008, 2007) and Engelhardtsen (2007). Collision avoidance by exposing a manipulator to a artificial potential field was proposed by Oussama Khatib (Khatib, 1985, 1986). The work with potential field has been explored further in Borenstein & Koren (1989) and limitations and possible solutions to these have been presented in Koren & Borenstein (1991). The work done led to development of the Vector Field Histogram which solves some of the problems with the pure potential field methods (Borenstein & Koren, 1991). The Dynamic Window approach to collision avoidance is presented in Fox et al. (1997), which takes an entirely different approach to collision avoidance by basing the algorithm on the physical limitations of the vehicle.

1.3 Contributions

The contributions of this thesis is within path planning, guidance principles and collision avoidance.

- A path following scheme for ROV Minerva to be able to follow a predefined path. The theory for the scheme is presented in Section 3.4 and Section 3.5, implementation considerations are done in Section 4.1.2. Full scale experiments of the path following scheme are presented in Section 4.4.
- Strategy proposals for guiding a vessel through a lawnmower pattern. Lawnmower patterns are addressed in Section 3.3, different strategies for manouvering through a 90 degree corner is discussed in Section 3.6. Simulations and an evaluation of the different strategies is carried through in Section 4.3.
- A collision system has been implemented for ROV Minerva. The theory behind this can be found in Chapter 5, while full scale experiments demonstrating the ability of the collision avoidance system are found in Chapter 6.

In order to put the contributions into a context, Figure 1.8 presents the architecture of the control system for ROV Minerva. The contributions within this thesis is generally within the guidance block. Within this block user interface, mission planning considerations and guidance algorithms is located.

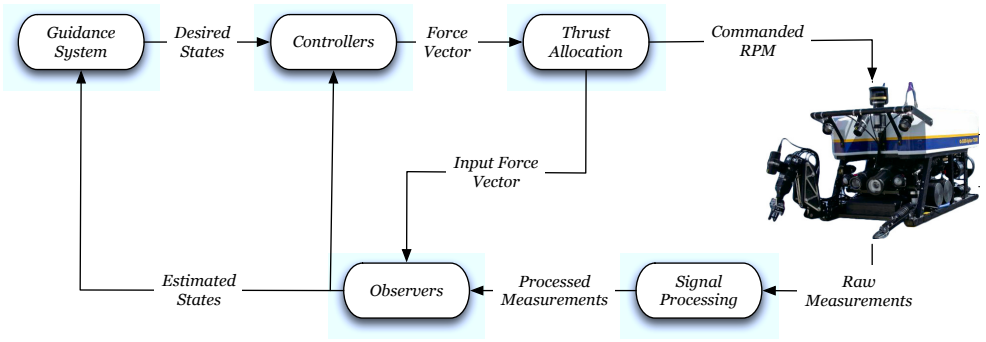


Figure 1.8: Overview of a marine control system.

1.3.1 Publication

The author has contributed to a conference paper which is a result of work done in the AUR-Lab.

- Daniel de Almeida Fernandes, Steffen Kørte, Fredrik Dukan, Martin Ludvigsen and Asgeir Sørensen (2011), "Exploring Limitations and Constraints of Guidance Systems for UUVs with Experimental Results", Submitted: OCEANS '11 MTS/IEEE KONA, Hawaii.

1.4 Outline of Thesis

Chapter 2 contains background information and preliminaries to the field of guidance and motion control.

Chapter 3 contains relevant theory and principles within motion control, path planning, path representation and path following. It also contains theory on propeller characteristics, and relevant considerations on the topic with respect to ROV Minerva.

Chapter 4 contains the description of the guidance system which has been implemented for the ROV Minerva. Simulations done with a model of the ROV Minerva is presented based on different guidance strategies. Several full scale experiments conducted are presented, in particular experiments on turning versus forward speed ability is demonstrated in addition to successful full scale lawnmower pattern experiments.

Chapter 5 contains relevant theory on collision avoidance and collision avoidance for underwater vehicles. The focus is on local collision avoidance algorithms for reactive behaviour with respect to obstacles. To put the local algorithms into context, also global collision avoidance algorithms are discussed in this chapter.

Chapter 6 contains the description of the collision avoidance system which has been implemented for ROV Minerva. The theory used, assumptions and considerations which have been done is described. Relevant simulations have been done in order to verify the functionality of the collision avoidance algorithm, while initial full scale results are presented.

Chapter 2

Preliminaries in Motion Control

Motion control is based on the concepts of reaching a given control objective by the means of feasible control inputs. A guidance system needs to take into account the type of control objective and the physical restrictions of the vehicle and use these to produce meaningful inputs to the control system. These control inputs need to fulfil the control objective as well as to keep within the physical boundaries of the vehicle. In this chapter preliminaries and background for motion control will be presented.

2.1 Kinematics

The notation used in this work is adapted from Fossen (2010), which is based on the standard notation from SNAME (Society of Naval Architects and Marine Engineers). Table 2.1 shows the forces, velocities and position for their respective degree og freedom in standard SNAME notation. A degree-of-freedom (DOF) for a marine craft is according to Fossen (2010): *"the set of independent displacements and/or rotations that specify completely the displaced position and orientation of the craft."* The generalized position and velocity is then defined as

DOF		forces and moments	linear and angular velocities	positions and 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 about the x-axis (roll)	K	p	ϕ
5	rotation about the y-axis (pitch)	M	q	θ
6	rotation about the z-axis (yaw)	N	r	ψ

Table 2.1: The SNAME notation.

$$\boldsymbol{\eta} = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}^\top \in \mathbb{R}^3 \times \mathcal{S}^3, \tag{2.1}$$

$$\boldsymbol{\nu} = \begin{bmatrix} u & v & w & p & q & r \end{bmatrix}^\top \in \mathbb{R}^6, \tag{2.2}$$

where $\boldsymbol{\eta}$ denotes the position and orientation vector and $\boldsymbol{\nu}$ denotes the linear and angular velocity vector.

2.1.1 Reference Frames

Several reference frames are commonly used to define a marine vehicles degrees of freedom, some of which are Earth-centered and some geographical reference frames. In this work the

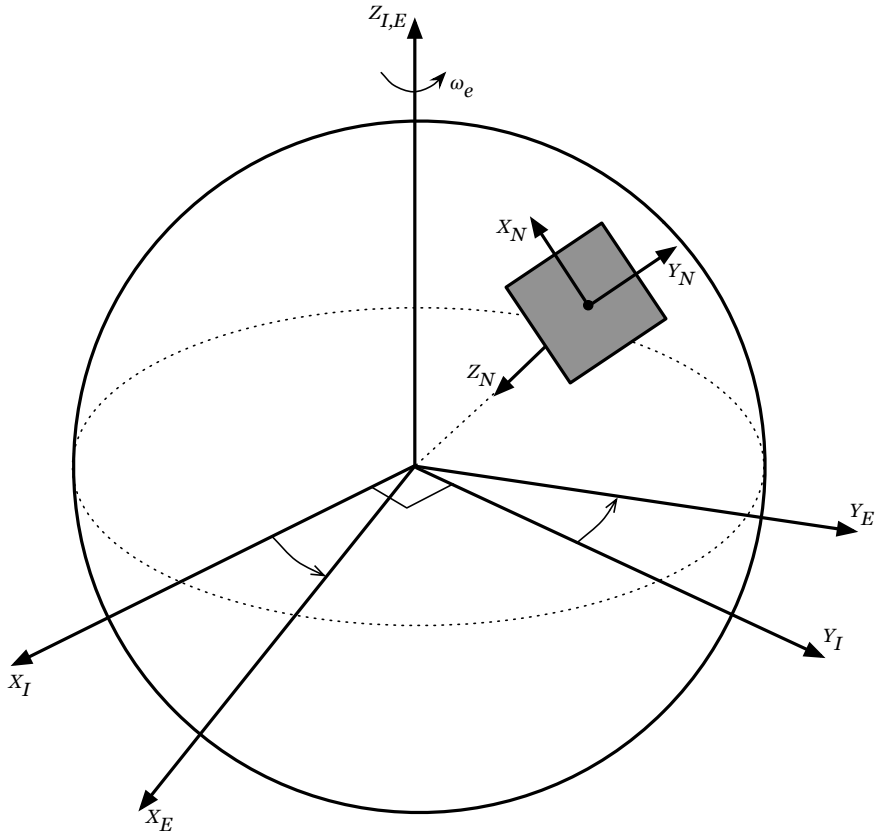


Figure 2.1: The three reference frames ECI, ECEF and NED, denoted by i, e and n respectively.

geographical reference frames will be considered for the most part, but a description of the Earth-centered frames will also be given in the following. The two Earth-centered reference frames are illustrated in Figure 2.1 together with the NED frame.

ECI Earth-Centered Inertial reference frame is denoted by subscript i, and position is given by $\boldsymbol{p}_i = [x_i, y_i, z_i]^\top$. This frame has its origin fixed at the center of the earth, and all the axes are also fixed. This means that it is a non-accelerating reference frame and Newtons laws thus apply.

ECEF Earth-Centered Earth-Fixed reference frame is denoted by subscript e, and position is given by $\boldsymbol{p}_e = [x_e, y_e, z_e]^\top$. This frame has, as the ECI reference frame, its origin fixed at the center of the earth and Z-axis fixed north. The X- and Y-axis however, rotates with the earth's angular rate of rotation $\omega_e = 7.2921\dot{1}0^{-5}$. So in cases where the rotation of the earth can not be neglected the ECEF reference frame should be used.

NED The North-East-Down reference frame is denoted by n, and position is given by $\boldsymbol{p}_n = [x_n, y_n, z_n]^\top$. It is a earth-fixed reference frame where the x-axis points North, the y-axis points East and the z-axis points down, with a fixed origin in the center of the tangent plane on the surface. When working in this reference frame it is usually referred to as flat Earth navigation, but usually the NED reference frame is considered inertial in order for the laws of Newton to apply. ROV Minerva for instance, has small operation areas, so assuming NED to be inertial is a good assumption. The position and orientation in this reference frame are given in $\boldsymbol{\eta}$.

BODY The body-fixed reference frame has its origin fixed to the marine craft. In this reference frame the x-axis is the longitudinal axis (from aft to fore), the y-axis is the transversal axis (directed to starboard) and the z-axis is the normal axis (from top to bottom). The linear and angular velocities are given in this reference frame in $\boldsymbol{\nu}$.

Path-fixed Path-fixed reference frame is the NED reference frame rotated such that the positive X-axis follows a straight line defined by an angle α_k from the NED frame. The X-axis will thus always be forward on the path, and the Y-axis will be perpendicular to it.

2.1.2 Transformation Between NED and Body

The relationship between the NED frame and the body-fixed frame is given as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{2.3}$$

where $\boldsymbol{J}(\boldsymbol{\eta})$ is the transformation matrix which is defined as

$$\boldsymbol{J}(\boldsymbol{\eta}) = \begin{bmatrix} \boldsymbol{R}(\boldsymbol{\Theta}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{T}(\boldsymbol{\Theta}) \end{bmatrix}. \tag{2.4}$$

The transformation matrix consists of the linear velocity rotation matrix $\boldsymbol{R}(\boldsymbol{\Theta})$ and the angular velocity transformation matrix $\boldsymbol{T}(\boldsymbol{\Theta})$. $\boldsymbol{\Theta}$ is defined as the Euler angles, $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$. These are given as

$$\boldsymbol{R}(\boldsymbol{\Theta}) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}, \tag{2.5}$$

$$\boldsymbol{T}(\boldsymbol{\Theta}) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix}, \tag{2.6}$$

where $c(\cdot) = cos(\cdot)$, $s(\cdot) = sin(\cdot)$ and $t(\cdot) = tan(\cdot)$. $\boldsymbol{T}(\boldsymbol{\Theta})$ is undefined for $\theta = \pm\frac{\pi}{2}$.

2.2 Guidance Terminology

Guidance systems are according to Fossen (2010) *Systems for automatic guiding the path of a vehicle system, usually without direct or continuous human control.*

2.2.1 Dynamic Positioning System

A dynamically positioned (DP) vessel is defined as *a vessel that maintains its position and heading (fixed location or pre-determined track) exclusively by means of active thrusters* (Sørensen, 2011). The concept dates back to the 1970s after a lot of success had been experienced with PID-based autopilot systems (Fossen, 2010). This initiated the development of a system with three decoupled PID controllers used to control the horizontal motion of a vessel, surge sway and yaw, exclusively by the means of active thrusters.

DP vessels are today commonly utilized, especially in the oil industry, and currently there exist approximately 1000 DP vessel (Wikipedia, 2011).

2.2.2 Actuation

In order to design a robust and meaningful guidance system, it is important to define the frames of your system. Therefore some terms will be clarified.

Degree of Freedom (DOF) A DOF is either a translation of a body or a rotation of a body, where translation is the ability to move without rotating while rotation is an angular motion around an axis. When talking about DOFs for a given body, we consider the set of independent DOF, so in a three dimensional space we have three translational and three rotational components.

Configuration Space The configuration space is the set of possible DOFs that a vehicle is able to obtain. So for a vehicle that moves in a three dimensional space the dimension of the configurations space is six, while for a vehicle moving in a two dimensional space the dimension of the configuration space is three.

Work Space Work space is used when defining the control objectives for a given vehicle. The work space is defined within the configuration space and has dimension equal to the number of DOFs the vehicles should control.

Actuation The term actuation is meant to express something about the relationship between independent control forces and moments that are available and the number of DOFs that are considered. If r is the number of independently controlled actuators and n is the dimension of the configuration space, a vehicle that has $r = n$ is said to be **fully actuated**. On the other hand if $r \leq n$ the vehicle is said to be **underactuated**.

Although a vehicle may be underactuated with respect to the configuration space a vehicle could be well defined within the work space and could perform meaningful tasks within the work space (Breivik, 2010). If we define m to be the dimension of the work space, then if $r = m$ the vehicle is fully actuated in the working space. Controlling such a vehicle is called control of underactuated vehicle. This should be distinguished from underactuated control of a vehicle. Underactuated control is different in the way that instead of reducing your work space to the number of independently controlled actuators you want to control more DOFs than number of independently controlled actuators, $r < m$. This is the case for example if DP in 3 DOF is the objective, and the vehicle only uses surge force and yaw moment, as can be further investigated in Pettersen & Fossen (2000).

When considering a ROV we have a configuration space with dimension $n = 6$, but usually only independent actuators in 4 DOFs ($r = 4$), hence we have an underactuated vehicle. Usually there are no independent actuators for roll and pitch control, but the ROV is stable in these DOFs, so we can define a work space $m = 4$. Control of the underactuated ROV is then possible in surge, sway, heave and yaw, which usually satisfies the control objectives of the ROV.

An AUV, on the other hand, is typically underactuated within its workspace, since it normally only can control three degrees of freedom, while it will typically have control objectives within a 6 DOF workspace.

2.3 Motion Control Scenarios

It is important to define the control objectives well, in order to approach the guidance problem in a correct matter. In the design of a guidance system the different types of motion control scenarios can be divided into four categories, *point stabilization, trajectory tracking, path following* and *maneuvering* (Fossen, 2010; Skjetne, 2005).

Point stabilization Restrict the motion of the vehicle towards, and converging, to a stationary point.

Trajectory tracking Restrict the motion of the vehicle along a time-dependent path.

Path Following Restrict the motion of the vehicle along a time-independent path.

Maneuvering Restrict the motion of the vehicle along a path, while satisfying some dynamic task.

2.3.1 Maneuvering

In maneuvering theory the path following problem is divided into two different tasks. The primary task being to converge to the path, while the second task is to satisfy a dynamic behaviour along the path. The maneuvering problem was proposed by Professor Roger Skjetne, and is thoroughly discussed in Skjetne (2005), Skjetne et al. (2004). The main task is to converge to and follow a path parametrized path, which can be represented as

$$\boldsymbol{p}_d^n(\varpi) = [x_d(\varpi), y_d(\varpi), z_d(\varpi)]^\top \in \mathbb{R}^3 \tag{2.7}$$

where the path is parametrized by the continuous path variable ϖ. Solving the path following problem for such a path can be done by solving the maneuvering problem. The two tasks for the maneuvering problem are defined as follows

Definition 2.1 (The Maneuvering Problem).

1. Geometric Task

For any continuous function $\varpi(t)$, force the position $p^n(t)$ to converge to the desired path $p_d^n(\varpi)$,

$$\lim_{t\to\infty} [\boldsymbol{p}^n(t) - \boldsymbol{p}_d^n(\varpi(t))] = 0 \tag{2.8}$$

2. Dynamic Task

The dynamic task can be any combination of the following tasks.

2.1 Time Assignment *Force the path variable ϖ to converge to a desired time signal $v_t(t)$,*

$$\lim_{t\to\infty} [\varpi(t) - v_t(t)] = 0 \tag{2.9}$$

2.2 Speed Assignment *Force the path speed $\dot{\varpi}$ to converge to a desired speed $v_s(\varpi, t)$,*

$$\lim_{t\to\infty} [\dot{\varpi}(t) - v_s(\varpi(t), t)] = 0 \tag{2.10}$$

2.3 Acceleration Assignment *Force the path acceleration $\ddot{\varpi}$ to converge to a desired acceleration $v_a(\dot{\varpi}, \varpi, t)$,*

$$\lim_{t\to\infty} [\ddot{\varpi}(t) - v_a(\dot{\varpi}, \varpi, t)] = 0 \tag{2.11}$$

The definition is taken from Skjetne (2005), but the denotation has been adapted from Fossen (2010) for consistency. In addition to the stated overall objectives in Definition 2.1, in order to define the maneuvering control objective, conditions and feasibility of the two tasks must be investigated and boundedness of the system states must also be required.

2.3.2 Alternative Classification of Motion Control Scenarios

A different classification of motion control scenarios was suggested in Breivik (2010) where the four categories above are replaced by the following four categories.

Target tracking The vehicle should track the position of a target of which only its instantaneous motion is known. A scenario where the target is a stationary point, corresponds to *point stabilization*.

Path following The vehicle should converge to and follow a predefined time-independent path. This corresponds to *path following* in the introduction of this section.

Path tracking The vehicle should track the position of a target of which the path is predefined and known. This could be divided into two tasks according to maneuvering by defining the geometric tasks as converging and staying on the path, and the dynamic task as synchronizing with the target.

Path maneuvering By optimizing a predefined path with respect to the vehicles maneuverability constraints, the vehicle should traverse the path in an optimal manner. This could for instance be by traversing the path as fast as possible. This involves the geometric maneuvering tasks and any combination of the dynamic maneuvering tasks.

2.3.3 UUV Scenarios

AUV missions will always involve some kind of path, due to their autonomy. The missions are often inspection missions, where sonar data, video data etc. is captured. Examples of AUV missions are:

- Following a predefined pattern to do seabed mapping.
- Following pipes of which the path is only partly known.
- Following a formation reference frame in formation with other AUVs.
- Following a predefined path under ice.
- Following an unknown path based on gradients of temperature, altitude etc.

ROVs have most often the possibility of dynamic positioning due to the actuation, which gives a ROV the possibility of doing missions which for an AUV is impossible. Example of missions for a ROV are:

- Dynamic Positioning. This involves both station keeping and trajectory tracking.
- Follow a predefined pattern to do seabed mapping.
- Follow an unknown path based on gradients of temperature altitude etc.
- Follow a predefined path, and stop and take samples or do some intervention work at given way-points

It should be noted that there is a great deal of overlap between mission that can be done by an AUV and a ROV, but some tasks are more common for one or the other. Such as dynamic positioning is something an AUV usually is not able to do, due to lack of actuation. A ROV is on the other hand not commonly used in formation control or lengthy mapping missions. Important restrictions for the two types are that an AUV has limited energy, while the ROV needs a support vessel which makes the restriction purely economic.

For this thesis focus will be on path following, with some overlap to maneuvering due to speed assignments. When using the term path following in this thesis, it will involve only the spatial constraint if no reference to speed assignment is given.

Chapter 3

Guidance Concepts

In this chapter guidance concepts and strategies for the different motion control scenarios will be discussed. In addition theory on path planning and path representation will be presented and demonstrated.

3.1 Guidance Objectives

3.1.1 Target Tracking

If the objective of a marine craft is to follow a target where no information about the targets path is available beforehand, a target tracking scenario should be considered. Advances in this topic traces back to missile guidance, in which a missile is controlled to destroy a given target. In marine applications, missile is substituted for interceptor, where the objective is for the interceptor to converge to a target. The objective is formulated as

$$\lim_{t\to\infty} [\boldsymbol{p}(t) - \boldsymbol{p}_t(t)] = 0, \tag{3.1}$$

where $\boldsymbol{p}(t) = [x(t), y(t), z(t)]^\top$ is the position of the craft, while $\boldsymbol{p}_t(t) = [x_t(t), y_t(t), z_t(t)]$ is the poistion of the target at time t. This definition is consistent with Breivik & Fossen (2008) and Fossen (2010). In these, three different guidance schemes are presented.

Line of Sight: This is considered a 3-point guidance scheme, with a stationary reference point, the interceptor and the target. The objective is for the interceptor to align its velocity with the line-of-sight between the reference point and the target. The line-of-sight vector will gradually change as the target moves, and the interceptor should converge to this vector.

Constant Bearing: This is a 2-point guidance scheme where only the interceptor and the target is considered. The objective is for the interceptor to align its speed relative to the target speed, such that the interceptor closes in on the target at a constant bearing. In other words it means to set out the course and speed assignments such that the interceptor and target is on collision course.

Pure Pursuit: This is also a 2-point guidance scheme, and as for the constant bearing guidance this scheme also considers only the interceptor and the target. The objective is to align the speed with the line-of-sight vector between the interceptor and the target.

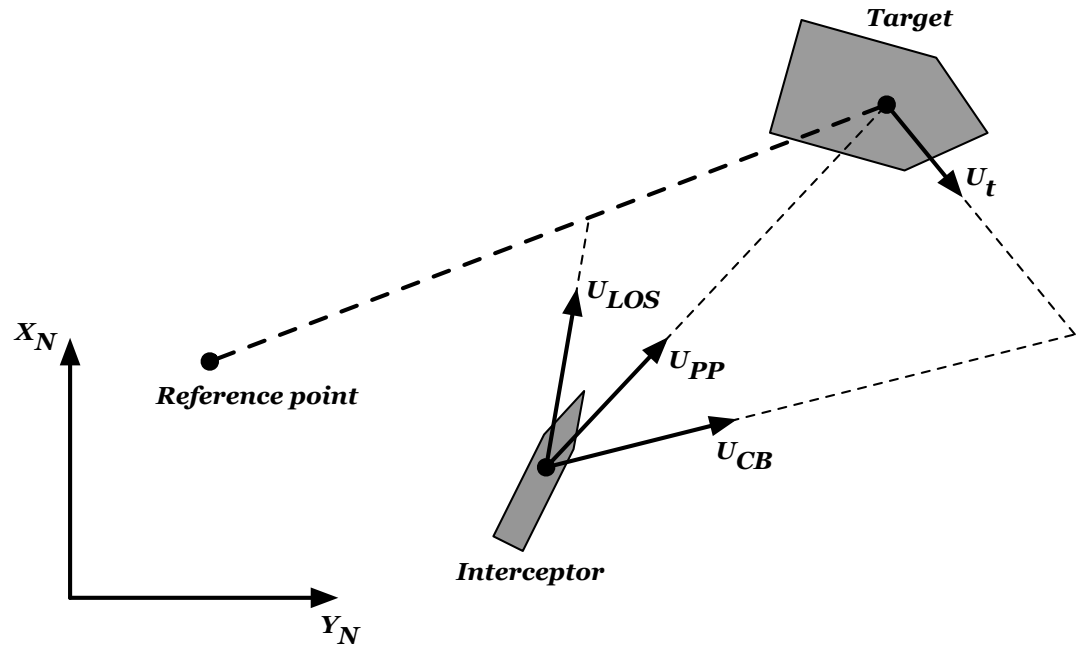


Figure 3.1: The three different guidance scheme in 2D. The arrows indicates where the interceptors velocity should be headed for the Line of Sight (LOS), Pure Pursuit (PP) and Constant Bearing (CB) scheme.

3.1.2 Trajectory Tracking

A control system that forces the system output $\mathbf{y}(t) \in \mathbb{R}^m$ *to track a desired output* $\mathbf{y}_d(t) \in \mathbb{R}^m$ *solves a trajectory tracking problem* (Fossen, 2010).

This means that in order to solve the trajectory tracking problem we want to generate a trajectory such that

$$\lim_{t\to\infty} (\mathbf{y}(t) - \mathbf{y}_d(t)) = 0. \tag{3.2}$$

This scenario is equivalent to considering a virtual target that follows a predefined path, which in some literature is called path-tracking. Thus, the problem could be solved using target tracking guidance laws. Differentiating $\mathbf{y}_d(t)$ once will reveal the desired velocities, and a second differentiation will reveal the desired accelerations. This means that information about the desired state, and the differentiated states are available for the controller. Different methods for generating $\mathbf{y}_d(t)$ are available in known literature, and some of these methods will be accounted for in this section.

Position Reference Model

Step inputs to a control system are generally undesirable, and could in worst-case scenarios make the system unstable. Therefore it is convenient to use reference models to generate a smooth trajectory. Position reference models generates a smooth trajectory from the initial position or orientation to a new desired position or orientation. The most common reference models are designed as a low-pass filter of the form

$$\frac{x_d}{r}(s) = h_{lp}(s). \tag{3.3}$$

where x_d is the desired signal, r is the reference signal and h_{lp} is the low-pass filter. Such a filter should be designed based on the physics of the process, and for a marine craft it is convenient

to design a reference model based on a mass-damper-spring system (Fossen, 2010). Based on this we can use a mass-damper-spring filter for generation of a smooth trajectory $\mathbf{x}_d \in \mathbb{R}^n$ of the form

$$h_{lp} = \frac{\omega_{n_i}^2}{s^2 + 2\zeta_i \omega_{n_i} s + \omega_{n_i}^2}, \tag{3.4}$$

where ζ_i are the relative damping ratios and ω_{n_i} are the natural frequencies. For position and attitude a reference model of third order is normally used. As Fossen (2010) suggest, a 1st-order low-pass filter should be cascaded with a mass-damper-spring system. We have the mass-damper-spring system from (3.4) and the a 1st-order low-pass filter

$$h_{lp}^1 = \frac{1}{1 + T_i s}. \tag{3.5}$$

The time constant is given as $T_i = 1/\omega_{n_i} > 0$. The resulting filter can then be written as

$$\frac{\eta_{d_i}}{r_i^n}(s) = \frac{\omega_{n_i}^3}{s^3 + (2\zeta_i + 1)\omega_{n_i} s^2 + (2\zeta_i + 1)\omega_{n_i}^2 s + \omega_{n_i}^3}, \quad (i = 1, \ldots, 6). \tag{3.6}$$

By the n in the notation r_i^n we mean that the input signal is given in the NED reference frame.

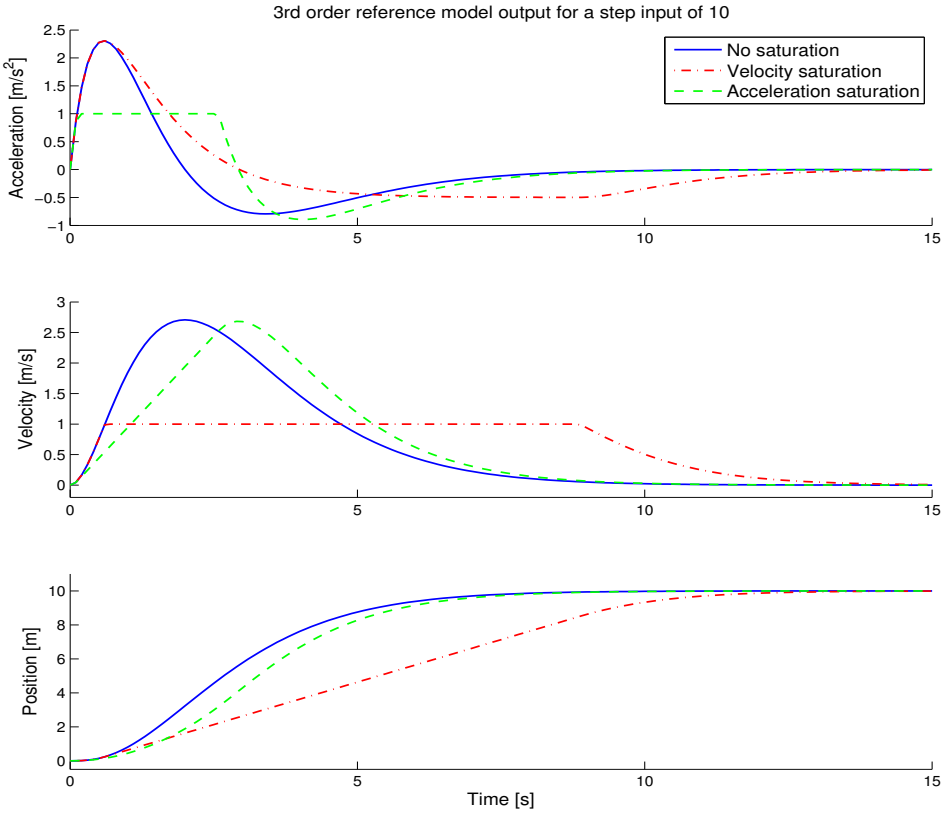


Figure 3.2: A reference model with $\omega = 1$ and $\zeta = 1$ (critical damping), with velocity- and acceleration saturation.

In order to obtain stability with such a model, it is important that the bandwidth of the reference model is chosen lower than the bandwidth of the motion control system. In order for this model to work satisfactory for all input signals r_i, the relative damping ratios and natural frequency

need to be chosen with respect to the magnitude of r_i. If information about the maximum and minimum velocities and/or the acceleration are available, saturating elements in these should be added to the reference model. This will assure that the reference model will stay within the physical limitations of the vehicle.

Three different trajectories are generated in Figure 3.2, where one that smoothly moves towards the reference point while the two others are saturated in velocity and also in acceleration in the third trajectory. The velocity saturation can be seen in the second plot while the acceleration saturation can be seen in the first plot. The problem with saturations is that in some cases the trajectory will get an overshoot, and in extreme cases also get unstable, which is undesirable, so measures should be taken to avoid this. In a reference model this is usually done by altering the natural frequency of the model. For multivariable systems the reference model in (3.6) can be written in vectorial setting as (Fossen, 2010)

$$\boldsymbol{\eta}_d^{(3)} + (2\boldsymbol{\Delta} + \boldsymbol{I})\boldsymbol{\Omega}\ddot{\boldsymbol{\eta}}_d + (2\boldsymbol{\Delta} + \boldsymbol{I})\boldsymbol{\Omega}^2\dot{\boldsymbol{\eta}}_d + \boldsymbol{\Omega}^3\boldsymbol{\eta}_d = \boldsymbol{\Omega}^3\boldsymbol{r}^n, \tag{3.7}$$

where $\boldsymbol{\Omega}$ and $\boldsymbol{\Delta}$ are diagonal design matrices containing the natural frequencies ω_{n_i} and the relative damping ratios. ζ_i. The reference model as considered in (3.7) assumes linear damping, which is most often not the case. Therefore it could for some cases be desirable to include nonlinear damping. The effect of adding nonlinear damping to the reference model is that velocity is reduced for large steps. In Fossen (2010) a revised model is suggested as

$$\boldsymbol{\eta}_d^{(3)} + (2\boldsymbol{\Delta} + \boldsymbol{I})\boldsymbol{\Omega}\ddot{\boldsymbol{\eta}}_d + (2\boldsymbol{\Delta} + \boldsymbol{I})\boldsymbol{\Omega}^2\dot{\boldsymbol{\eta}}_d + \boldsymbol{d}(\dot{\boldsymbol{\eta}}) + \boldsymbol{\Omega}^3\boldsymbol{\eta}_d = \boldsymbol{\Omega}^3\boldsymbol{r}^n. \tag{3.8}$$

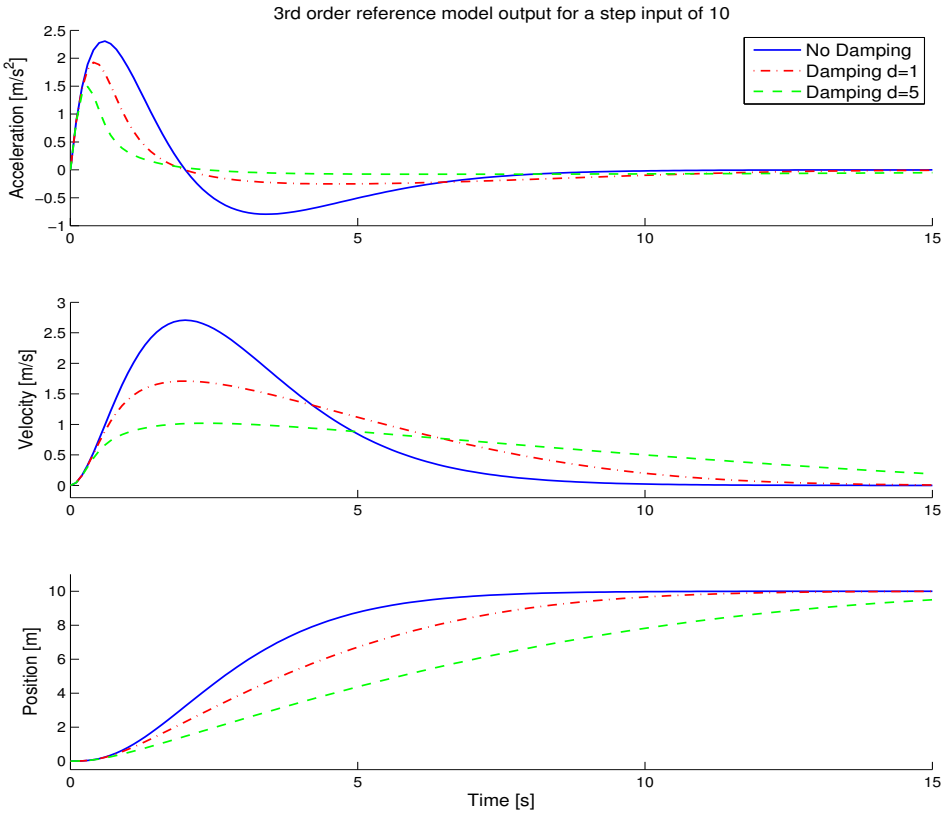


Figure 3.3: 3rd order reference model response with nonlinear damping.

The nonlinear damping term $\boldsymbol{d}(\dot{\boldsymbol{\eta}})$ could for the different DOFs be chosen as

$$d_i(\dot{\eta}_{d_i}) = \sum_j \delta_{ij} |\dot{\eta}_{d_i}|^{p_j} \dot{\eta}_{d_i}, \qquad (i = 1, \ldots, n) \tag{3.9}$$

where $\delta_{ij} > 0$ and $p_j > 0$ are design parameters.

To demonstrate the effect, consider (3.8), where the nonlinear term is $\delta_i |\dot{\eta}_i| \dot{\eta}_i$. The response of the new model can be seen in Figure 3.3 for $\delta = 0$, $\delta = 1$ and $\delta = 5$ respectively. The damping is apparent when the velocity in Figure 3.3 is considered, when the damping coefficient δ_i is increasing the response is less aggressive.

Velocity Reference Model

In many cases the main objective of a low-level control system is to control the velocity of a vehicle, where the position might be controlled by higher order control systems. In such cases it is convenient to use velocity reference models to smooth out step inputs in velocity. Where position reference models should be of 3rd order, velocity reference models are typically of 2nd order which is enough to produce smooth references for velocity and acceleration. This implies that (3.4) can be used as follows (Fossen, 2010)

$$\ddot{\boldsymbol{\nu}}_d + 2\boldsymbol{\Delta}\boldsymbol{\Omega}\dot{\boldsymbol{\nu}}_d + \boldsymbol{\Omega}^2\boldsymbol{\nu}_d = \boldsymbol{\Omega}^2\boldsymbol{r}^b. \tag{3.10}$$

This produces the desired velocity $\boldsymbol{\nu}_d$, the desired acceleration $\dot{\boldsymbol{\nu}}_d$. $\boldsymbol{\Delta}$ and $\boldsymbol{\Omega}$ are diagonal design matrices as defined for position reference models. This type of reference model will assure

$$\lim_{t \to \infty} \boldsymbol{\nu}_d(t) = \boldsymbol{r}^b. \tag{3.11}$$

In Figure 3.4 the response of a velocity reference model is shown for $\omega_n = \zeta = 1$. The velocity

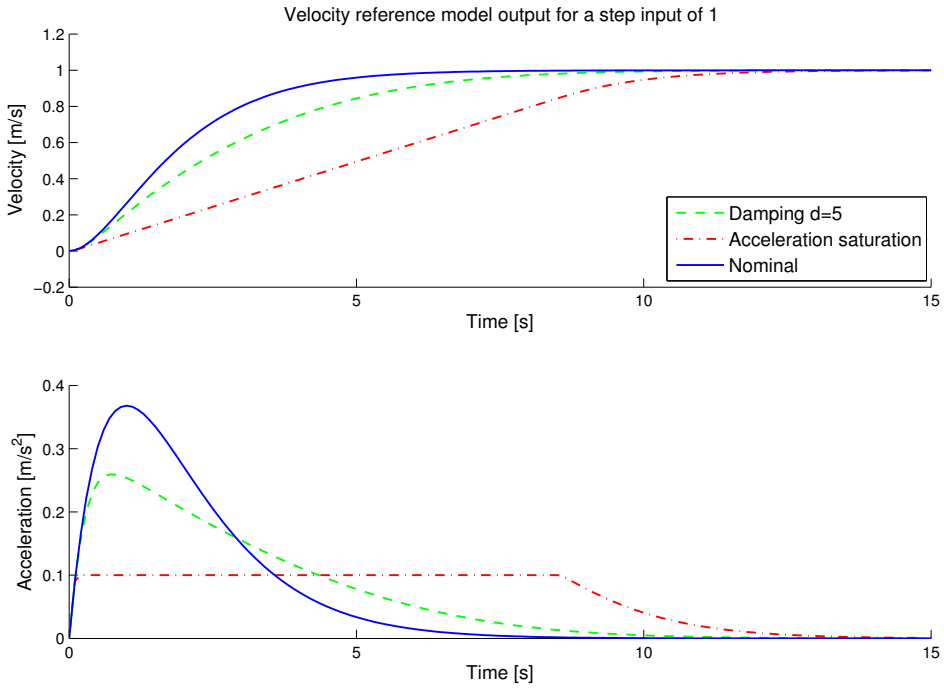


Figure 3.4: Response for a step using a velocity reference model.

reference model can, as position reference models can, be implemented with saturating elements

and damping. The damping will for a velocity reference model be on the acceleration as can be seen by the green line. This is not actually damping, as described in theory, but will give a reduction in acceleration nonetheless. Acceleration saturation is demonstrated by the red line.

3.2 Path Planning

When considering what type of theory to apply, it is important to take the mission criteria into account. There are several different criteria that can be taken into account, many of which can be used in planning the mission off-line while others are more dynamic and need to be considered during the mission and need to be taken into account on-line. Criteria that should be considered during off-line path planning are:

- Mission objective - From a starting point (x_0, y_0, z_0) the vehicle should go to a end point (x_n, y_n, z_n) through some way-points (x_i, y_i, z_i).
- Geographical considerations - Coastline, islands, shallow waters etc. should be taken into account if known. This can save a lot of time and energy instead of having to re-plan long routes to avoid obstacles that could be foreseen.
- Environmental considerations - Weather forecasts should be taken into account. Information about the wind, current and waves could be used to avoid bad weather, or take advantage of the weather.
- Physical constraints - The mission needs to be physically feasible with respect to velocity, acceleration limitations, turning rate etc.
- Energy constraints - Energy available and energy consumption. There needs to be enough energy to fulfil the mission and also for vehicle retrieval upon finishing the mission.

As can be seen, some of these factors may not necessarily be available beforehand which implies that the factors also need to be taken into account during on-line replanning of the mission. On-line criteria can be such as:

- Obstacles - This may be offshore constructions, geographic constraints which were not available beforehand etc. These are in general fixed obstacles.
- Environmental constraints - In cases where wind, waves or current change or where data was not available beforehand.
- Collision - Dynamical obstacles such as other vehicles must be avoided, and can not be foreseen.

All of these criteria will not be taken into account for this thesis, but it is important to be aware of them when designing a guidance system. If the vehicle is autonomous there should also be a decision system for deciding whether the mission should continue or abort during re-planning stages. If for instance an AUV has detected a object that is blocking the path, and preliminary avoidance algorithms do not succeed, the system can not know how big this obstacle is and must consider turning back for retrieval in case there is limited energy left.

3.3 Path representation

A predefined path is usually based on a set of way-points. How these way-points are defined will be decided through a human-machine interface (HMI). A way-point database could look like this,

$$\boldsymbol{W} = [(x_0, y_0, z_0), (x_1, y_1, z_1), \ldots, (x_n, y_n, z_n)], \tag{3.12}$$

where (x_i, y_i, z_i) for $i = 1, \ldots, n$ are the position coordinates for way-point i. Other information such as heading and speed assignments can also be given in the same manner. In addition to just saying something about position, the database could also include information about other criteria as mentioned in the preamble to this chapter. For now only position criteria will be considered, which means that a vehicle should pass through the way-points or within a area encircling the way-point.

3.3.1 Steering for Straight Lines and Circular Arcs

A simple and practical method for path representation is to steer for straight lines and circular arcs. In this method the path is represented by straight lines between the way-points while the path close to the way-points is defined by a radius R_i for $i = 1, \ldots, n$ around the way-point. The intersection between the arc of the circle and the straight line defines the turning point. The radius of the circles is found by

$$\underline{R}_i = R_i tan(\alpha_i), \tag{3.13}$$

where α_i, R_i and $\underline{R}_i$ is as defined in Figure 3.5. Considering the geometry of the figure, α_i can be found from the equation

$$a^2 = b^2 + c^2 - 2bc\cos 2\alpha_i \tag{3.14}$$

This means that (3.12) is expanded to include information about the radius that should be kept

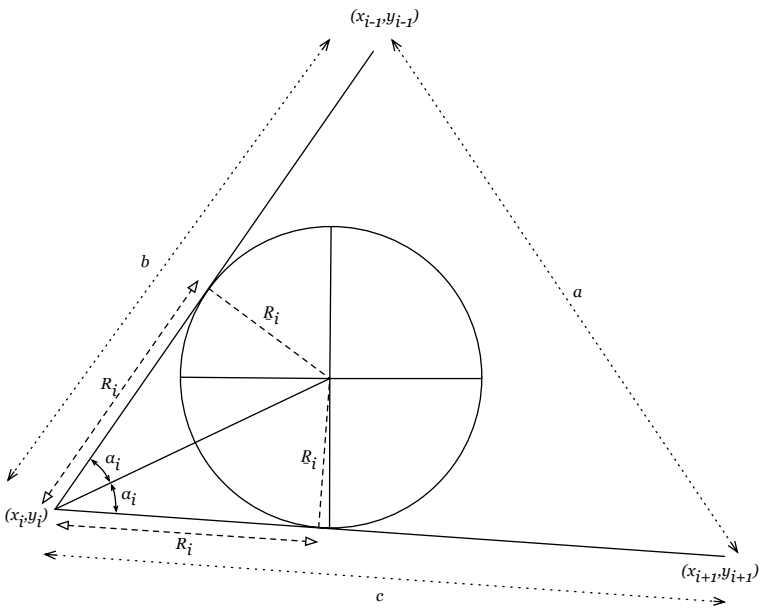


Figure 3.5: Geometric outline around a way-point in steering for straight lines and circular arcs path representation.

going through a way-point.

$$\boldsymbol{W} = \left[\{(x_0, y_0, z_0), (x_1, y_1, z_1), \ldots, (x_n, y_n, z_n)\}, \{R_0, R_1, \ldots, R_n\}\right], \tag{3.15}$$

The radius R_i reflects how long before the way-point the path is allowed to cut the corner. The radius chosen for each way-point should reflect the physical limitations of the vehicle. In case of α_i being very small, a large radius R_i should be chosen, which means the circle will start long before the vehicle reaches the way-point. This is due to the limitation in turning rate most ships and AUVs have when going at higher speeds. For a ROV moving quite slowly, while also having

full actuation, it could go close to the way-point and still manage the turn. $R_i = 0$ corresponds to only steering on straight lines between the way-points. In Figure 3.6 a path is represented by straight lines and circular arcs. The path is defined by

$$\boldsymbol{W} = [\quad \{(0,0),(50,50),(100,60),(100,90),(50,90),(30,130),(70,170),(0,200)\}\,, \\ \{10,10,10,10,10,10,10,10\}]. \tag{3.16}$$

The radius is chosen as $R_i = 10$ in all way-points for simplicity. This circle can be seen

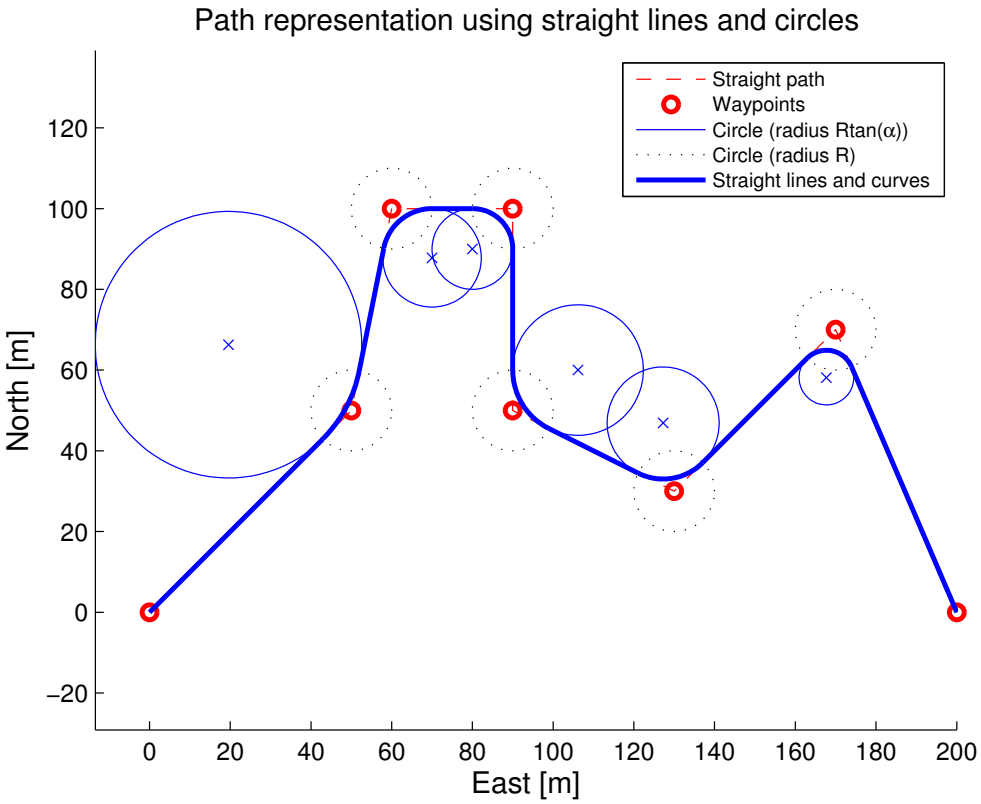


Figure 3.6: A path representation using the principle of straight lines and circular arcs.

encircling the way-points, while the steering circles vary depending on the angle between the lines connected to the way-points. The result is a smooth path as can be seen by the thick blue line in Figure 3.6. Depending on the objective of a given mission, a drawback with this method would be that the path does not actually pass through the way-points, which is also apparent in the figure. The circles encircling the way-points can in that sense be seen as a circle of acceptance. This means that the vehicle is considered to have reached the way-point when it is inside the circle of acceptance.

Lawnmower Patterns

When conducting seabed mapping and similar surveys, an attractive pattern to follow is a lawnmower pattern. Obviously the word lawnmower comes from the way most people are mowing their lawn, so the term should be intuitively known. Basically it is a nice way to cover an area by defining parallel lines which, if followed, will assure that the entire are is covered. The pattern in itself does not set any restriction on the path between the lines, only that the lines are covered. This type of pattern is the main focus concerning path following for the work done in this thesis.

One of the objectives for the ROV Minerva is to be able to follow lawnmower patterns in order to be able to to structured surveys such as seabed mapping or photo mosaic of the seabed.

To follow the lines is generally the easy part, while the vehicles trajectory between the lines is a little more challenging. Between leaving a line and entering the next line the vehicles has to turn 180 degrees. The desired path for handling both following the lines, and taking care of the behaviour between lines can be generated using straight lines and circular arcs. An example is shown in Figure 3.7. The user input is typically the end points and the starting points of the lines to be covered represented in the figure by the small circles. These are altered before entering the generator as represented in the figure by the crosses. The radius is for this example chosen as 2 metres, which generates a halfcircle as the trajectory between the lines.

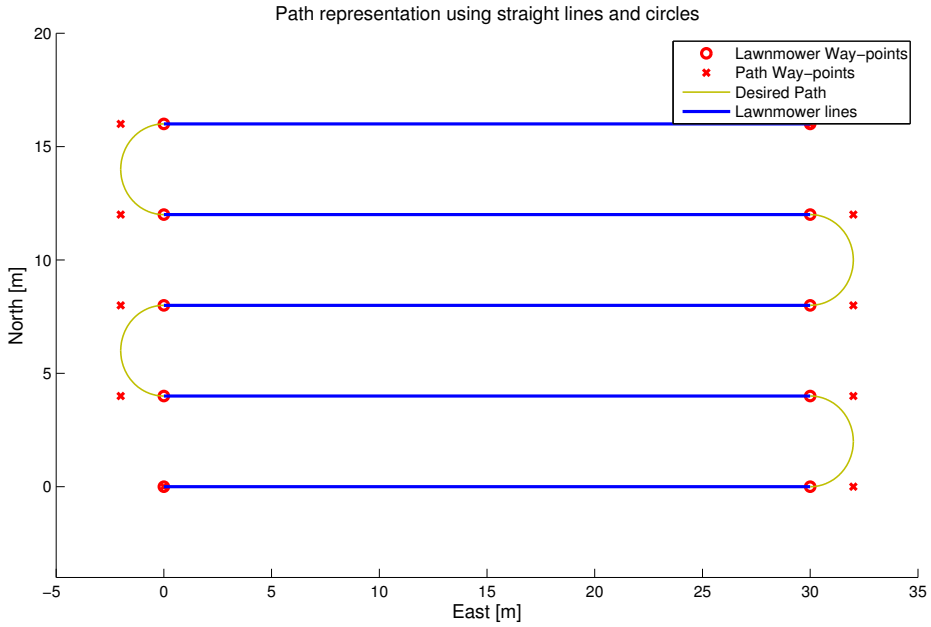


Figure 3.7: A path based on the steering for straight lines and circular arcs principle for following a lawnmower pattern.

If the lines are further from each other, and the vehicle can do faster turns than following half circles as in the example above, a smaller radius could be defined. In Figure 3.8 a lawnmower pattern is represented with radius of 0, 1, 3 and 5 metres, where the first one correspond to no circular path and the latter one corresponds to a halfcircle.

The paths generated in the two examples above, assumes that there are no restrictions between the lines. If the lawnmower pattern does not only involve following the horizontal lines, but the vehicle also must follow the vertical lines connecting them then a different approach is needed. It depends highly on the actuation and dynamics of the vehicle which approach that can be applied. For a ROV with full actuation in the horizontal plane with fast dynamics, it could stop at each way-point and change heading while staying at the way-point before starting on the next line. For a torpedo shaped AUV this is not possible, since forward speed is needed to manoeuvre. In that case an approach as presented in Figure 3.9 could be applied. Instead of cutting the corner the turn is taken outside the path, in order to have correct heading when reaching the lawnmower pattern again. The size of the circle should reflect the vehicles dynamics.

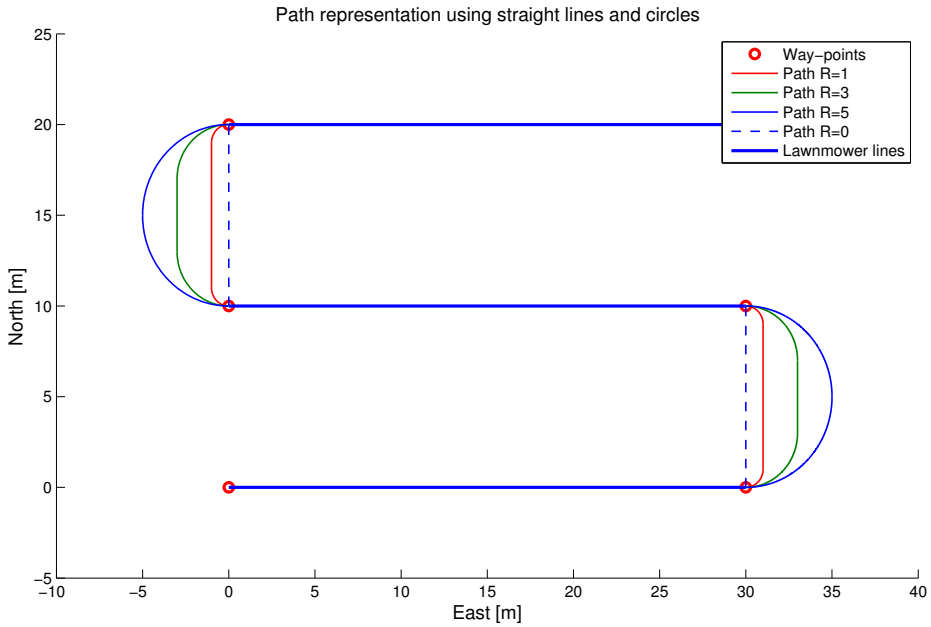


Figure 3.8: Lawnmower pattern using arcs of different radius.

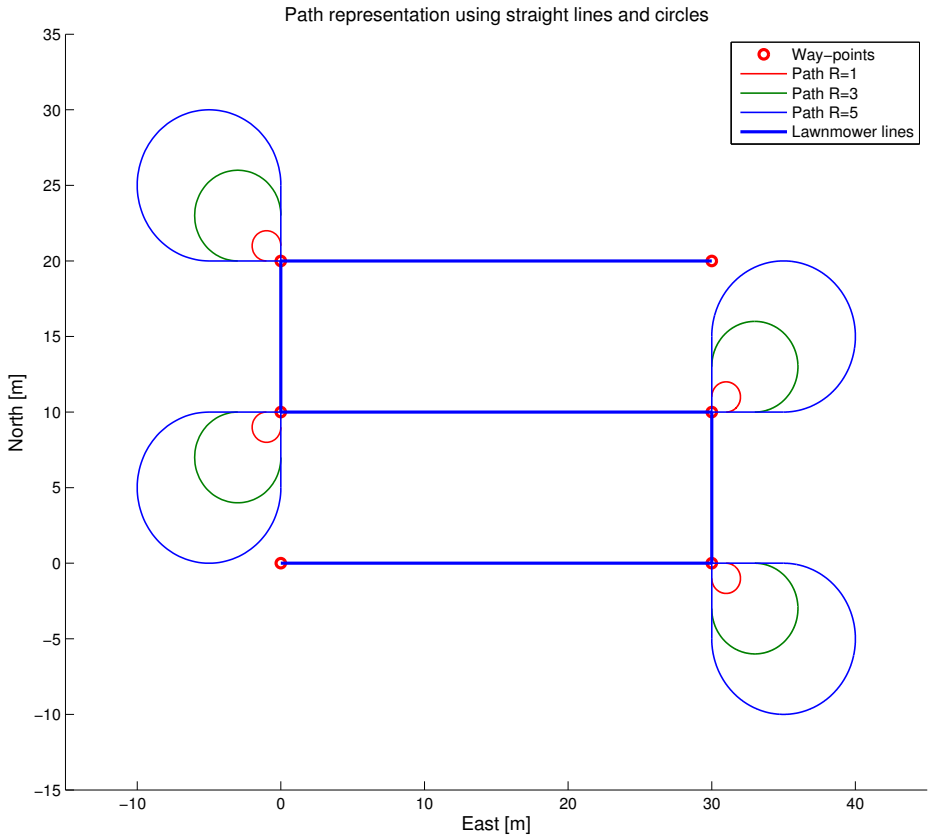


Figure 3.9: Lawnmower pattern where the lines between the lines also has to be covered.

3.3.2 Parametrized Paths

Methods using parametrized paths are also based on way-points, but now the way-points are used to generate a parametrized path. A 3-D representation of the path is commonly done like this,

$$\boldsymbol{p}_d^n(\varpi) = [x_d(\varpi), y_d(\varpi), z_d(\varpi)]^\top \in \mathbb{R}^3, \tag{3.17}$$

where the path is parametrized by the continuous path variable ϖ. For surface vehicle only $x_d(\varpi)$ and $y_d(\varpi)$ are used. For simplicity this part will only consider a 2-D path parametrization.

Two methods will be considered, Hermite interpolation and Cubic spline interpolation. Both methods consider N predefined way-points and the curve $(x_d(\varpi), y_d(\varpi))$ is a continuous curve passing through the way-points. The desired trajectory $(x_d(t), y_d(t))$ is found by choosing $\dot{\varpi} = k$ which implies $\varpi = kt$ were $k = 1, \ldots, N$ (Fossen, 2010). The methods are easily computed using Matlab and a path representation using Cubic spline and Hermite interpolation is presented in Figure 3.10. The path is defined by (3.16). There are some differences between the methods.

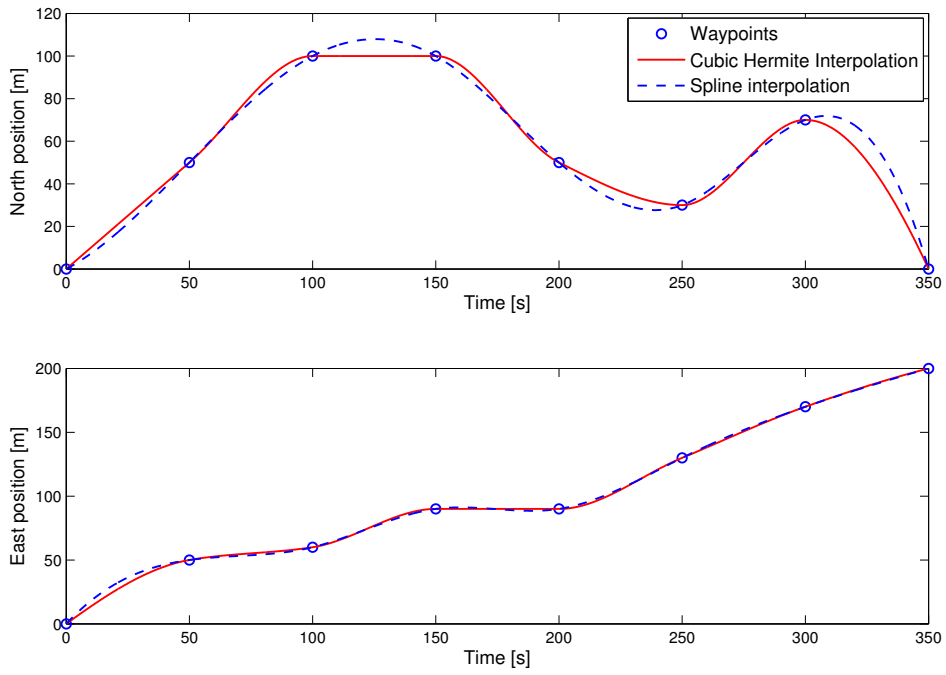


Figure 3.10: Path representation using parametrized paths.

The piecewise cubic Hermite interpolation ensures continuous 1st order derivatives, while the cubic spline requires that the 2nd order derivatives at the endpoints of the polynomial are equal. This will make the cubic spline smoother than the Hermite interpolation, as is demonstrated in Figure 3.10.

The two methods are presented in the horizontal plane in Figure 3.11, where also the path generated by using straight lines and circles is shown for comparison. It can be seen that the cubic spline gives a smoother path, and also smoother transition between way-points. The Hermite interpolation generates a path that is tighter with respect to the straight lines connecting the way-points. Both methods are generated such that the path passes through the way-points, so if the objective is to reach the given way-points no additional calculations are needed as opposed to the straight lines and circular arcs method.

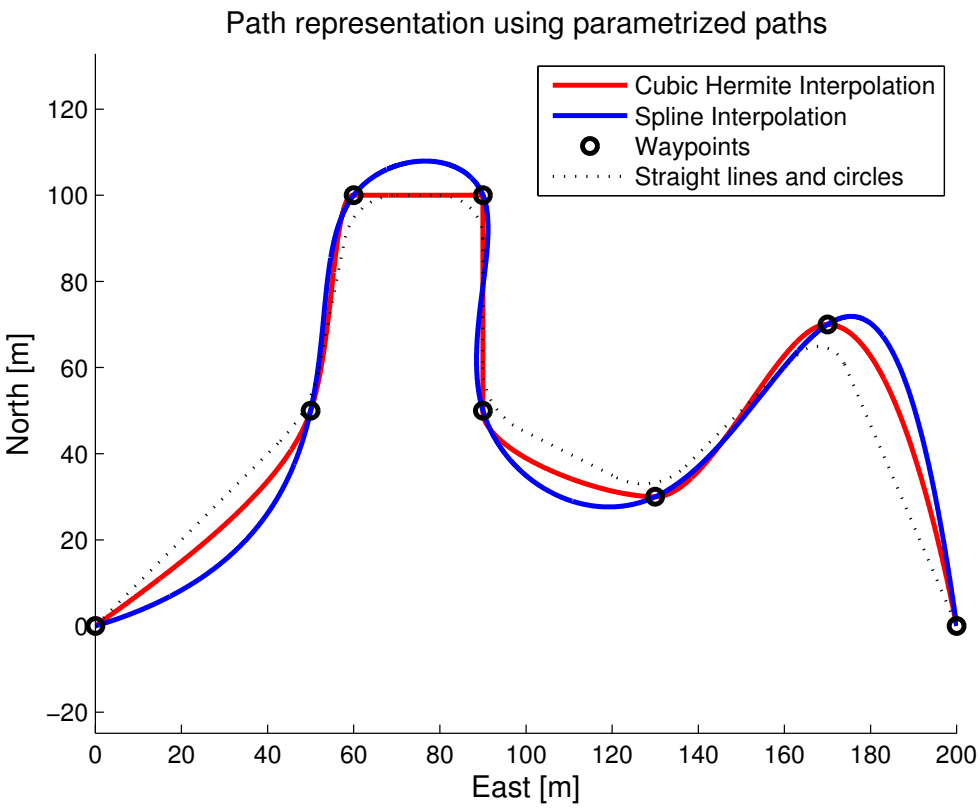


Figure 3.11: Horizontal path representation using parametrized paths, compared to straight lines and circular arcs.

3.4 Path Following

Consider a straight line between two way-points $\boldsymbol{p}_k^n = [x_k, y_k]^\top$ and $\boldsymbol{p}_{k+1}^n = [x_{k+1}, y_{k+1}]^\top$, and define a path-fixed reference frame with a rotation angle (Fossen, 2010)

$$\alpha_k = \text{atan2}(y_{k+1} - y_k, x_{k+1} - x_k) \in \mathbb{S}. \tag{3.18}$$

The coordinates of the vehicle in the path-fixed reference frame is $\boldsymbol{\varepsilon}(t) = [s(t), e(t)]^\top \in \mathbb{R}^2$ where

$$s(t) = \text{along-track error}$$
$$e(t) = \text{cross-track error}$$

as can be seen illustrated in Figure 3.12. If the objective of the path following is considered as primarily a spatial constraint, then the path following problem is solved by defining the control objective

$$\lim_{t \to \infty} e(t) = 0. \tag{3.19}$$

3.4.1 Steering on a Straight Line

When steering on a straight line, it is a matter of calculating the cross-track error and regulating it to zero. The coordinates in the path-fixed reference are given as (Fossen, 2010)

$$\boldsymbol{\varepsilon}(t) = \boldsymbol{R}_p(\alpha_k)^\top (\boldsymbol{p}^n(t) - \boldsymbol{p}_k^n), \tag{3.20}$$

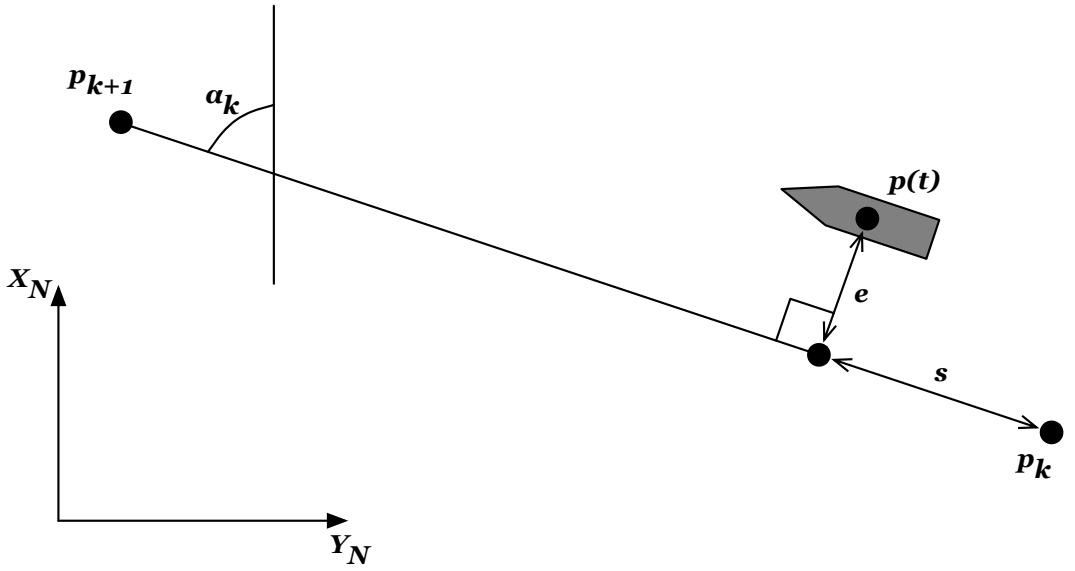


Figure 3.12: Path-fixed reference frame and its parameters cross-track error (e), along-track error (s) and rotation angle (α_k) for steering on a straight line.

where $\boldsymbol{R}_p(\alpha_k)$ represents the rotation from NED to path-fixed reference frame, and it is given by

$$\boldsymbol{R}_p(\alpha_k) = \begin{bmatrix} \cos(\alpha_k) & -\sin(\alpha_k) \\ \sin(\alpha_k) & \cos(\alpha_k) \end{bmatrix}. \tag{3.21}$$

The cross-track and along-track error is based on (3.20) and (3.21) calculated as

$$s(t) = \cos(\alpha_k)(x(t) - x_k) + \sin(\alpha_k)(y(t) - y_k), \tag{3.22}$$
$$e(t) = -\sin(\alpha_k)(x(t) - x_k) + \cos(\alpha_k)(y(t) - y_k). \tag{3.23}$$

The heading assignment and how to ensure (3.19) will be based on which method is applied, which will be addressed later in this section.

3.4.2 Steering on a Circle

When steering on a circle the path has non-zero curvature which implies that α_k is time-varying, and will here be denoted as $\alpha(t)$. Consider a circle as in Figure 3.13 where the center of the circle is denoted as $\boldsymbol{p}_c = [x_c, y_c]^\top \in \mathbb{R}^2$. The equation for $\alpha(t)$ can now be found based on the geometry in Figure 3.13

$$\alpha(t) = \chi_c + \frac{\pi}{2} \tag{3.24}$$

for clockwise circular motion or

$$\alpha(t) = \chi_c - \frac{\pi}{2} \tag{3.25}$$

for counter-clockwise circular motion (Breivik & Fossen, 2004). The angle depending on progress on the circle, χ_c is defined as

$$\chi_c = \arctan\left(\frac{y(t) - y_c}{x(t) - x_c}\right). \tag{3.26}$$

The radius of the circle is denoted r, and the cross-track error can now be found as

$$e(t) = r - \|\boldsymbol{p}(t) - \boldsymbol{p}_c\|_2. \tag{3.27}$$

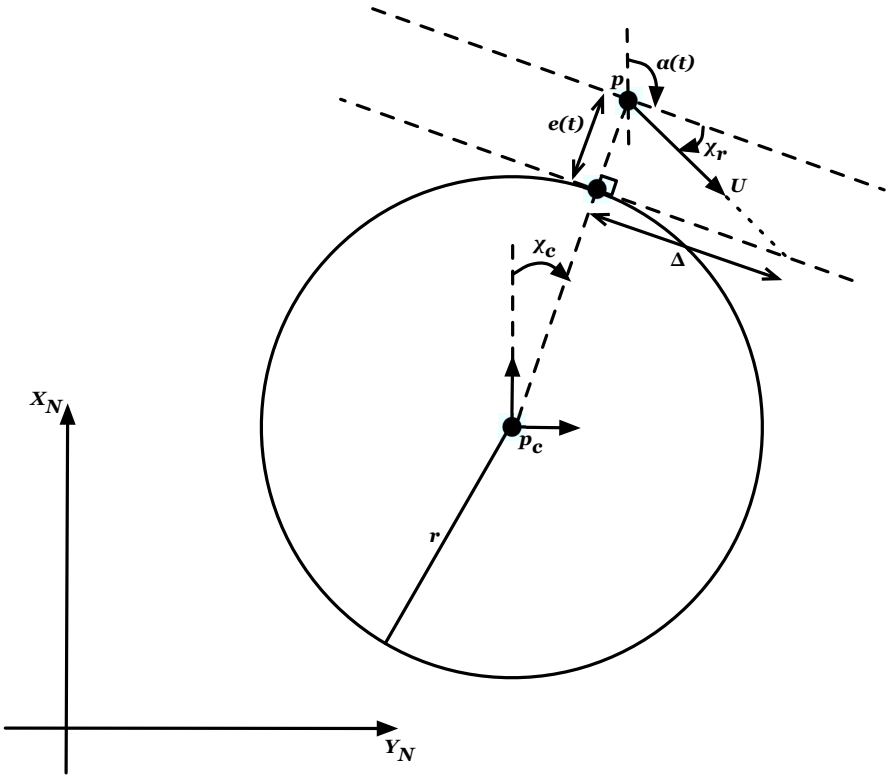


Figure 3.13: Path-fixed reference frame when steering on a circle.

The principle of steering on a circle is shown in Figure 3.13. The relevant parameters can be seen inscribed, where in particular the cross-track error e, progression on circle χ_t and the path-fixed rotation angle $\alpha(t)$ are the parameters used in the calculations above. The figure also has inscribed the parameters which are relevant for using line-of-sight principles for path following which will be addressed later in this section.

3.4.3 Line of Sight Path Following

For under actuated vehicles an attractive path following principle is line-of-sight path following. The principles was previously briefly discussed for target tracking, but will here be applied to path following. For a straight line the geometric considerations and parameters are presented in Figure 3.14. There are two different guidance principles which are used for LOS path following, enclosure-based steering and lookahead-based steering. In this the lookahead-based steering will be the focus. The course angle is defined by

$$\chi_d(e) = \chi_p + \chi_r(e), \tag{3.28}$$

which is dependent on the cross-track error, e and

$$\chi_p = \alpha_k. \tag{3.29}$$

The path-tangential angle α_k is found from (3.18), and

$$\chi_r(e) = \arctan\left(\frac{-e(t)}{\triangle}\right) \tag{3.30}$$

is the velocity-path relative angle, which ensures that the velocity is directed towards the point defined by the lookahead-distance. Definition on the angles defined in Figure 3.14 (ψ, χ, β) are adapted from Breivik & Fossen (2004).

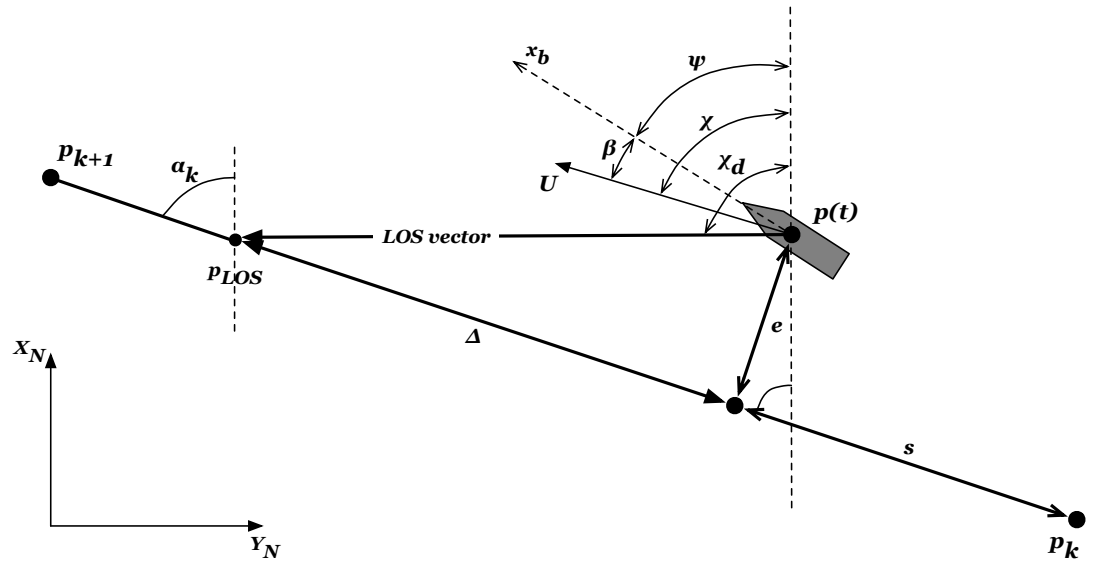


Figure 3.14: LOS path following, where the desired course (χ_d) is pointing towards a point ($\boldsymbol{p}_{LOS}$) at a lookahead distance ($\triangle$) ahead on the path.

Definition 3.1 (**Course angle** χ)**.** *The angle from the x- axis of the NED frame to the velocity vector of the vessel, positive rotation about the z-axis of the NED frame by the right-hand screw convention.*

Definition 3.2 (**Heading (yaw) angle** ψ)**.** *The angle from the x-axis of the NED frame to the x-axis of the BODY frame, positive rotation about the z-axis of the NED frame by the right-hand screw convention.*

Definition 3.3 (**Sideslip (drift) angle** β)**.** *The angle from the x-axis of the BODY frame to the velocity vector of the vessel, positive rotation about the z-axis of the BODY frame by the right-hand screw convention.*

The sideslip occurs due to currents. The speed is defined as $U = \sqrt{u^2 + v^2 + w^2}$, while in the presence of currents the speed is defined as $U = \sqrt{(u - u_c)^2 + (v - v_c)^2 + (w - w_c)^2}$ where u_c, v_c and w_c are the velocity components of the current. The course angle can easily be shown to be

$$\chi = \psi + \beta. \tag{3.31}$$

This relationship leads to the heading assignment

$$\psi_d = \chi_d - \beta. \tag{3.32}$$

Switching Between Way-points

If the path is made out of piecewise linear path made up of a number of straight lines, switching logic is needed for the algorithm to know when to regulate the cross-track error with respect to the next straight line segment. A common method is to define a circle of acceptance with a radius R. If the scenario in Figure 3.14 is considered, the vehicle approaches the point $\boldsymbol{p}_{k+1} = (x_{k+1}, y_{k+1}$ and should at some point switch to approaching $\boldsymbol{p}_{k+2} = (x_{k+2}, y_{k+2})$ from $\boldsymbol{p}_{k+1}$. The criteria based on a circle of acceptance principle is then to switch if

$$(x_{k+1} - x(t))^2 + (y_{k+1} - y(t))^2 \leq R_{k+1}^2. \tag{3.33}$$

Another method is to consider only the along-track error $s(t)$, which gives the advantage that no restrictions are set on the cross-track error in the switching mechanism. The total along-track distance between two way-points is denoted as s_{k+1}. The switching criterion can then be written as

$$s_{k+1} - s(t) \leq R_{k+1}, \tag{3.34}$$

where in this case R_{k+1} instead of being read as a radius now can be seen as a distance of acceptance.

For 3-dimensional scenario, (3.33) can be extended to include the depth w, while the second method is valid also for a 3-dimensional scenario.

Line of Sight Path Following on a Circle

If the path instead of being made up of only straight lines, is made up of straight lines and circular arcs as discussed in Section 3.3.1 some considerations has to be done. The lookahead principle on a circle is pretty much the same as for a straight line, where the lookahead distance is inscribed in Figure 3.13 and instead of using α_k as defined in (3.18) the equations for $\alpha(t)$ in (3.24 or 3.25) is replacing it. The equation in (3.28) is used but the cross-track error in (3.30) is now taken from (3.27) and $\chi_p = \alpha(t)$.

The switching logic also needs alteration when the path is different. Instead of switching to the next straight line, now the algorithm should switch to the arc at the point where the straight line is tangential to the arc. This point is previously discussed in Section 3.3.1 where the switch to circle guidance should be done when the vehicle is within a distance of $s_{k+1} - s(t) \leq R_i$ from the the next point $\boldsymbol{p}_{k+1}$.

3.5 Cross-Track Error Regulation for a Fully Actuated ROV

The Line of Sight principle is effective for under actuated vehicles since they will not be able to control motion in sway. Most ROVs have actuation in 4 DOFs (surge,sway,heave,yaw), and are not relying upon heading assignments to follow a path. As most vehicles a ROV can move fastest in surge, it implies that the heading should be kept equal to the direction of the path. If no sway motion is assumed, then surge motion would make the vehicle traverse the path, as long as the heading always is equal to the direction of the path, $\psi = \alpha(t)$.

This means if actuators are available to control sway motion, this principle could be applied also with motion in sway, where sway motion is in charge of fulfilling the control objective

$$\lim_{t \to \infty} e(t) = 0. \tag{3.35}$$

If the cross-track error is fed to the controller as feedback on the error, the cross-track error will be regulated as in (3.35). A position controller would have to be body frame in order to assure sway motion, and not motion in the East direction. Alternatively, the control objective could be reached by applying a body velocity-control for sway based on the cross-track error. In that case the objective is to regulate the sway motion to zero while not violating (3.35).

For a ROV this may be a better approach than a LOS approach, since the actuation is better utilized and should also give better performance since the heading is kept constant on straight lines, while yaw-rate is kept constant during turns. The problem with LOS guidance is that for short lookahead distance it could easily result in oscillations in yaw, especially for a small ROV compared to larger vessels and torpedo shaped AUVs.

This thesis considers only constant depths, with the assumption that the ROV Minerva, given feasible desired states, is able to follow a pattern in the vertical plane only by the means of the two vertical thrusters. This means that the theory presented in this chapter could be extended

without including new considerations. The vertical thrusters are assumed independent of the other thrusters. This is not entirely true, since the horizontal thrusters are located slightly below the middle line, such that if the two forward thrusters are given high thrust, some some pitch angle will be induced. It is this pitch angle which is neglected, based on the low velocities which in general will be used.

3.6 Strategies for Path Following

In Section 3.3, lawnmower patterns were discussed. In this section the focus will be on different strategies for following a pattern consisting of straight lines. Methods for following a path has been presented, but how should the path be represented in order to fulfil the objectives? Another point of interest is how the different objectives should be weighted to find out which objectives take priority.

3.6.1 Four Approaches to a Corner

Lawnmower patterns will be used as a reference. The different principles will be based on what type of actuation the vehicle is equipped with, mission objective and energy consumption. Considering UUVs it is interesting to look at the restriction of the under-actuated AUV, compared to the fully actuated ROV. An AUV is dependent on forward speed to manoeuvre, while the ROV normally has fast dynamics and can move in 4 DOFs. In this section a 90 degrees corner will be considered, which can be seen as a corner in a lawnmower pattern. For this corner 4 different strategies for taking the turn will be considered. The 4 different strategies are depicted in Figure 3.15, and are denoted by their number.

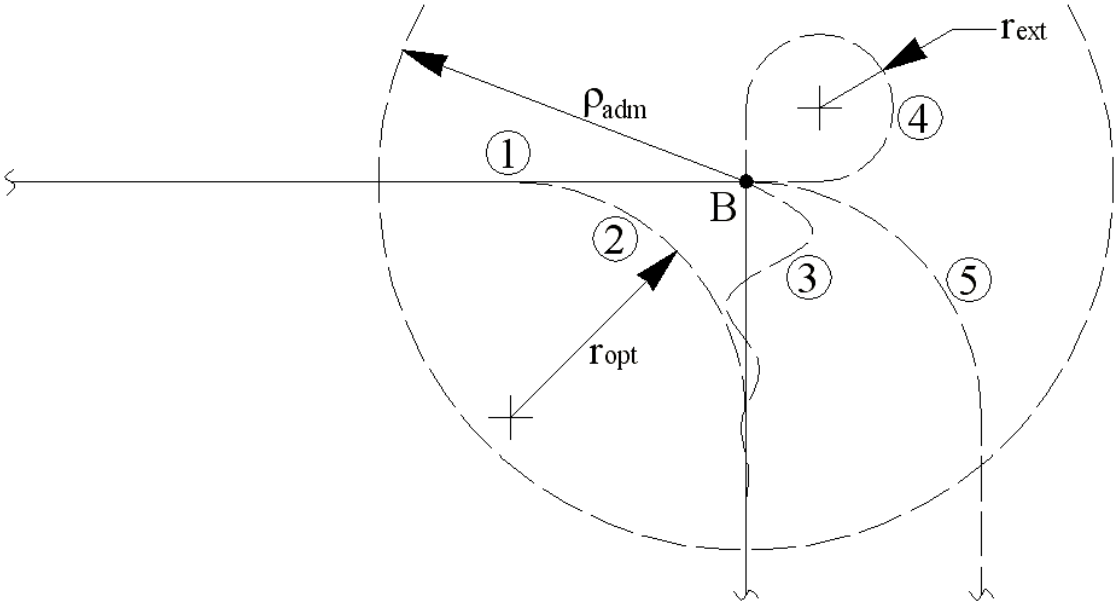


Figure 3.15: A section of a lawnmower pattern is seen, where the path comes from left with a direction of 90 degrees, before it takes a 90 degree turn to -180 degrees. The encircled number represents different strategies for turning the corner. Illustration: Daniel de Almeida Fernandes

The different approaches will needs some further explanation:

① This line presents a scenario where the vehicle should stop in B, change its heading corresponding to the line leading away from point B, and then accelerating to a desired velocity

along the new line. With respect to UUVs, this scenario is only feasible for a ROV with the needed actuation available. A traditional torpedo shaped will not be able to do this manoeuvre.

② This line presents a scenario where the system is given permission to relax the restriction of following the lines, and can cut the corner, and this way feasibly manoeuvre from on line to the next line. This scenario is feasible for both an AUV and a ROV.

③ This line presents a scenario where the behaviour during the switch from one line to the next is neglected, and the drawbacks by doing this is accepted. In general, this will provoke oscillations, as demonstrated by the line. This way of handling the corner will put much of the responsible of the manoeuvring on the actuators, which could cause wear and tear.

④ The last scenario which is discussed is the scenario where the restriction to cover all the parts of the lines is kept tight. The vehicle is allowed to go outside of the lines in order to obtain the desired heading when starting along the next line. This is an attractive approach for an under actuated AUV that needs forward speed to manoeuvre.

Scenario ③ can actually be considered a special case of ①, in the sense that velocity should be reduced by a factor during switching between lines, where in ③ this factor is zero. For both scenario ② and ④, the velocity could be reduced while moving on the arcs, this in order for the traversing velocity for the entire path to be restricted by the maximum velocity the vehicle is able to keep on the arcs. All of the methods have both advantages and drawbacks, which will be further discussed in Section 4.3, where simulations of the different scenarios will be presented as well.

3.6.2 Velocity Reference Model for Steps in Position

The scenario presented in Figure 3.15, denoted by ① requires an algorithm for the vehicle to decrease its velocity to zero, while also assuring that the vehicle reaches the way-point B. The reference models described in 3.1.2 are difficult to control, due to the tuning parameters nature.

Another type of reference model has been developed through the AUR-Lab and will be presented in Fernandes et al. (2011). A line from A to B is divided into different phases. Firstly the acceleration phase will be active until reaching a constant desired velocity, where the next phase is to keep this velocity until a flag is raised. This flag says that in order to reach zero velocity when reaching B, the deceleration phase should be set in effect. The deceleration phase will be in effect until the velocity reaches zero again. If the vehicle is able to follow the generated acceleration and velocity signals, the vehicle should stop at point B, where some other guidance scheme should takeover, such as a DP mode.

3.6.3 Propeller Characteristics

In order to compare the different strategies in terms of propeller force, a quick look into propeller theory is needed. The relationships between the shaft speed of a propeller n , and the thrust T_a and torque are according to Sørensen (2011)

$$T_a = f_T(\cdot) = \text{sign}(n) K_T \rho D^4 n^2, \tag{3.36}$$

$$Q_a = f_Q(\cdot) = \text{sign}(n) K_Q \rho D^5 n^2, \tag{3.37}$$

Where D is the propeller diameter and ρ the density of water. The thrust and torque coefficient, $K_T > 0$ and $K_Q > 0$ are connected to the open-water efficiency η_0 of the propeller through

$$\eta_0 = \frac{J_a K_T}{2\pi K_Q}, \tag{3.38}$$

where the advance ratio J_a is given by

$$J_a = \frac{V_a}{nD}, \tag{3.39}$$

where V_a is the inflow velocity on the propeller. The propeller power consumption is calculated as (Sørensen, 2011)

$$P_a = 2\pi n Q_a = \text{sign}(n) 2\pi K_Q \rho D^5 n^3. \tag{3.40}$$

The thrust coefficient is assumed to account for thrust losses. The thrust loss is explicitly included by modifying (3.36) by including a thrust loss facto θ as follows

$$T_a = f_T(\cdot) = \text{sign}(n) K_T \theta \rho D^4 n^2. \tag{3.41}$$

The thrust loss factor is estimated based on bollard pull, found in (Ludvigsen & Ødegård, 2006). The bollard test results and the estimated thrust loss factor can be found in Table 3.1. The

	Bollard Pull [N]	Estimated Bollard Pull [N]	Thrust Loss Factor θ
Forward	478	663	0.72
Backwards	219	414	0.53
Sideways	195	187	1.04
Down	389	673	0.58

Table 3.1: Bollard Pull for ROV Minerva.

thrust loss factor for the lateral thruster can be seen to increase, which is probably due to that the bollard pull did not take into account that the lateral thruster has two propellers working in opposite directions (Kirkeby, 2010a). The thrust coefficient for zero inflow velocity has been found from testing, and the following values will be used:

$$K_{T0_{forward}} = 0.24, \tag{3.42}$$

$$K_{T0_{backwards}} = 0.15, \tag{3.43}$$

where the first coefficient is used for positive thrust, and the second is used for negative thrust.

There have not been conducted test to find a torque coefficient, which makes finding the energy difficult. In order to compare energy consumption for the different strategies, a torque coefficient will be an assumed estimate. This will be explicitly explained when such assumptions are made. The thrust coefficient is based on the nominal thrust, which means that it is based on zero inflow velocity on the propeller. The thrust and torque calculations are therefore the nominal thrust and nominal torque. In simulations and experiments only the desired rotation speed n_d is known, such that (3.36), (3.37) and (3.40) is modified as follows:

$$T_d = f_T(\cdot) = \quad \text{sign}(n_d) K_{T0} \rho D^4 n_d^2, \tag{3.44}$$

$$Q_d = f_Q(\cdot) = \quad \text{sign}(n_d) K_{Q0} \rho D^5 n_d^2, \tag{3.45}$$

$$P_d = 2\pi n_d Q_d = \quad \text{sign}(n_d) 2\pi K_{Q0} \rho D^5 n_d^3. \tag{3.46}$$

To calculate energy consumption, the power is integrated as

$$\textit{Energy Consumption} = \int P_d dt. \tag{3.47}$$

Chapter 4

Guidance System for ROV Minerva

This chapter presents the work done particularly for the ROV Minerva. This chapter will explain which functions that are implemented and will also give some insight into how the implementation has been conducted. During the process of implementation and testing it has been made clear that implementing the actual guidance laws is fairly simple, but the logistics that connects everything is the time consuming part. Result from full scale experiments conducted with ROV Minerva will be presented towards the end of this chapter.

4.1 The Guidance System

An outline of the control system is given in Figure 4.1. In addition to the blocks in the figure, there are also extensive logic connecting the different blocks. Blocks that are not included in the figure include signal input and output blocks. The work done in this thesis is mainly found in the guidance block. The guidance system has in addition to input from operator, also

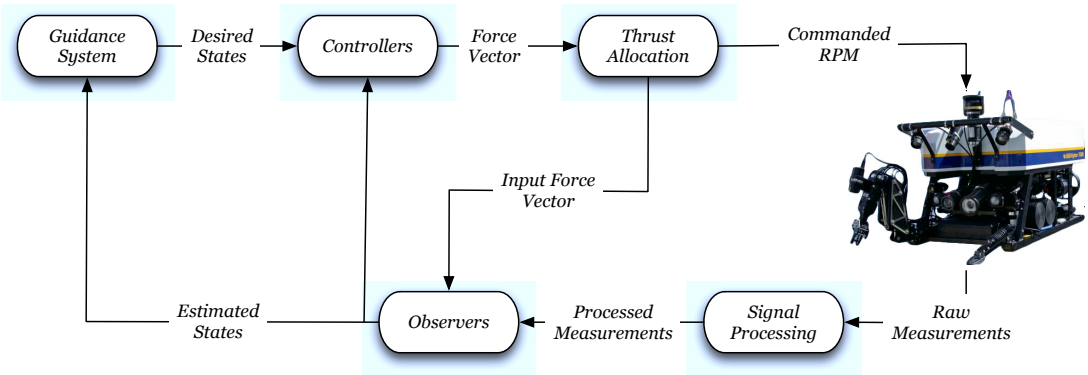


Figure 4.1: Signal flow of the control system for ROV Minerva. The guidance block is the focus of this thesis.

input from motion sensors. Also, information about wind, waves and current are sometimes available. In the guidance system described in this thesis, the inputs are from operator and from motion sensors. Information about wind and waves are irrelevant as long as the ROV is deeply immersed. Information about the currents would be useful, but no information from sensors are currently available for the ROV Minerva.

The main functionality within the guidance block can be seen in Figure 4.2. The operator enters input parameters through a HMI, and choses which mode to set the guidance system to use. The different modes also have state feedback since the algorithms are based on current position and velocity. Through the HMI the operator gets information about all aspects of the

control system, and has the possibility to chose which type of guidance strategy that should be used at any time. An important aspect of the system which is not shown properly in the figures is the switching of controller based on the control objective. For instance will the Dynamic Positioning (DP) mode require a position controller, while the path following mode requires a velocity controller. This is controlled by some supervisory algorithm. The Joystick control mode

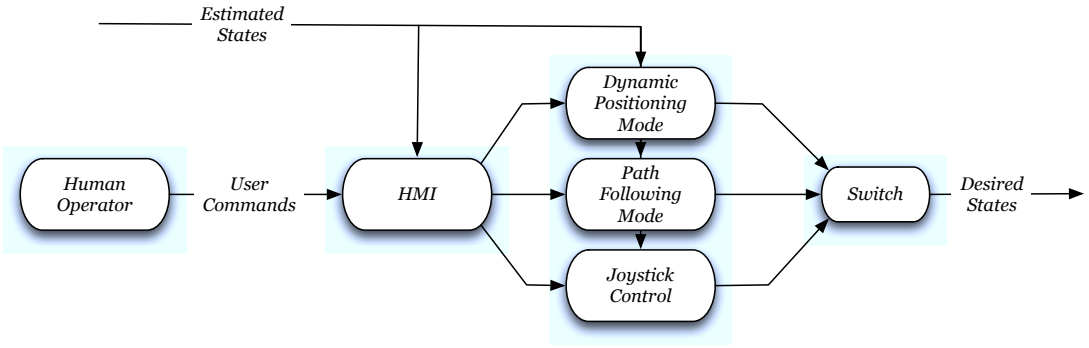


Figure 4.2: The generic principle of the guidance system for ROV Minerva.

is a joystick in the loop control mode, where the joystick generates references in either NED or Body coordinates, and in addition it is possible to generate references in cylinder coordinates. The feature with using cylinder coordinate references gives the possibility to keep the heading towards an object, and keeping the heading towards the object while the ROV is moving on a circle with a defined radius where the object is the origin. This feature is implemented by Fredrik Dukan, and is only included to give a full picture of the system to date.

The simulations and full scale experiments that will be presented in this chapter will be restricted to the horizontal plane. The vertical thrusters are assumed to be independent of the horizontal motion, which has shown to be a valid assumption, although some minor effects will be seen from time to time.

4.1.1 Dynamic Positioning Mode

The DP mode is the standard mode. The user defines the set-point and the guidance system sends this as desired position, while desired velocity and acceleration is equal to zero. The DP system has functionality such that the ROV can move in surge, sway, heave and yaw. The trajectories for movement are generated by a reference model such as described in Section 3. In addition a feedback algorithm for exception handling has been implemented.

Reference Model

A 3rd order position reference model is implemented for dealing with steps in the four different DOFs. The architecture of the reference model is as in (3.7). Velocity and acceleration saturating terms are added, to avoid that the reference model is generating infeasible velocities and accelerations. In the specifications for Minerva given in Table 1.1 there are given maximum velocities for horizontal, lateral and vertical direction in addition to turn rate. In 2004 full scale tests with ROV Minerva were conducted in the towing tank at the Marine Technology Centre at Tyholt, Trondheim. The results from these test are presented in Ludvigsen & Ødegård (2004). The results concerning maximum velocity are given in the 4th column of Table 4.1. The 5th column shows which maximum velocity has been assumed as actual maximum velocity for the implementation. The velocities found during experiments are given higher weight than those in

the specifications. Vertical velocity is adjusted down, mainly because there is a great difference in maximum velocity upwards and downwards due to positive buoyancy.

Direction	Max velocity (Specifications)	Max velocity (Specifiactions)	Max velocity (Tests)	Max velocity (Used)
Forwards	2 knot	1.03 m/s	1.03 m/s	1.03 m/s
Backwards	-	-	0.55 m/s	0.55 m/s
Lateral	1.3 knot	0.67 m/s	0.5 m/s	0.5 m/s
Vertical	1.2 knot	0.62 m/s	-	0.5 m/s
Turn	60 deg/s	60 deg/s	-	60 deg/s

Table 4.1: Maximum velocities for the ROV Minerva.

Feedback Algorithm

In order to achieve robustness for the guidance system it is important to monitor the relationship between the actual state $\boldsymbol{\eta}$ and the desired state $\boldsymbol{\eta}_d$. If it turns out that the ROV is unable to follow the desired trajectory, action needs to be taken in order for the control system to keep control of the ROV.

A simple and intuitive method of achieving the wanted robustness is to design a waiting algorithm. This basically means that if the ROV trails behind by a defined threshold, the reference model is ordered to wait until the ROV is able to catch up with the reference model before starting up again. The value of this threshold is a parameter which should be chosen differently depending on the strictness of the operation. It should also be noted that we only want to do this operation if the ROV lags behind, not if it leads the desired trajectory. The reason for this is that in the case of the ROV leading the desired trajectory it could be a result of the controller, and it would be meaningless to drag the ROV back. The controller will do that job for you, but in the case of the ROV lagging behind, the controller is unable to make the ROV follow the trajectory, and the waiting algorithm should take effect.

The feedback algorithm is implemented by checking the relationship

$$\boldsymbol{\eta}_d - \boldsymbol{\eta} < \boldsymbol{t}_{stop}, \tag{4.1}$$

where $\boldsymbol{t}_{stop}$ is the threshold vector containing the threshold for each DOF. When checking this relationship, the direction of the change must be taken into account, since the sign of (4.1) will be different for negative and positive direction. If (4.1) is violated then the reference model is ordered to stop by changing the input from the reference $\boldsymbol{r}^n$ to the value at the given point. It is also important that the integrators are reset.

In deciding when to start up, a similar relationship as (4.1) is used; only now we want to check if it is within a different threshold of lesser magnitude,

$$\boldsymbol{\eta}_d - \boldsymbol{\eta} > \boldsymbol{t}_{start}, \tag{4.2}$$

where $\boldsymbol{t}_{start} < \boldsymbol{t}_{stop}$ that defines when the ROV is close enough to start up again. So when (4.2) is violated the reference model starts up again. In this work, $\boldsymbol{t}_{start}$ is chosen at a larger value than needed, in order for the reference model to get started. Alternatively one could start up the reference model with the velocity and acceleration at the start up point.

4.1.2 Path Following Mode

The ROV has implemented functionality such that it can follow a predefined path. Both a LOS algorithm for following a piecewise linear path is implemented, in addition to an algorithm for regulating cross-track error by the means of sway motion.

Line of Sight Path Following

A lookahead-based LOS algorithm has been implemented as presented in Section 3.4. The lookahead distance $\triangle$ is a tunable parameter, and speed is set manually. No information about current is currently available in the navigation system for ROV Minerva, which means that the heading assignment is given as

$$\psi_d = \chi_d. \tag{4.3}$$

The switching criteria is implemented based on the along-track error as in (3.34), where the acceptance distance or circle of acceptance defined by R is set to 1 metres.

Cross-track Error Regulation

Based on the principle of steering for straight lines and circular arcs, a direct cross-track regulating algorithm has been implemented. The algorithm takes in an array of way-points, and one single radius for which the arcs are calculated as described in Section 3.3.1. The cross-track error is calculated based on whether the vehicle is located at a straight line or at a circular arc.

The heading assignment is independent of the cross-track error. The desired heading is designed to always be equal to the direction of the path at a any given time. In order to get desired yaw-rate and angular acceleration for yaw, the desired heading signal is going through a 3rd order filter with high bandwidth in order to avoid large time delays.

The algorithm assumes $\psi = \psi_d$ and velocity assignment in sway is given based on the cross-track error. A desired sway velocity towards the path is generated if the cross-track error deviates from zero. A proportional gain on the cross-track error gives a reference to sway velocity. This produces linearly increasing sway velocity with respect to the cross-track error, where saturating elements is added as follows

$$v_d = \begin{cases} K_e e & \text{if } K_e e < 0.5 \\ 0.5 & \text{if } K_e e \geq 0.5 \end{cases} \tag{4.4}$$

where $K_e > 0$ and can be determined based on how large cross-track error to be allowed before giving maximum speed to oppose the cross-track error. A velocity reference model is implemented to get smooth velocity trajectories in sway and surge. The reference model for sway is conservatively tuned in order to limit the aggressiveness of the control law implemented for sway motion in (4.4).

The velocity in surge defines how fast the path should be traversed. Two different approaches will be addressed, with concern to the surge velocity during heading transition phases. In particular when the path consists of circular arcs with a small radius, the surge velocity is critical in order for the ROV to be able to stay on the path on arcs. The most intuitive approach to avoid getting overshoot on the arc is to decrease forward speed while the ROV is on the arc, which will be presented with a simple speed assignment. Another method which has been investigated is to add a lookahead angle when the ROV is on the arc. The lookahead angle on an arc is shown in Figure 3.13 and is calculated as in (3.30).

4.2 Implementation

The control system, including the guidance system, has been implemented in Labview (Labview, 2011) for real-time control of the ROV Minerva. The program code is deployed on both a real-time processor and on the host computer. For real-time processor a compact reconfigurable inputs and outputs (I/O) (cRIO) (ConmpactRIO, 2011) has been used.

4.2.1 Human Machine Interface

Labview has made it very simple to make Human Machine Interfaces (HMI). This has made the system easier to tune, by having all the parameters available for tuning through the HMI. There are also nice and simple possibilities for visualization and graphs in real-time, such that the operator is able to monitor the ROV not only through cameras, but also through horizontal graph, 1 DOF graphs, velocity graphs, thruster RPM graphs and control force graphs. This is convenient during the development phase of such a system. For a user, many of these features should be deleted.

In Figure 4.3 a screen shot of the system is shown. Different modes can be chosen, and parameters can be chosen on-line. The different guidance modes is found in tabs at the top. In Figure 4.4 a screen shot of a real-time visual of the ROV while following a line. Different states can be monitored by choosing in the tab at the top.

Figure 4.3: The guidance system HMI, where a lot of indicators and control inputs can be seen. The yellow LEDs indicates that the corresponding function is on.

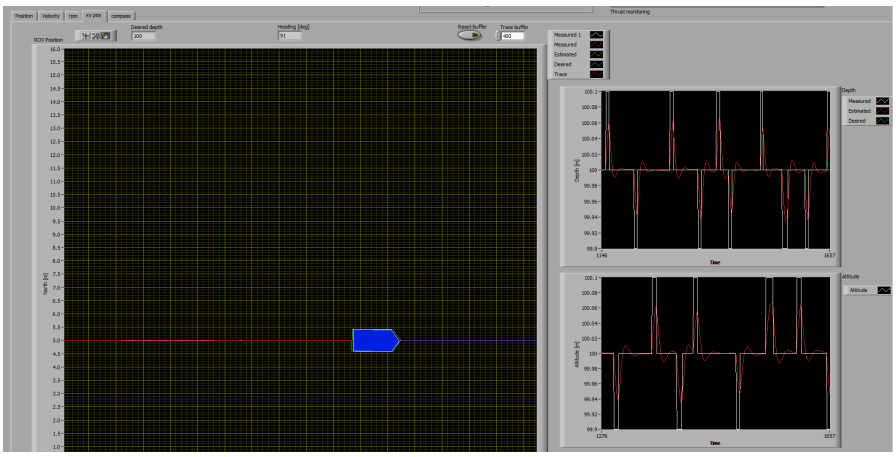


Figure 4.4: The behaviour of the ROV can be monitored through a number of graphs, such as a horizontal graph as shown, position graphs, velocity graphs, rpm graphs and control force graph.

4.3 Simulations of Guidance Strategies

In Section 3.6, several different strategies for following a lawnmower pattern were discussed. The different principles have only been through initial testing during full-scale experiments, so for the reader to get an impression of the drawbacks and advantages of the different principles, will simulations of the different principles be presented in this section. The simulations are done using Matlab/Simulink based on a model of ROV Minerva. The model is for the most part developed by Marianne Kirkeby, and details on the model should be investigated in Kirkeby (2010a) for the interested reader.

In the simulations that are done with reduced forward speed on the arcs, this corresponds to a reduction with a factor of 0.4. The transit speed during the simulations is 0.3 m/s, which means that, when reduced, the speed is 0.12 m/s. The reduction in speed is added to be able to follow a circular arc with radius of 2 metres. All the simulations are run through a lawnmower pattern where the way-points are given by

$$W = [(0,0),(0,30),(4,30),(4,0),(8,0),(8,30)]. \tag{4.5}$$

For the scenarios where there are circular arcs involved, the radius of the arcs is defined as $R = 2$. The cross-track error gain $K_e = 0.3$ for all the scenarios, which is quite conservative. The simulations will be based on the numbers used for the different approaches in Section 3.6.1.

Thrust force and thrust revolution speed will be discussed, to compare the load on the thrusters during the different scenarios. Mainly the horizontal thrusters will be considered, that is starboard, port and lateral thruster. The thrust force and effect is calculated from (3.44) and (3.46) respectively. The total energy consumption is numerically calculated based on (3.47).

As was mentioned in Section 3.6, the torque coefficient is currently not available. The coefficient is needed to calculate energy, which has required that a value for K_{Q0} has been assumed. Based on common open-water tests for propellers, and the resulting advance ratio J versus K_T and $10K_Q$, the torque coefficient should be a little higher than $0.1K_{T0}$, and will for the remainder be assumed to be around $K_{Q0} = 0.03$. As the thrust coefficient is different for positive and negative thrust it is assumed the same for the torque coefficient. The torque coefficients will be used as follows:

$$K_{Q0_{forwards}} = 0.03, \tag{4.6}$$

$$K_{Q0_{backwards}} = 0.03 \cdot \left(\frac{0.15}{0.24}\right) = 0.0187. \tag{4.7}$$

Since the energy calculations are based on estimated torque coefficients, the energy will not be correct, but should give an estimate of the actual desired energy. Since the coefficients are used consequently, the relative energy consumption from the different scenarios should give a good indication of energy efficiency relationship between the different scenarios.

4.3.1 Simulation of ①

This scenario involves stopping at each way-point to change heading, before accelerating onto the new line. The ROV in the horizontal plane can be seen in Figure 4.5, the cross-track error and surge velocity can be seen in Figure 4.6 and the commanded thrust is presented in Figure 4.7.

The ROV can be seen to follow the lines closely at all times, as intended. Ideally a position controller should take over during the heading changes, but for saving a lot of time programming logics, and the lack of current there is no position control during the stationary parts of the path. This is the reason for the small drift around the way-points, which can be seen in the cross-track error plot.

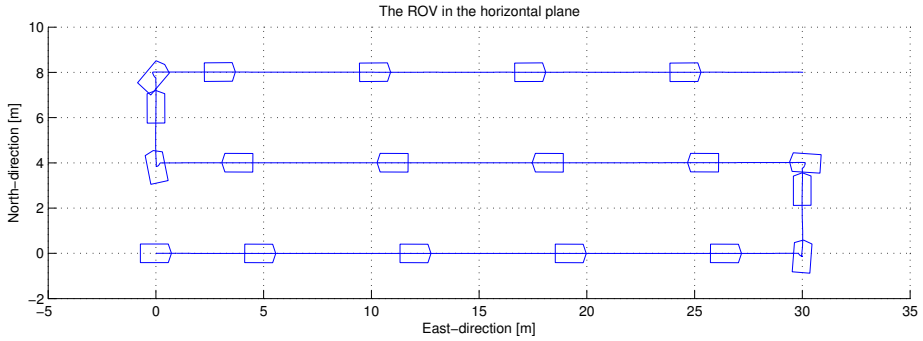


Figure 4.5: The ROV stops at each way-point, changes heading, and accelerate onto the next line.

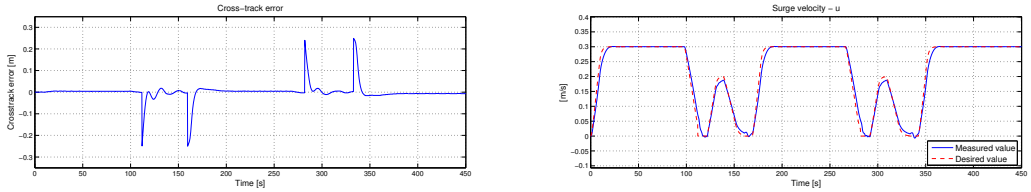


Figure 4.6: The cross-track error increases at the way-point due to a considered circle of acceptance of 0.25 metres. The surge velocity is not able to reach the maximum velocity of 0.3 m/s on the short lines.

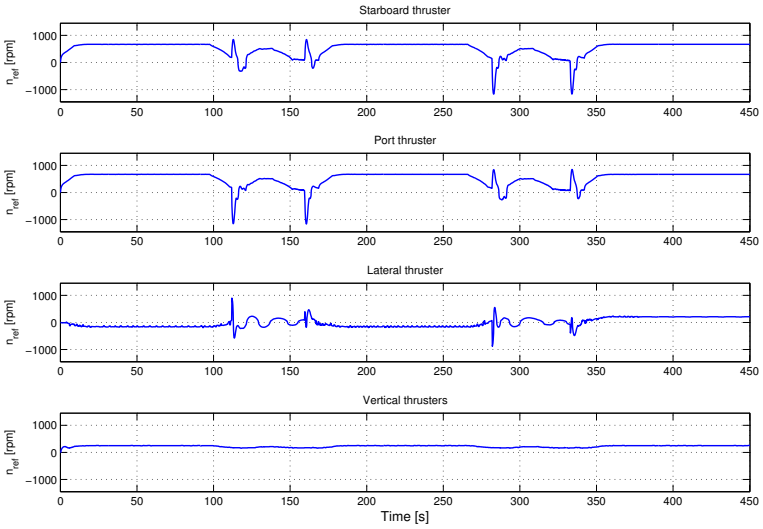


Figure 4.7: The thrust gets quite big variation during acceleration phases for surge velocity and heading.

It can be seen from Figure 4.7 and Table 4.2 that the the thrust input to the horizontal thrusters have high variation, but a quite conservative mean value. Due to the fact that the ROV is commanded to stop, it will get high negative thrust on the forward thrusters. The lateral thrusters get high thrust during the turning over a short period of time. The total energy consumption is heavily dominated by the forward thrusters. The lateral thruster will have almost no thrust during transit between points, and just a short period of thrust during the turning manoeuvre.

Time:	450.2 seconds			
Tot Energy	19.0 Wh			
Thruster:	Starboard	Port	Lateral	Vertical
Min RPM	-1166.8	-1165.9	-882.8	0
Max RPM	848.8	849.9	898.9	255.3
Mean RPM	548.8	548.5	154.5	223.7
Min Force [N]	-72.2	-72.1	-72.2	0
Max Force [N]	83.0	83.2	74.8	6.1
Mean Force [N]	38.3	38.3	2.7	4.8
Max Effect [W]	242.6	242.0	348.2	4.5
Mean Effect [W]	70.8	70.8	3.6	3.2
Energy [Wh]	8.9	8.9	0.5	0.4

Table 4.2: Overview of the RPMs and forces demanded of the thruster during simulation of Scenario ①.

4.3.2 Simulation of ②

This scenario involves cutting the corners in order to achieve a smooth switch between the lines in the pattern. A scenario where the forward speed is kept constant will be presented first, and then a scenario where the forward speed is reduced on the arc will be presented. The ROV can be seen traversing the path in Figure 4.8, while the cross-track error and surge velocity is shown in Figure 4.9.

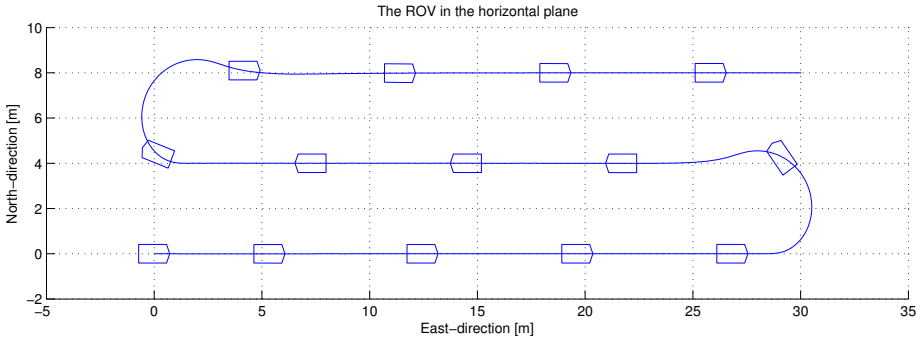


Figure 4.8: The ROV will drift of the circle due to the constant speed through the arcs.

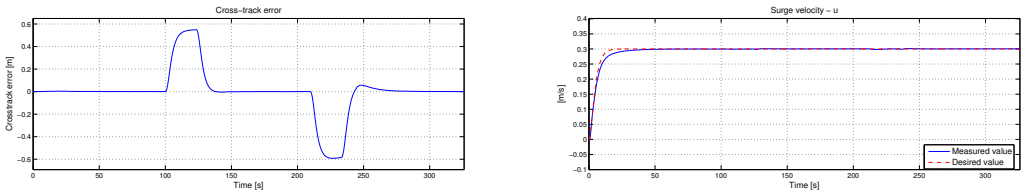


Figure 4.9: The cross-track demonstrates the drift through the arc, while the surge speed is constant.

Time:	326 seconds			
Tot Energy	23.4 Wh			
Thruster:	Starboard	Port	Lateral	Vertical
Min RPM	0	0	-899.7	0
Max RPM	727.6	728.9	1028.8	273.0
Mean RPM	656.0	655.8	272.8	248.8
Min Force [N]	0	0	-75.0	0
Max Force [N]	61.0	61.2	98.0	6.9
Mean Force [N]	49.9	49.9	14.0	5.8
Max Effect [W]	127.8	128.5	522.0	5.4
Mean Effect [W]	95.2	95.1	59.2	4.2
Energy [Wh]	8.6	8.6	5.4	0.4

Table 4.3: Overview of the RPMs and forces demanded of the thruster during simulation of Scenario ② with constant speed.

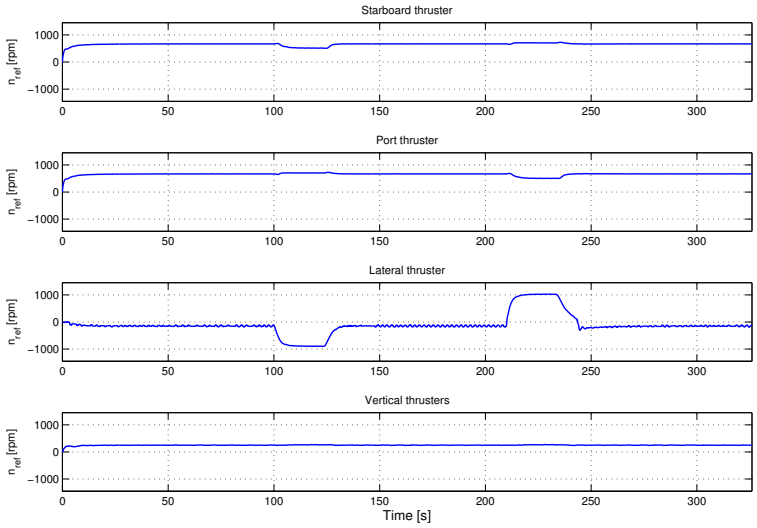


Figure 4.10: Most notably, the lateral thruster has to work quite hard during the turns in order to keep the heading in the direction of the path.

Due to a constant surge velocity, the thrust command to starboard and port thruster will always be positive, and approximately constant during the entire run as can be seen in Figure 4.10. In Table 4.3 it can be seen that since the surge velocity is kept constant, the forward thrust is quite stable, but the lateral thrusters are varying more due to the overshoots which requires sideways thrust to drag the ROV back to the line. This can also be seen by looking at energy consumed by each thruster, where the lateral thruster is using more energy now, than for the previous simulation.

The same simulation is now run, only now the surge velocity is reduced on the circular arcs. This will make the ROV stay closer to the path while on the arc, but will obviously take longer time. The ROV can be seen in Figure 4.11, and it is clearly following the arc more closely. This is confirmed by taking a closer look at the cross-track error in Figure 4.12. Here can it also be seen that the surge velocity is varying between 0.3 m/s while on the line, and 0.12 m/s while on the arc.

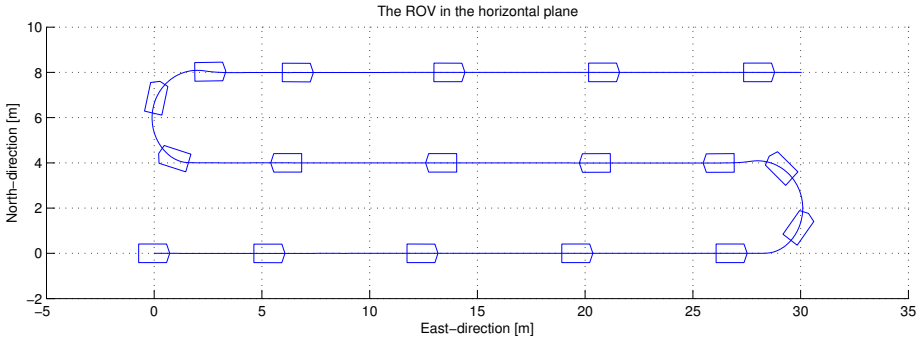


Figure 4.11: The ROV is now following the arcs more closely, due to the reduced forward speed.

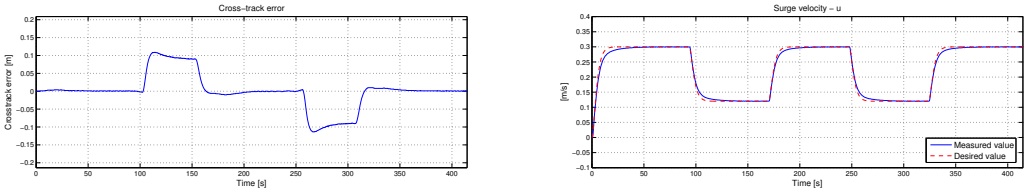


Figure 4.12: The cross-track error has the same behaviour as in Figure 4.9, but at a lower magnitude due to the reduced speed in surge velocity on the arcs.

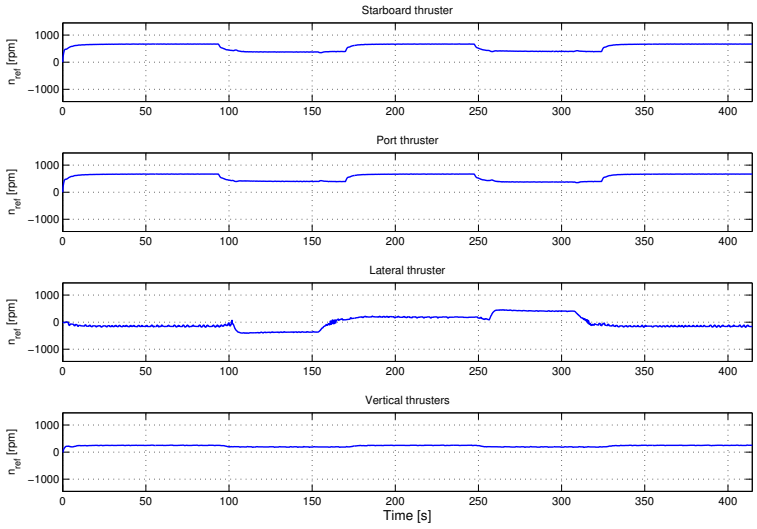


Figure 4.13: Due to the lower forward speed on arcs, all the horizontal thruster have reduced thrust commands during turning.

As expected the thrust commands are more conservative now, than for using constant surge velocity as is demonstrated in Figure 4.13 and Table 4.4. Especially for the lateral thruster the maximum and minimum values are much lower. This is on expense of time. The time using constant speed is 326 seconds, while with the reduced speed the ROV takes 414.6 seconds to finish the path. If the mean forces is compared to the constant speed scenario, it can be seen that the mean force is decreased with 10 Newton for the horizontal thrusters when reducing the velocity on the arcs. The total energy consumption is reduced by almost 35 percent by only reducing the surge velocity on the arcs. This is not mainly because the forward thrusters are

Time:	414.6 seconds			
Tot Energy	17.4 Wh			
Thruster:	Starboard	Port	Lateral	Vertical
Min RPM	0	0	-409.250	0
Max RPM	672.7	674.1	455.4	258.7
Mean RPM	562.2	562.3	209.3	225.3
Min Force [N]	0	0	-15.5	0
Max Force [N]	52.2	52.4	19.2	6.2
Mean Force [N]	38.3	38.3	5.2	4.8
Max Effect [W]	101.0	101.7	45.3	4.6
Mean Effect [W]	67.7	67.7	8.8	3.2
Energy [Wh]	7.8	7.8	1.0	0.4

Table 4.4: Overview of the RPMs and forces demanded of the thruster during simulation of Scenario ② with reduced speed on arcs.

working less, because the energy consumed by the forward thrusters is only reduced by 0.8 Wh. It is due to the fact that the ROV is able to stay closer to the path, which means that the lateral thruster is not used for regulating the cross-track as heavily as for the previous simulation.

4.3.3 Simulation of ③

This scenario involves letting the vehicle overshoot by defining the turn radius as 0, and keeping constant velocity through the way-points. This is a special case of ① only now the surge velocity is kept constant during the turn. This is expected to be faster than ①, but will have bigger deviation with respect to the control objective.

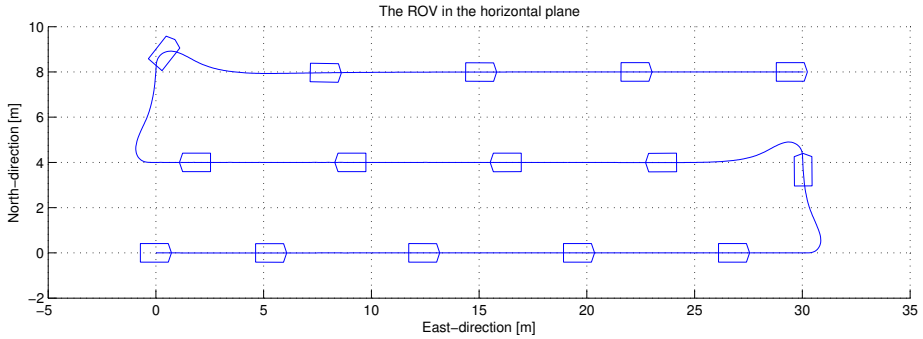


Figure 4.14: The ROV is clearly overshooting at the corners of the pattern due to neglecting the infeasibility of the manoeuvre.

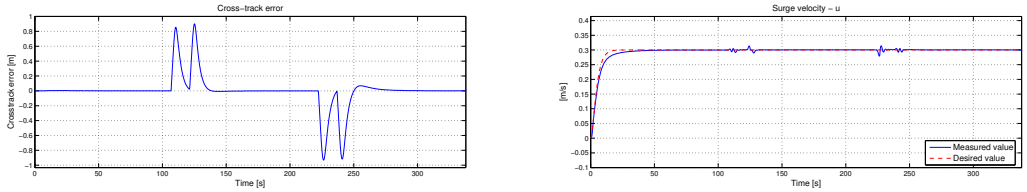


Figure 4.15: The cross-track error is drifting when the ROV is overshooting as expected, since the surge velocity is kept constant during a 90 degree turn.

The trajectory of the ROV can be seen in Figure 4.14 while the cross-track error and surge velocity can be sen in Figure 4.15. The ROV can be seen overshooting the way-points by almost 1 metre, and needs approximately 4 metres to stabilize towards the next line. Due to the conservative gain with respect to the path following control objective, oscillations are avoided but as a drawback it takes longer time to converge to the each new line.

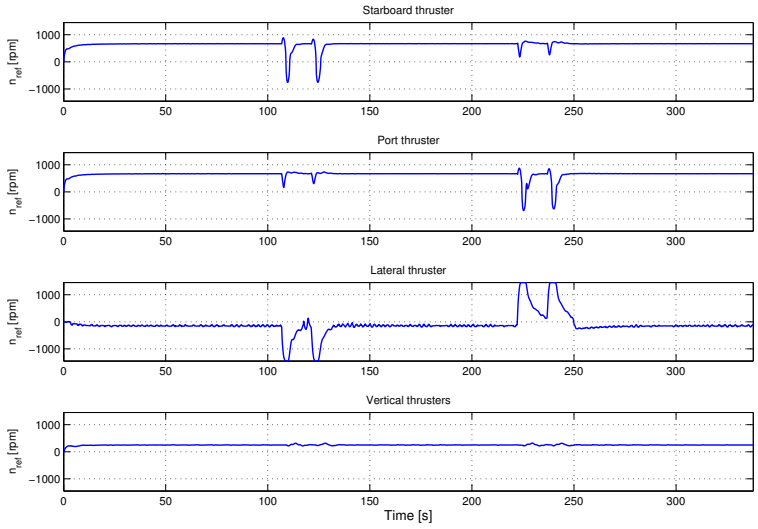


Figure 4.16: Even though the lateral thruster is assisted by the starboard and port thruster during turning, saturation for the lateral thruster is unavoidable.

Time:	337.8 seconds			
Tot Energy	25.4 Wh			
Thruster:	Starboard	Port	Lateral	Vertical
Min RPM	-758.4	-692.1	-1450	0
Max RPM	889.7	876.2	1450	319.9
Mean RPM	657.4	655.0	238.2	248.0
Min Force [N]	-30.5	-25.4	-194.7	0
Max Force [N]	91.2	88.5	194.7	9.5
Mean Force [N]	50.0	49.8	12.6	5.7
Max Effect [W]	233.7	223.3	1461.6	8.8
Mean Effect [W]	96.2	95.7	70.3	4.1
Energy [Wh]	9.0	9.0	6.6	0.4

Table 4.5: Overview of the RPMs and forces demanded of the thruster during simulation of Scenario ③.

The thrust commands can be seen in Figure 4.16, and from the summary of thrust commands in RPM and Newton in Table 4.5. The lateral thruster is saturated two times between 100 and 150 seconds and two times between 200 and 250 seconds. This is due to the overshoots while changing heading, which makes the system both try to change heading and force the ROV back to the path. This is most undesirable, but should be expected when this type of strategy is used. Compared to the strategy ① it can be seen that by using this strategy it takes over 100 seconds shorter time to complete the path, but the energy consumption has increased by 34 percent.

4.3.4 Simulation of ④

This scenario involves making the vehicle take a turn outside the way-point in order to approach the way-point, first with the heading corresponding to the line approaching the way-point, and secondly with heading corresponding to the line leaving the way-point. A scenario where the forward speed is kept constant will be presented first, and then a scenario where the forward speed is reduced on the arc will be presented. This strategy is expected to be the most time consuming strategy due to the fact that the path is extended by $(3/4)2\pi r + 4 \approx 13$ metres at each way-point, which is a 46 % increase of the original path.

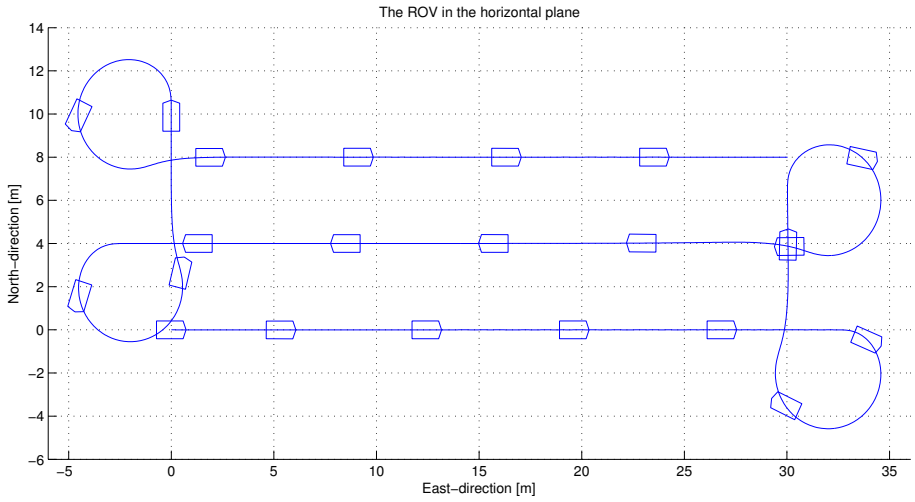


Figure 4.17: Due to the constant forward speed, the ROV is drifting during the turns, as previously seen in Figure 4.8.

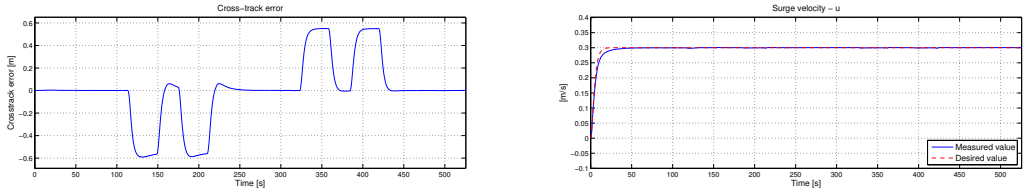


Figure 4.18: The drifting during turning is seen from the cross-track error, while the surge velocity is kept constant during the run.

The trajectory of the ROV when applying constant speed can be seen in Figure 4.17, while the cross-track error and surge velocity is presented in Figure 4.18. The forward speed is to high for the ROV to stay on the path during the turns, but is almost able to drag itself onto the path before entering the lines after a turn. The cross-track error demonstrates that the ROV is drifting during the turns, but is converging nicely during the lines.

The mean forces are generally high, due to constant forward speed as shown in Table 4.6, which demands a lot of thrust from the lateral thruster during turning. The data is quite comparable to what was experienced in strategy ② with constant speed, although now the turns are longer, which demands more from the lateral thruster. This strategy is the most energy consuming strategy so far, over double the energy consumed by some of the previous simulations. To some extent this is expected, due to the longer path, but this should be reflected by increased energy consumption for the forward thruster. The fact is, that it is the energy consumption by the lateral thruster which stands for the highest increase. Based on the results seen in ②, this

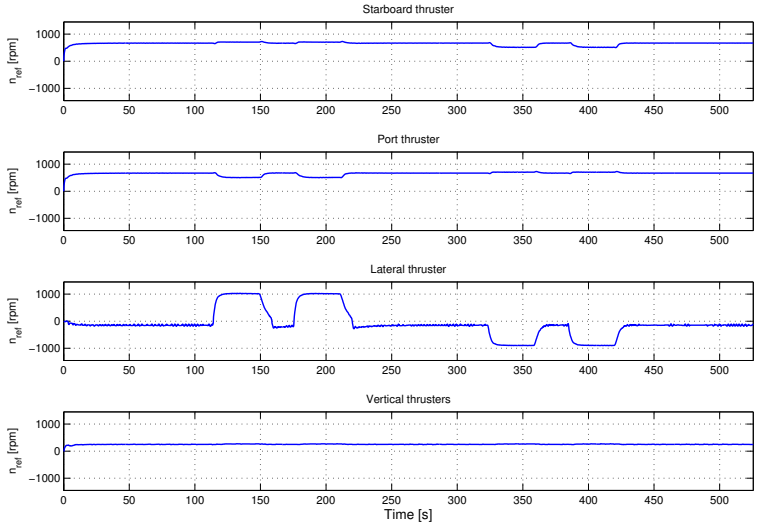


Figure 4.19: The thrust commands are stable for starboard and port, while the lateral thrust commands increases significantly during turns.

should be reduced dramatically in the next simulation, when forward speed is reduced on the arcs.

Time:	525.6 seconds			
Tot Energy	44.7 Wh			
Thruster:	Starboard	Port	Lateral	Vertical
Min RPM	0	0	-902.7	0
Max RPM	728.6	728.9	1027.3	274.4
Mean RPM	650.9	650.5	375.1	251.4
Min Force [N]	0	0	-75.5	0
Max Force [N]	61.2	61.2	97.7	7.0
Mean Force [N]	49.2	49.2	24.2	5.9
Max Effect [W]	128.4	128.5	519.7	5.5
Mean Effect [W]	93.6	93.5	110.1	4.3
Energy [Wh]	13.7	13.7	16.8	0.6

Table 4.6: Overview of the RPMs and forces demanded of the thruster during simulation of Scenario ④ with constant speed.

To be able to follow the arcs more closely, the forward speed is now reduced to 0.12 m/s while on the arcs. The ROV following the path can be seen in Figure 4.20, the cross-track error and surge velocity is presented in Figure 4.21. The ROV can be seen to follow the arcs more closely now. This is backed up by looking at the cross-track error which is significantly reduced. This is due to the fact that the ROV is traversing at 0.12 m/s as much as at 0.3 m/s, which on the downside obviously will affect the time for completing the path.

The thrust command is now more conservative, especially for the lateral thruster, as can be seen in Figure 4.22. Summary of thrust use is presented in Table 4.7, where it can be seen that the force is massively reduced compared to applying constant forward speed. This is on the expense of time to completion, which has increased from 525.6 seconds to 763.8 seconds, which is almost a 4 minutes increase. Despite the time to complete the path is significantly increased, the

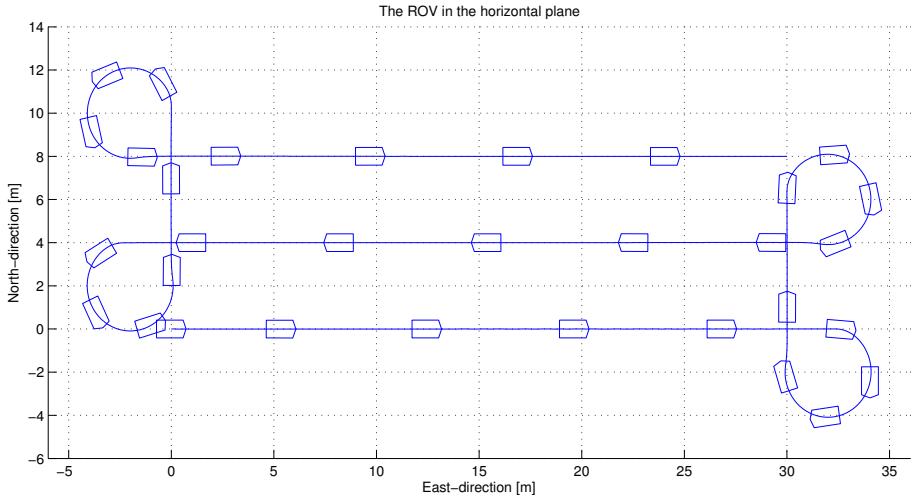


Figure 4.20: The ROV is now, in addition to following the lines, also able to follow the circular arcs closely.

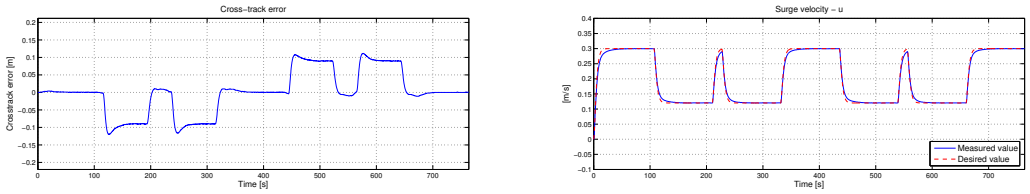


Figure 4.21: The cross-track error is reduced significantly because of the reduction in forward speed on the arcs as can be seen by the surge velocity.

energy consumption is reduced by a massively 63 percent. The energy consumption is actually only 2 Wh higher than for the strategy in , despite the path being 46 percent longer.

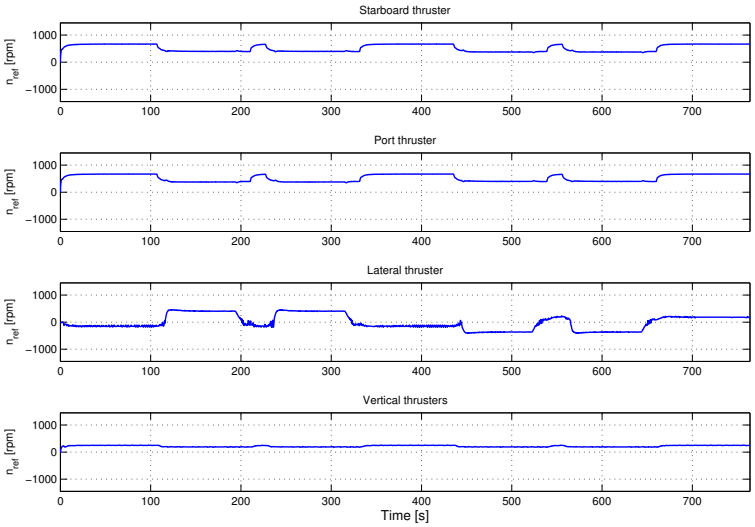


Figure 4.22: The thrust commands are reduced with respect to using constant velocity during the turns, this is especially apparent for the lateral thrust commands.

Time:	763.8 seconds			
Tot Energy	27.053 Wh			
Thruster:	Starboard	Port	Lateral	Vertical
Min RPM	0	0	-406.7	0
Max RPM	672.8	673.2	460.5	259.4
Mean RPM	515.1	515.0	246.7	215.8
Min Force [N]	0	0	-15.3	0
Max Force [N]	52.2	52.2	19.6	6.2
Mean Force [N]	32.6	32.6	7.1	4.4
Max Effect [W]	101.1	101.3	46.8	4.7
Mean Effect [W]	54.4	54.4	13.0	2.8
Energy [Wh]	11.5	11.5	2.8	0.6

Table 4.7: Overview of the RPMs and forces demanded of the thruster during simulation of Scenario ④ with reduced speed on arcs.

4.3.5 Summary of Guidance Strategies

All four strategies have been simulated, and ② and ④ have also been simulated for both constant forward speed and with reduced forward speed on arcs. Table 4.8 summarizes some of the important factors when evaluating the different strategies.

Evaluation is not straight forward, since different missions have different priorities. It can, however, quickly be seen that if the lines have to be covered entirely, strategy ① will always be more efficient than strategy ④, both time efficiently and energy efficiently. Even the cross-track error will give the same conclusion, since the deviation of 0.25 metres is due to the lack of position control during heading change for strategy ①. Although ① theoretically is more efficient than ④, it is not definite, since for an AUV ① in most cases will not be feasible. So for an AUV that has to cover the lines entirely, strategy ④ CS is the most time efficient while ④ RS is the most energy efficient. If restriction on cross-track error is loosened between the horizontal lines, then ② CS is the most time efficient, being slightly faster than ③ but being more energy efficient and also gives tighter control as can be seen from the cross-track error.

In general it can be said that ①, ② RS and ④ RS are both more energy efficient and tighter with respect to the control objective, than their counterparts. If time is an important factor then ② CS, ③ and ④ CS are more efficient than their counterparts. Note that ① and ③ are considered counterparts. It has been clear that when applying to high forward velocity when travelling along arcs, the energy consumption is increasing drastically, since the ROV will drift which will make the lateral thruster consume a lot of energy bringing the vehicle back to the path. It is therefore important to consider the forward speed and radius of the arcs, when such strategies are applied.

Strategy	Time [s]	Energy [Wh]	Max CT error [m]	Mean CT error [m]
①	450.2	19.0	0.2500	0.0138
② CS	326.0	23.4	0.5915	0.0840
② RS	414.6	17.4	0.1138	0.0260
③	337.8	25.4	0.9320	0.0709
④ CS	525.6	44.7	0.5909	0.1526
④ RS	763.8	27.1	0.1201	0.0409

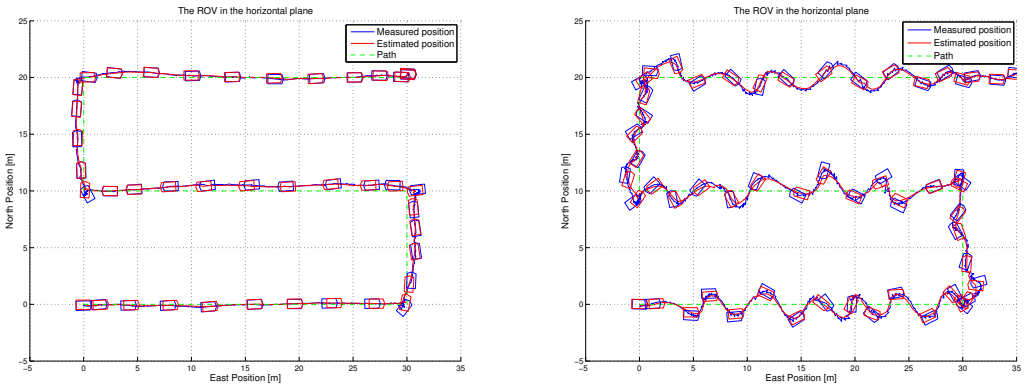
Table 4.8: Summary of the different strategies, with time consumption, energy consumption and mean cross-track error. CS and RS is short for constant forward speed and reduced forward speed on arcs CT is short for cross-track.

4.4 Full Scale Experiments with ROV Minerva

A handful of one day cruises has been carried through this winter and spring, where full scale experiments with the ROV Minerva has been conducted through the AUR-Lab. The missions have been done with the RV Gunnerus as the base for deployment. Experiments have been done at several different depths, but at no depths deeper than 350 metres. It should be noted that the experiments have been done at constant depth. The results which will be presented in this section are an assortment of what have been done on the guidance system through the period of December 2010 to May 2011. The system has been configured as was shown initially in this thesis in Figure 1.6.

4.4.1 Line of Sight Path Following

A LOS guidance scheme for path following has been implemented. Lawnmower pattern trials have been conducted with the LOS guidance scheme, where in particular it is interesting to see how different lookahead distances affect the overall performance of the algorithm. Figure 4.23 present to scenarios of following a lawnmower pattern. Figure 4.23b presents the results when using the lookahead distance $\triangle = 1$ metres, from which it can be concluded that 1 metre is too short to get a smooth convergence towards the lines. When the lookahead distance is too short, once the cross-track error increase the ROV is trying to counteract by changing the heading towards the path only one metre ahead, which propagates into heavy oscillations and instability. This observation was duly noted and the lookahead distance was increased to $\triangle = 5$ metres, and the result can be seen in Figure 4.23a. The ROV is now able to make progress on the path



(a) Lookahead distance of 5 metres.

(b) Lookahead distance of 1 metres.

Figure 4.23: The ROV Minerva following a lawnmower pattern using a Line of Sight path following algorithm.

without oscillations, with the drawback that it takes the ROV longer time to correct when it finds itself outside the path. The principle can be compared to driving a car, where in order to follow the road smoothly it is better to look forward on the road than to look on the road only a couple of metres in front of the car. When switching way-points it takes the ROV some time to get back to the new line, which indicates that this approach is better suited for long distances between way-points than such scenarios as lawnmower patterns with small distances between the lines. The cross-track error of the successful test is shown in Figure 4.24, which stays within ± 0.5 metres on the lines, while it peaks at 1 metres due to the acceptance distance of 1 metres.

Figure 4.24: The cross-track error stays within ± 0.5 metres, while since the acceptance distance is 1 metre, the cross-track error will peak to this distance at the time where this switch is made.

4.4.2 Square Tests

Several tests have been done in order to investigate how the forward speed affects the capability to stay on arcs of different radius. Since time has been limited for full scale tests, and several tests have been distorted by occasional bad HiPaP and DVL readings, there has not been conducted a structured investigation of the speed versus radius correlation. Despite this, the following results will clarify and give some insight to the relationship between the radius and the forward speed and the different approaches that have been investigated. The maximum forward speed for arc with radius of 2, 3 and 5 metres will be investigated, and an estimated maximum forward speed will be found. It should be noted that the performance of the ROV during these tests is dependent on the quality of the signals the system receives, and especially the heading measurements is critical. For all these tests, the fluxgate compass is the main sensor for heading. In addition, the quality of the signals from HiPaP and DVL can vary, so the maximum forward speed can not be determined exactly, but within some range.

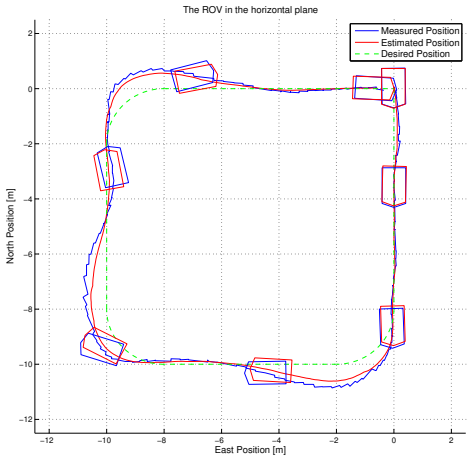
Arc Radius of 2 Metres

For survey and mapping missions it is of interest to keep the turning arc as small as possible. Results where an arc with radius of 2 metres is considered will be presented. In Figure 4.25a the result of the square test with constant speed of 0.25 m/s, and none action taken to avoid the overshoot which is expected. It is obvious that the ROV is not able to follow the arc at the desired speed. In Figure 4.25b a lookahead distance of $\triangle = 0.7$ is used on the arc to add a lookahead angle on the arc, which as can be seen shows promising behaviour. The overshoot is reduced which was the objective, but not in that extent which was hoped for.

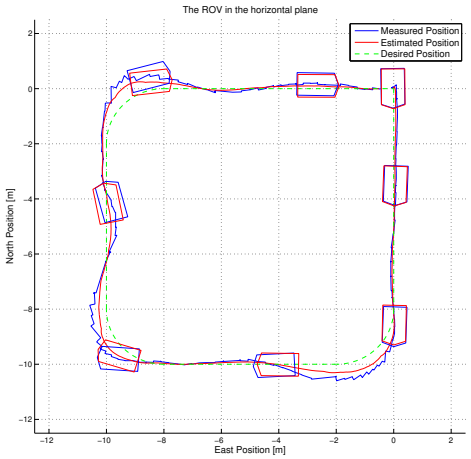
The approach of using a lookahead angle on the arcs has been experienced to decrease the cross-track error, but will provoke oscillations which gives undesirable behaviour, especially for low speeds. Oscillations will usually be provoked if the lookahead distance is short with respect to forward speed, which actually is quite intuitive, since the ROV will have longer time to change heading towards a point a shorter distance ahead. As can be seen in Figure 4.26 where the cross-track error for the two scenarios is shown, there is no sign of provoked oscillations. The cross-track error is reduced when a lookahead angle is applied, but since it should not be to tightly tuned, this approach has a limited application base.

The surge velocity is constant at 0.25 m/s for both approaches as can be seen in Figure 4.27, where it is the desired velocity that is 0.25 m/s. The measured velocity can be seen to oscillate somewhat.

A different approach to the line of sight approach is to decrease the velocity on the arcs. Figure 4.28a presents the result of the same square test, but this time the forward speed is 0.25

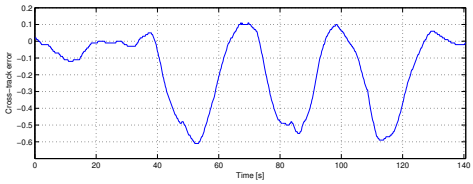


(a) Surge velocity $u = 0.25$ m/s

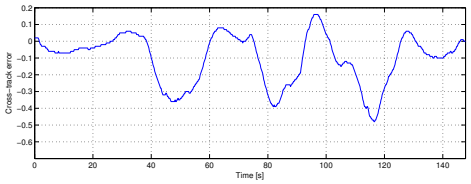


(b) Surge velocity $u = 0.25$ m/s w/ lookahead angle on arcs

Figure 4.25: Square tests with radius of 2 metres. In (a) no action is taken to avoid overshoot, while in (b) a lookahead approach is taken on the arcs.

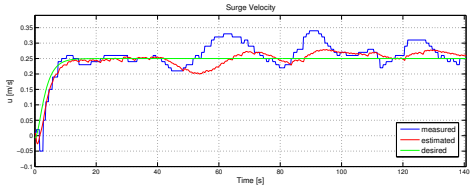


(a) Surge velocity $u = 0.25$ m/s

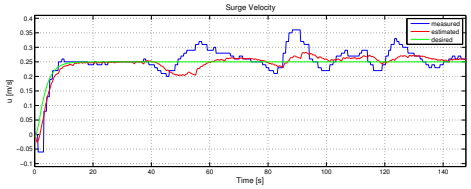


(b) Surge velocity $u = 0.25$ m/s w/ lookahead angle on arcs

Figure 4.26: The cross-track error in (a) is smoother, but is generally higher than in (b).



(a) Surge velocity $u = 0.25$ m/s

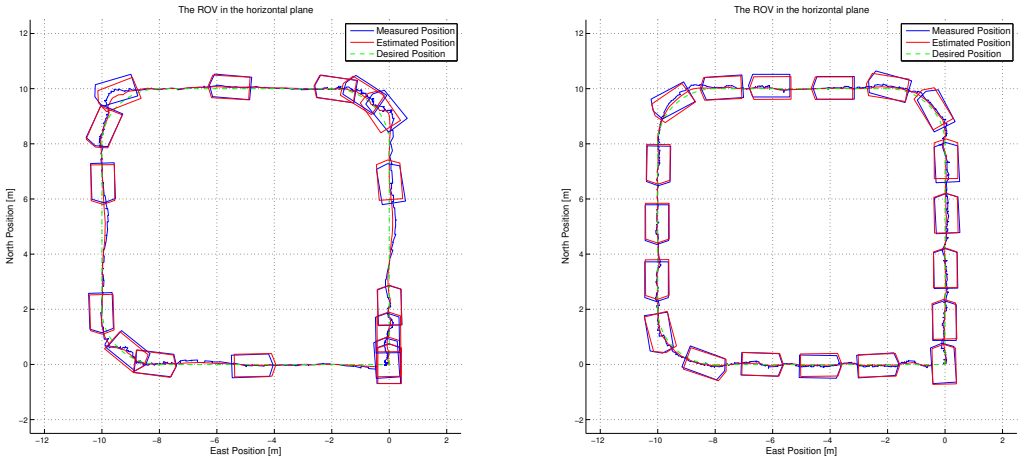


(b) Surge velocity $u = 0.25$ m/s w/ lookahead angle on arcs

Figure 4.27: Surge velocity is constant at 0.25 m/s for both experiments.

m/s on the lines while it is decreased to 0.05 m/s when the ROV is on the arc. As can be seen, major overshoots are avoided while no oscillations are provoked. To decrease the velocity to 0.05 m/s seems a bit drastic, which the result in Figure 4.28b confirms. In this test, a constant forward speed of 0.1 m/s was used and the performance is actually better than when the speed is decreased to 0.05 m/s on the arcs.

The cross-track error is shown in Figure 4.30a, where it can be seen that the peaks are drastically decreased with respect to the results presented in Figure 4.25, which also makes the ROV have very little cross-track error when reaching the straight lines. The cross-track error in Figure 4.30b, however, confirms that the performance is higher when a constant forward speed

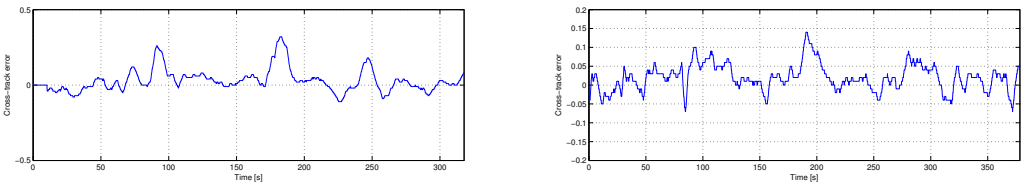


(a) Forward speed of 0.25 m/s, which is reduced to 0.05 m/s on the arcs.

(b) Constant forward speed of 0.1 m/s.

Figure 4.28: Square test with radius of 2 metres, where in (a) the velocity is heavily reduced on the arcs while in (b) the velocity is constantly kept low.

of 0.1 m/s is used. The reason that the speed assignment is less efficient is most probably that the ROV is not decreasing the speed until reaching the arc, such that the forward speed is actually not fully decreased until some way onto the arc.

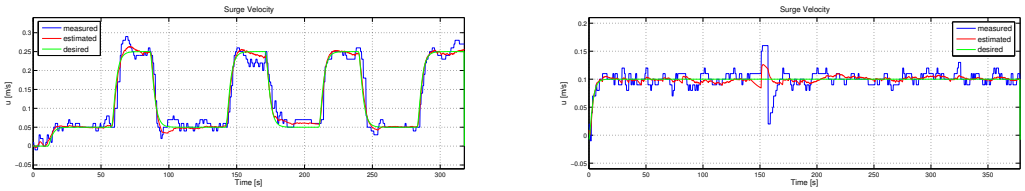


(a) Forward speed of 0.25 m/s, which is reduced to 0.05 m/s on the arcs.

(b) Constant forward speed of 0.1 m/s.

Figure 4.29: The cross-track error is heavily reduced, compared to Figure 4.26, due to the low speed on the arcs.

The surge velocity can be seen to increase and decrease between 0.25 and 0.05 m/s for Figure 4.30a, while in Figure 4.30b the velocity is constant at 0.25 m/s. Complete results for these two experiments can be found in Appendix A.1.



(a) Forward speed of 0.25 m/s, which is reduced to 0.05 m/s on the arcs.

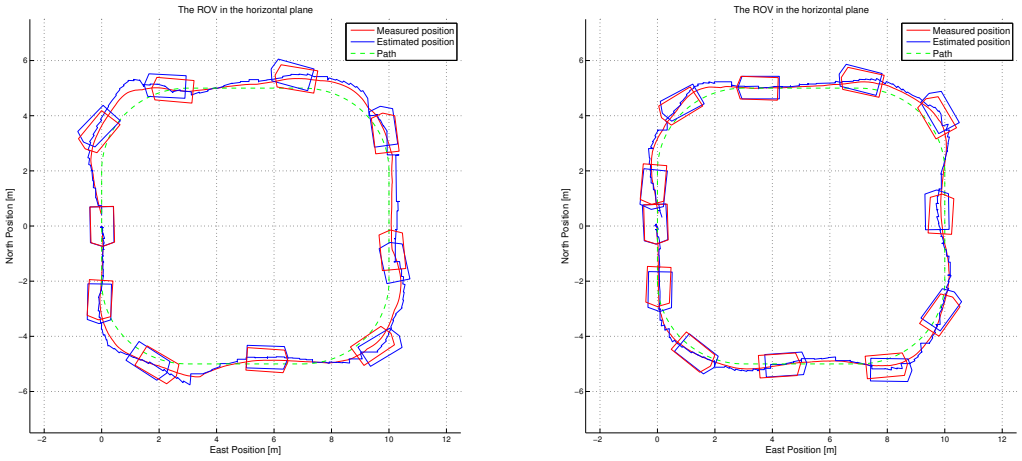
(b) Constant forward speed of 0.1 m/s.

Figure 4.30: The surge velocity is reduced to 0.05 m/s on the arcs for the first experiment, while it is kept constant for the second experiment.

An algorithm to calculate the deceleration such that the forward speed has reached the desired arc speed when entering the arc should be implemented. The maximum forward speed for ROV Minerva when moving on an arc with radius of 2 metres should be somewhere between 0.1 and 0.15 m/s.

Arc Radius of 3 Metres

The lookahead has now been abandoned, and the radius is increased to 3 metres. In the previous example, the speed was reduced to 0.05 m/s, which is a drastic change. Considering that the radius now is increased, a test was done where the speed is reduced to 0.2 m/s on the arcs. The results can be investigated in Figure 4.31 where Figure 4.31a shows the ROV when applying constant velocity, while Figure 4.31b shows the ROV when the velocity is reduced by 0.05 m/s on the arcs.

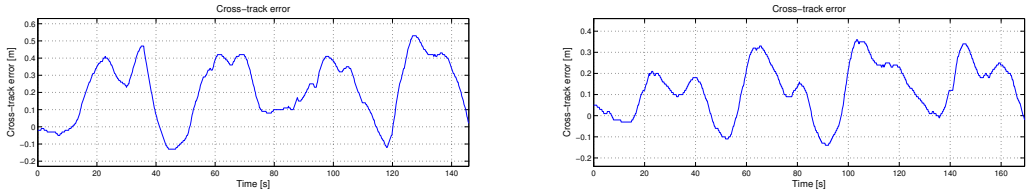


(a) Surge velocity $u = 0.25$ m/s

(b) Surge velocity $u = 0.25$ m/s w/ reduced velocity on arcs

Figure 4.31: In (a) the velocity is kept constant, while in (b) the velocity is reduced with 0.05 m/s on the arcs.

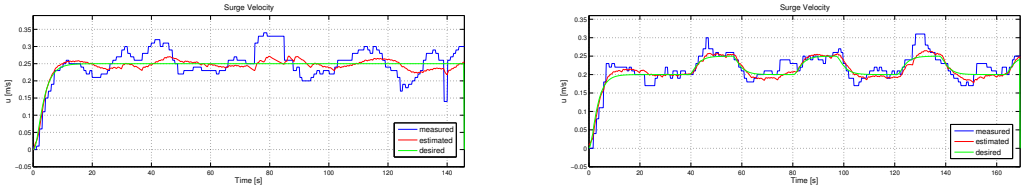
As expected, the performance is increased when the velocity is reduced. On the other hand, it seems obvious that the velocity should be decreased further to eliminate the overshoots and oscillations. The cross-track error in Figure 4.32 shows that when the velocity is reduced the ROV is able to follow the path more closely, but further improvement with the speed assignment is required for the ROV to be able to follow it as close as desired. The results does not give a



(a) Surge velocity $u = 0.25$ m/s

(b) Surge velocity $u = 0.25$ m/s w/ reduced velocity on arcs

Figure 4.32: The cross-track error is reduced when the speed assignment is used. Note that the axis in (a) and (b) are different.



(a) Surge velocity $u = 0.25$ m/s

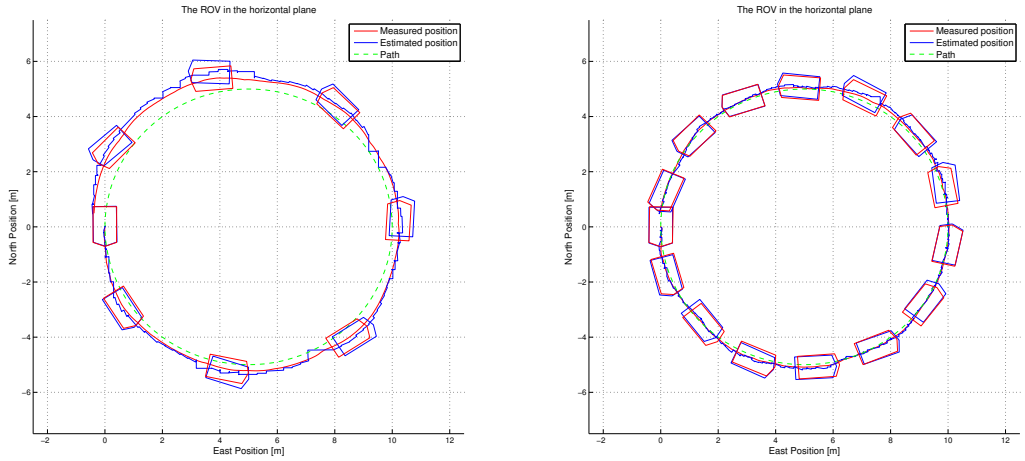
(b) Surge velocity $u = 0.25$ m/s w/ reduced velocity on arcs

Figure 4.33: The surge velocity is kept constant at 0.25 m/s for both experiments.

clear indication of what the maximum velocity is, but based on the results for the arc with radius of 2 metres and the arc with radius of 3 metres, it should be somewhere between 0.15 and 0.2 m/s.

Arc Radius of 5 Metres

The final square tests that will be presented are of squares with arcs of radius 5 metres, which in practice means that it is a circle with radius of 5 metres. The tests are conducted with constant speed, where the ROVs performance can be seen in Figure 4.34, where the speed is 0.3 and 0.15 m/s respectively for Figure 4.34a and 4.34b.



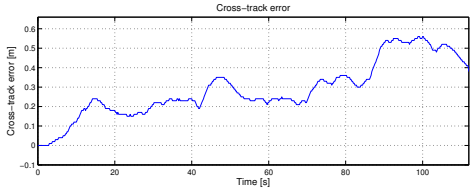
(a) Surge velocity $u = 0.3$ m/s

(b) Surge velocity $u = 0.15$ m/s

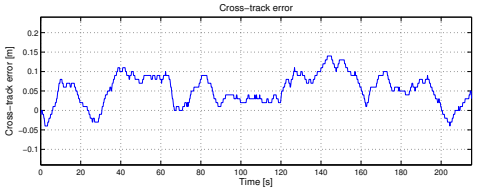
Figure 4.34: Maximum forward speed for an arc with radius of 5 metres is probably between 0.15 and 0.3 m/s based on the figures.

The ROV is not able to follow the circle with 0.3 m/s forward speed, while when the speed is reduced to 0.15 m/s the ROV is able to follow the circle very smoothly. The cross-track errors in Figure 4.35 shows this clearly, where it can be seen that when the speed is 0.3 m/s the ROV is getting further and further away and is not able to counteract the cross-track error. When the speed is 0.15 the cross-track is kept constantly low, which indicates that the velocity is within the physical limits of the ROV when moving on an arc with radius of 5 metres. The velocities in surge for the two different desired velocities are shown in Figure 4.36.

The maximum forward speed for ROV Minerva when moving on an arc with radius of 5 metres with the current system and configuration is assumed, based on these results, to be close to 0.2 m/s.

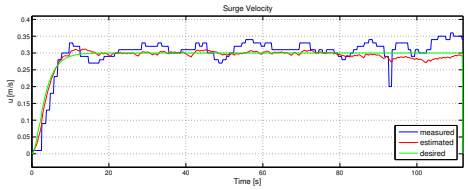


(a) Surge velocity $u = 0.3$ m/s

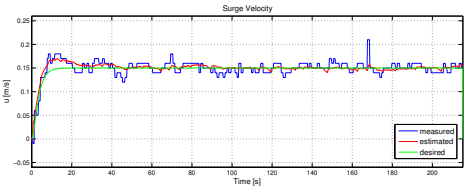


(b) Surge velocity $u = 0.15$ m/s

Figure 4.35: The cross-track error in (a) shows that the ROV is not able to come back to the path, while (b) shows that the ROV is staying within $[-0.05, 0.15]$ m/s.



(a) Surge velocity $u = 0.3$ m/s



(b) Surge velocity $u = 0.15$ m/s

Figure 4.36: The first experiment is run with 0.3 m/s surge velocity, this is decrease to 0.15 m/s for the second experiment.

4.4.3 Lawnmower Patterns

The perfect objective to use the steering for straight lines and circular arcs principle on, as described in Section 3.3.1, is to follow a lawnmower pattern. A couple of examples will be presented to show the functionality that has been implemented. For the lawnmower tests, MRU was used for heading and yaw measurements which is far superior to the gyro/compass.

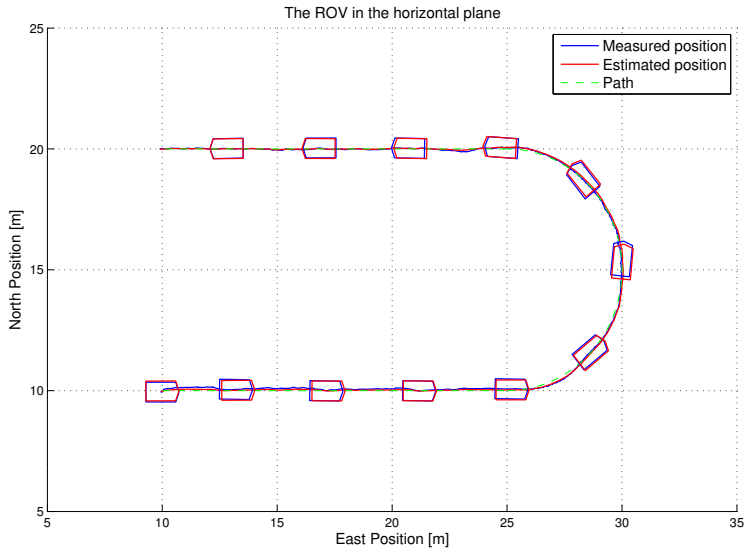


Figure 4.37: As expected when the forward speed is set to 0.2 m/s, the ROV is able to follow both the lines, which is the main objective, and the circle.

Lawnmower Pattern with 10 Metres Line Distance

A lawnmower pattern where the parallel lines are 10 metres apart, which indicates that a radius of 5 metres is the maximum allowed. The arc radius is set to 5 metres, which makes the turn between the lines a half circle. Constant forward speed is set to 0.2 m/s which reflects the results in the square tests with radius of 5 metres. Only two lines of 15 metres were traversed. The trajectory of the ROV can be seen in Figure 4.37. The ROV fulfills the objective satisfactorily by staying nicely on the path, even on the circular arc.

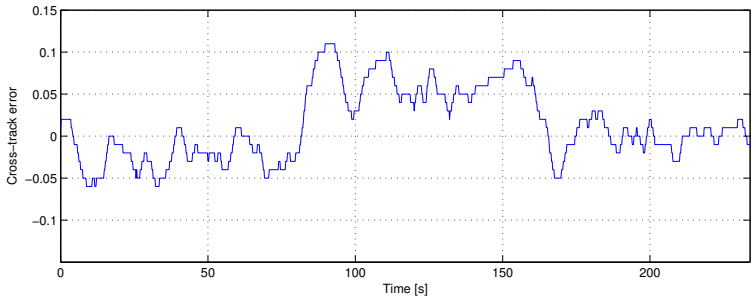


Figure 4.38: The cross-track error increases slightly while the ROV follows the circular path, but stays within $[-0.06, 0.11]$ metres.

The cross-track error clearly indicates that the performance is satisfactory, the cross-track error stays within ± 0.05 m while it is traversing the lines, while it increases insignificantly on the arc. This is, with good clearance, enough to do seabed mapping and photo mosaic for instance. The velocities can be seen in Figure 4.39. The desired surge velocity starts at zero and

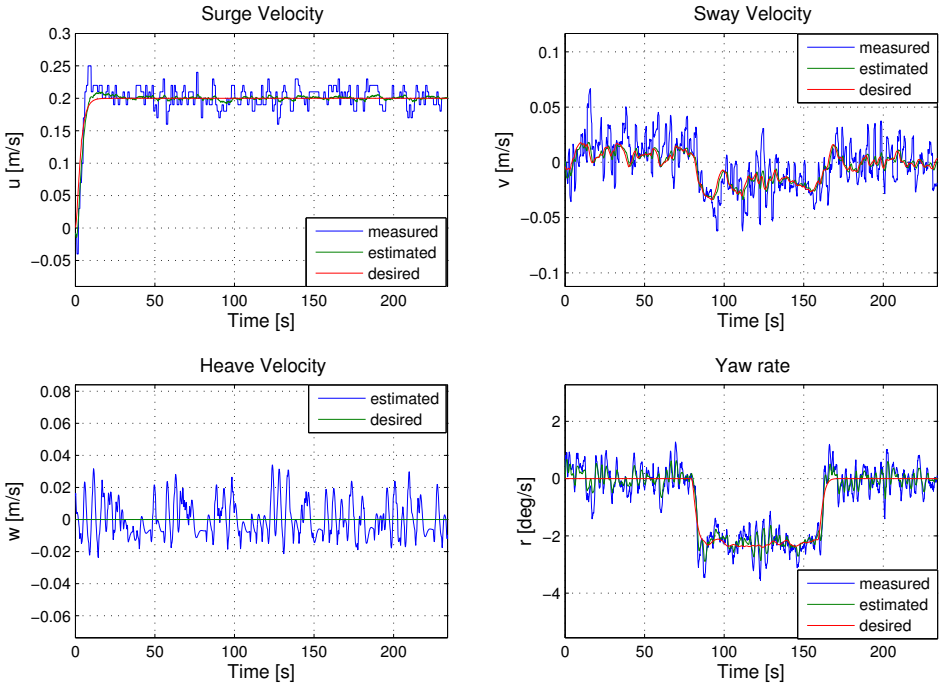


Figure 4.39: Surge velocity is kept constant, while sway velocity and yaw rate is dynamic due to cross-track error and the arc respectively.

increases smoothly, due to the velocity reference model, to the constant desired velocity. The

ROVs estimated velocity follows the desired signal smoothly, since no steps are required. The sway velocity is in charge of controlling the cross-track error, and it can be seen that just slow motion is required to stay on the path. The estimated sway velocity follows the desired sway velocity nicely. Yaw-rate is kept to zero while on the straight lines, and when the ROV enters an arc, the yaw-rate is received from the heading reference model which is followed nicely by the ROV.

The heading during the test is presented in Figure 4.40. The heading follows the desired heading closely, with a maximum deviation of 5 degrees at the beginning of the test. This corroborates with the assumption which were made initially in this section that $\psi \approx \psi_d$. This will obviously be affected by disturbances, and mainly the sea currents, for which no information currently is available. Complete result for this experiment can be found in Appendix A.2.

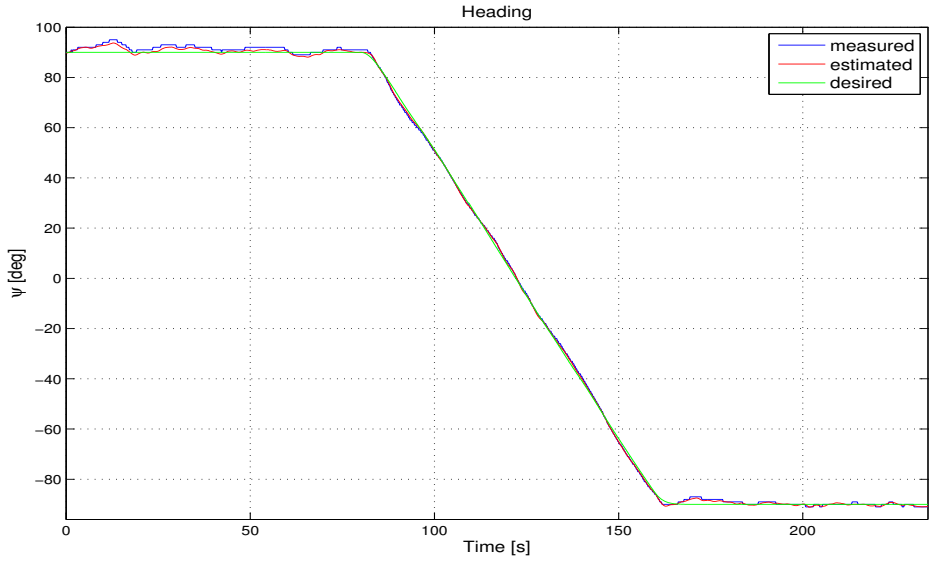


Figure 4.40: The heading follows the desired signal nicely through the turn, and stabilizes at the direction of the new line.

Lawnmower Pattern with 4 Metres Line Distance

The distance between the lines in the lawnmower pattern is dependent on the objective of the mission. In many scenarios, the distance between the lines could be quite short depending on the range of for instance sonar or video cameras. A test where the distance between the lines is 4 metres will be presented in the following. The arc radius is chosen as 2 metres, which as for the previous example means that the path between the lines will be a circle. The radius is now so small that the forward speed could not be set constant to higher than around 0.1 m/s in order to be able to follow the circular arc, as investigated in the square tests. The forward speed is now reduced on the circular arcs. The speed on the straight lines is 0.2 m/s, while it is reduced to 0.1 m/s on the circular arcs. The lawnmower pattern will for this test consist of 3 parallel lines of 16 metres, which involves two turns. The ROVs trajectory is presented in Figure 4.41.

The ROV follows the path closely, which fulfils the objective of the test. If the velocity had been constantly at 0.2 m/s, overshoots would have been expected, but is now avoided due to the reduced velocity. This can clearly be seen in Figure 4.42 where the cross-track error can be seen to stay within ± 0.1 metres of the path at all times. This result gives high confidence that the guidance system with the path following feature could be applied to a real survey mission. The velocities can be seen in Figure 4.43. It might be noticed that the forward speed initially was set

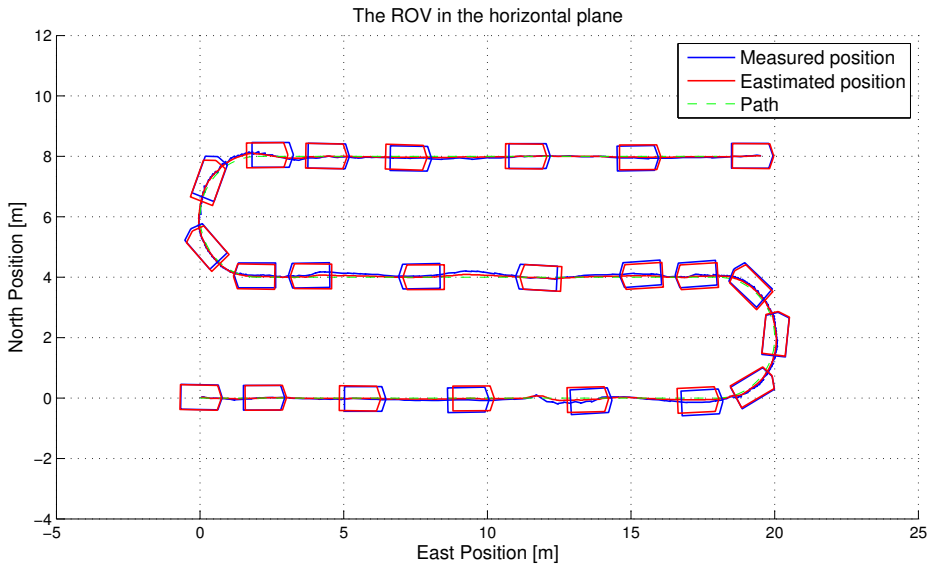


Figure 4.41: The ROV is able to stay on the path even during the turns, due to the reduced velocity on the arcs.

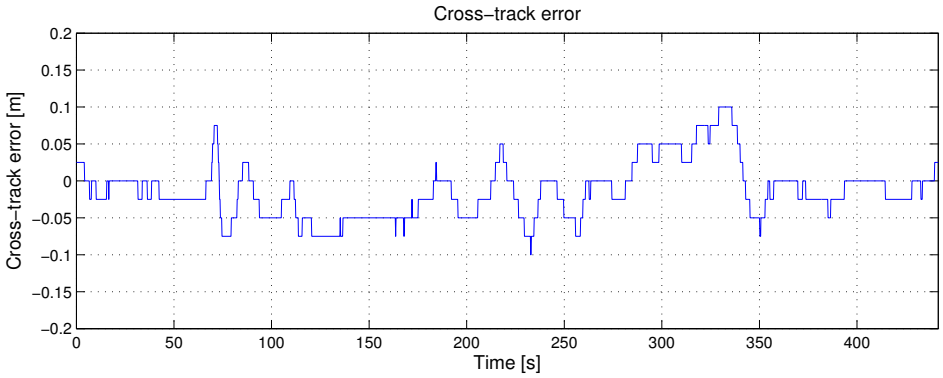


Figure 4.42: The cross-track error stays within ± 0.1 metres through the entire path.

to 0.15 m/s, but quickly changed to 0.2 m/s by an indecisive operator, but this does not affect the result. It can be seen that the surge velocity is decreasing when it enters an arc and increasing when leaving an arc. A bad DVL signal can be seen in the sway velocity measurement, but is filtered out by the observer. The desired sway velocity and yaw-rate is followed quite close, which in turn allows the ROV to stay so close to the path.

The heading is presented in Figure 4.44, and it can be seen that it follows the desired heading signal very nicely through the turns, but oscillates somewhat when stabilizing to the lines. Even though there are some oscillations, the heading stays within ± 5 degrees of the desired heading, which re-confirms that an assumption of $\psi \approx \psi_d$ holds.

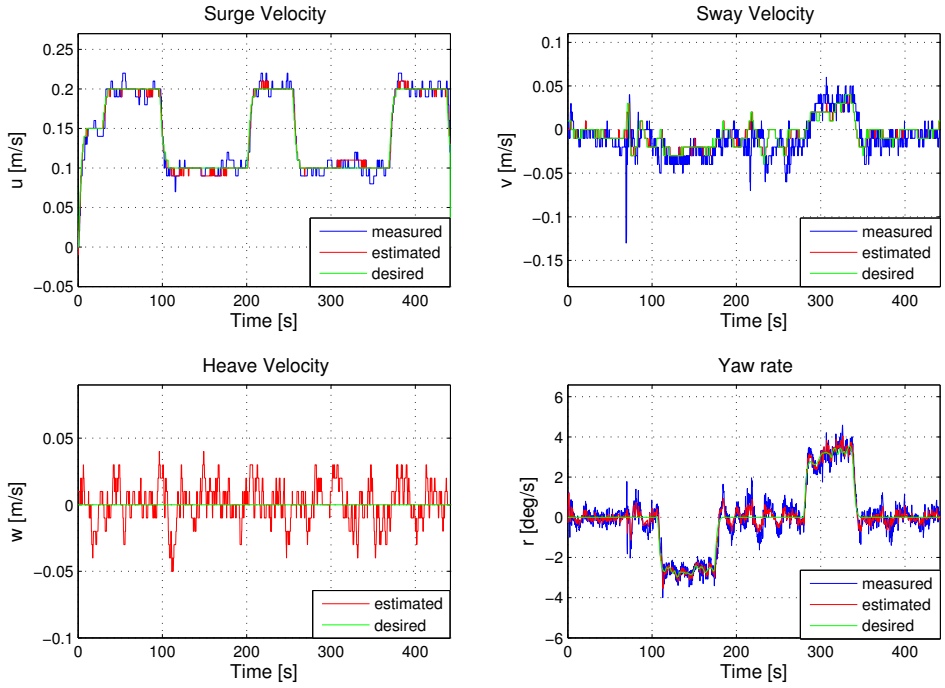


Figure 4.43: Surge velocity is varying dependent on the nature of the path, while sway velocity and yaw rate is dynamic due to cross-track error and the turn respectively.

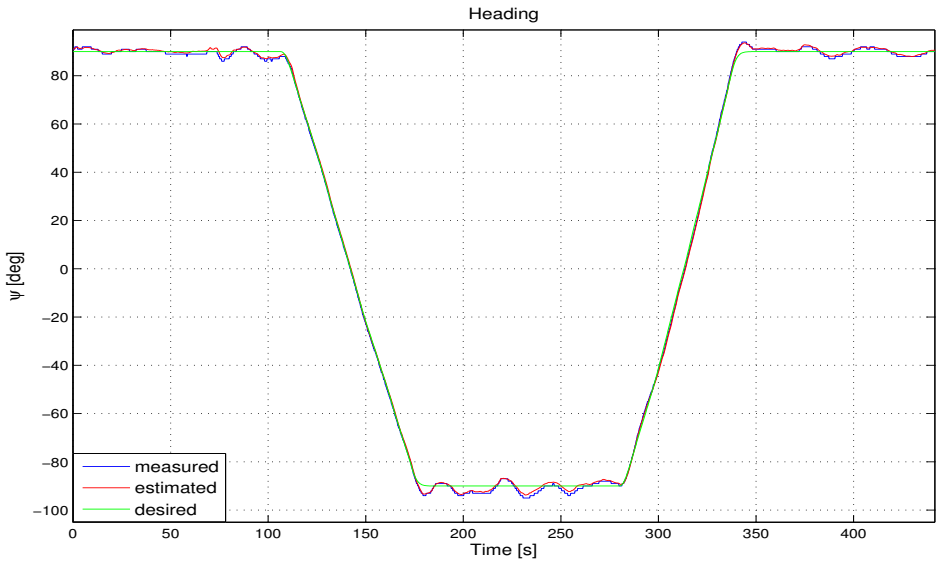


Figure 4.44: The heading is oscillating somewhat on the lines, but stays within ± 5 degrees.

4.4.4 An Alternative Approach to Lawnmower Path Following

In the previous section, it was assumed that there was no restriction on the ROV between the lines. If the pattern now is considered to consist of parallel lines, with straight lines connecting them, the approach with "cutting" the edges will not correspond with the mission objective. Two obvious approaches could be applied to fulfil this new mission criteria. The most intuitive

approach would be to stop at each way-point, change heading to align with the direction of the next line, and accelerate to full speed again. The second approach is to keep up the forward speed and take a turn outside the path starting at the way-point, such that the ROV ends up at the way-point with the heading aligned with the direction of the next line. The principle was tested, and the trajectory of the ROV can be seen in Figure 4.45.

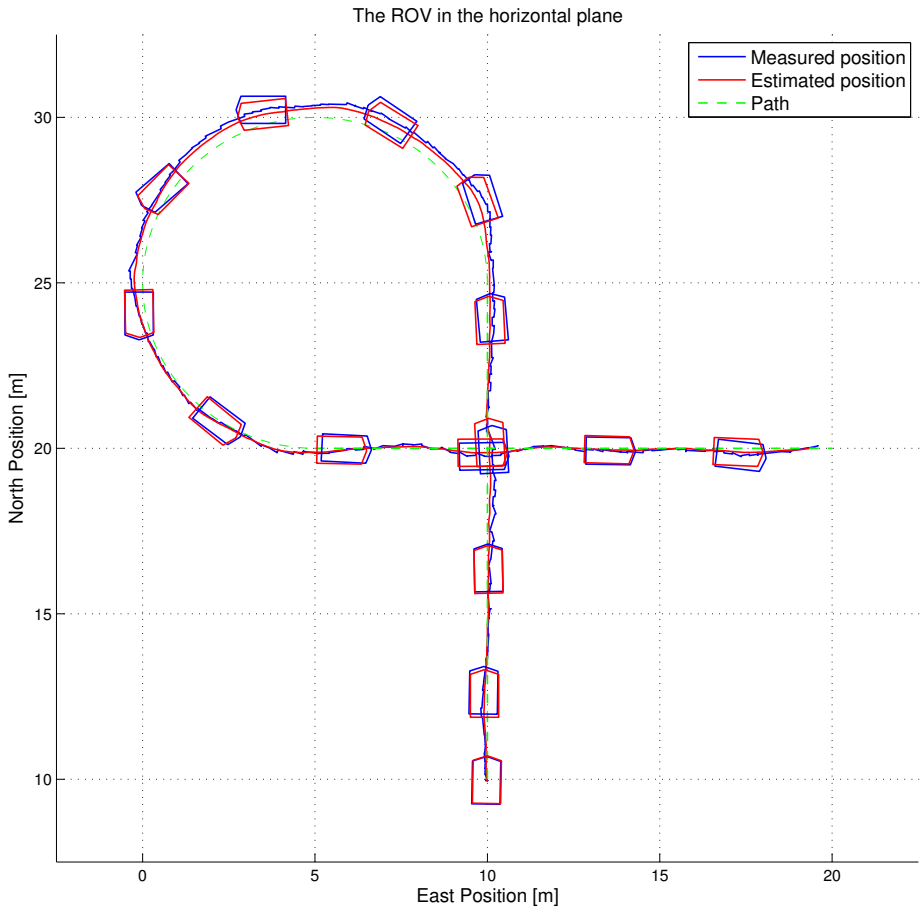


Figure 4.45: The ROV Minerva taking the turn outside the path, which makes it able to keep constant speed when switching between straight lines and does not skip any part of the path.

As can be seen the ROV initially follows the vertical line, and uses a turn outside the path to be able to approach the horizontal line with the desired heading. This gives the ROV the ability to follow both the vertical lines and the horizontal lines of a lawnmower pattern without having to expose the thruster to the constantly changing of input values. Depending on the different ROV configurations it would also be less time consuming to use this approach opposed to the first approach described above.

The cross-track error can be seen in Figure 4.46, and it is observed that the cross-track increases once the ROV enters the circle. This is not problematic, since the objective is to minimize cross-track error while on the lines, and as can be seen, the cross-track error decreases within it reaches the way-point and starts following the second line.

The other approach with coming to a halt at the way-point before changing heading and then accelerating onto the new line, was planned to be tested, but the last cruise was cut short because of a damaged fibre-optic cable. This was unfortunate, but the work on this will be continued in

Figure 4.46: The cross-track error increases when the ROV follows a circular path.

the AUR-Lab, and further work on this will be presented in Fernandes et al. (2011).

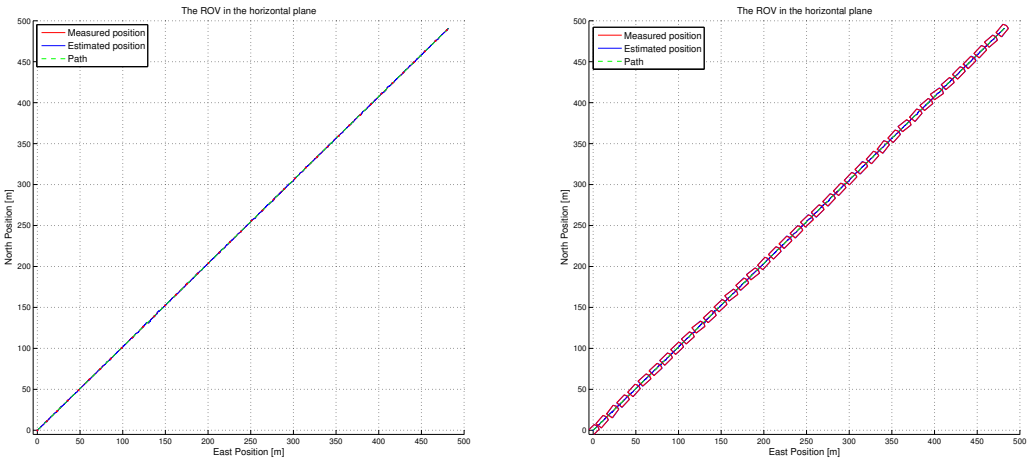
4.4.5 Mapping of Periphylla periphylla Population in Verrabotn

At the 30th of March, Dr. Martin Ludvigsen, Phat Truong and myself from AUR-Lab travelled to Verrabotn at Fosen, Trøndelag, to assist a group from the institute of biology with ROV operations. They were currently doing research with RV Gunnerus, and one of the interests was the population of the jellyfish Periphylla periphylla (Figure 4.47). Dead Periphylla periphylla was mapped at the seabed, which was done manually due to the need to be very close to the seabed with the possibility to frequently stop and inspect different areas. A second inspection was done at some altitude from the seabed, in order to get an overview of the living Periphylla periphylla population in the water column. During the second inspection the guidance system described in this thesis was used, mainly to follow, more or less, the same route as for the seabed inspection. The result from the survey can be seen in Figure 4.48, where Figure 4.48a presents

Figure 4.47: A Periphylla periphylla jellyfish.

the real measured results, while Figure 4.48b shows the same results, but the ROV outline is

scaled by a factor of 10, for the reader to be able to understand the figure. The forward speed is 0.3 m/s. As can be seen the ROV follows a line towards a point that lies 491 metres North and 482 metres East, which corresponds to a line of $\sqrt{491^2 + 482^2} = 688$ metres with the direction $\arctan(482/491) = 44.47$ degrees. The heading can be seen in the South-East corner of Figure 4.49 in which the position and orientation is presented. The North and East signals are



(a) Actual size ROV.

(b) ROV scaled by a factor of 10

Figure 4.48: The ROV Minerva is following a line, in order to make Periphylla periphylla observations in the water column.

steadily progressing on the line, while the depth is nicely controlled to 43 metres. The heading is measured by the fluxgate compass, and estimated by extended Kalman filtering. The heading is oscillating quite a lot, but generally stays within ± 10 degrees which could be expected due to the fluxgate compass.The cross-track error can be seen in Figure 4.50, which stays nicely within ± 0.5 metres through the whole line, or over 41 minutes. The peak at around 600 seconds is a drop-out of HiPaP signals followed by lagging HiPaP signals between 600 and 620 seconds. The ROV stayed within 1 metres despite this. Hopefully the results has proven that the system is reliable and able to achieve high performance with respect to mission objective, and that it can be used for future surveys and mapping missions as well.

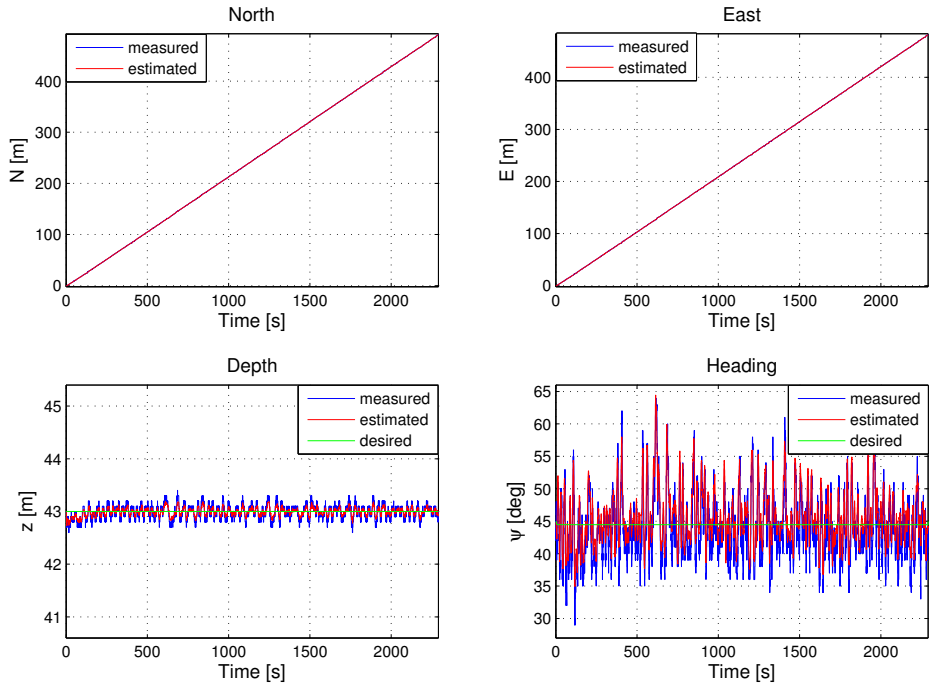


Figure 4.49: The position and orientation of the ROV while mapping Periphylla periphylla.

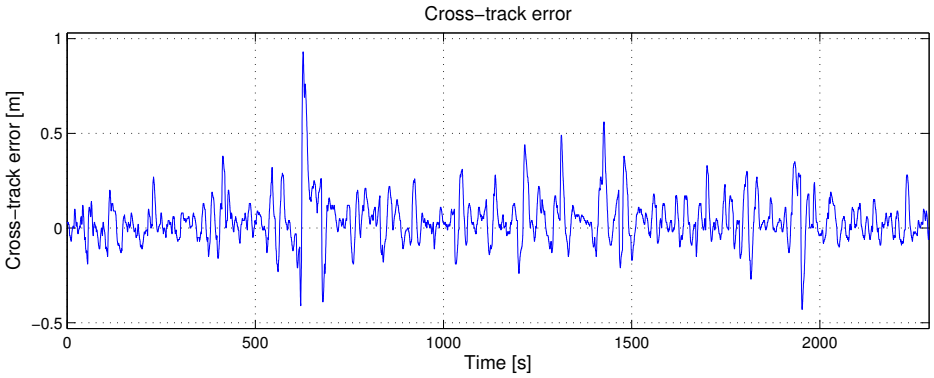


Figure 4.50: The cross-track error stays within ± 0.5 metres, where the peak at 600 seconds is due to a small period of bad HiPaP readings.

4.5 Discussion

Different guidance strategies has been simulated and evaluated. With respect to travelling along arcs, it has been shown that the energy consumption increases when a to high forward speed is applied. It has been demonstrated that the strategies that are the least time consuming is not always the least energy consuming. In most cases the fastest strategy has the highest energy consumption. It has been demonstrated that in order to plan lawnmower pattern missions smartly, the radius of the arc should always be considered when choosing the forward speed.

A fully integrated guidance system has been implemented in the overall control system developed through the AUR-Lab. The functionality presented in this section has mainly been within path following. The path following scheme has been tested thoroughly during full scale experiments. The path following scheme has shown to work well for different scenarios. The most problematic aspect has been to decide the radius for a given forward speed. Turning ratios have been investigated to find out at which forward speed narrow turns can be taken. This has been used to present results of successful lawnmower pattern path following scenarios. It has been shown that the guidance system can operate over long time without major bugs spoiling the performance of the system.

Chapter 5

Collision Avoidance

When planning a mission it is important to take into account all limitations off the vehicle in addition to restrictions with respect to geography and environmental disturbances. When a high number of relevant factors is taken into account during mission planning, the safety and probability of mission completion is increased. Although a mission is thoroughly planned beforehand, some factors will always be unforeseen.

5.1 Obstacles

The operations of obstacle handling can be divided into three different phases or objectives, *obstacle detection, obstacle identification and obstacle avoidance*. Detection and identification is dependent on sensors such as sonar or laser, that can give the system feedback if there exists an obstacle in the work space, and where the object is located. Once an obstacle is detected, a replanning algorithm needs to generate a new path or steering direction which is feasible with respect to the new constraint that has been introduced. Obstacle identification, or mapping of an obstacle, is not always a desired objective, but will in general be a result of avoiding an obstacle. In some cases avoiding an obstacle might be impossible or not immediately possible, then the objective of the mission could be changed to focus on identifying the obstacle for future missions.

5.2 Underwater Collision Avoidance

One of the main differences between collision avoidance for surface vessels and underwater vehicles is that for surface vessels much more information is available a priori and paths can be generated off-line, while for underwater vehicles collision avoidance must be ensured on-line. Because of the 3-dimensional operating space which makes the environment larger than for surface vessels, and the uncertainties with respect to this environment, it is preferable to use a local approach to collision avoidance.

5.2.1 Obstacle Mapping

Underwater mapping is usually done with sonar sensors. Main types are single beam sonar and multi beam sonar. The range of a sonar varies from a couple of metres to several thousand metres. For an UUV the range of a forward looking sonar is typically around $0.5 - 100$ metres. An UUV can be equipped with sonar in all direction, were a downwards scanning sonar is in most cases a minimum. For collision avoidance a minimum set up requires at least a forward looking sonar. An addition of side scan sonars will increase the robustness and accuracy of the system.

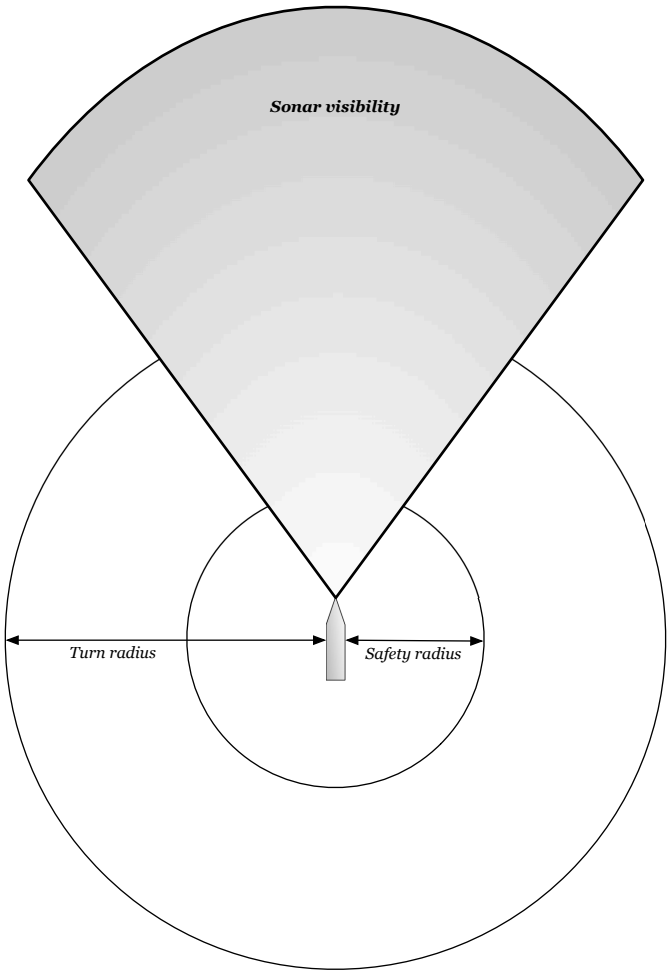


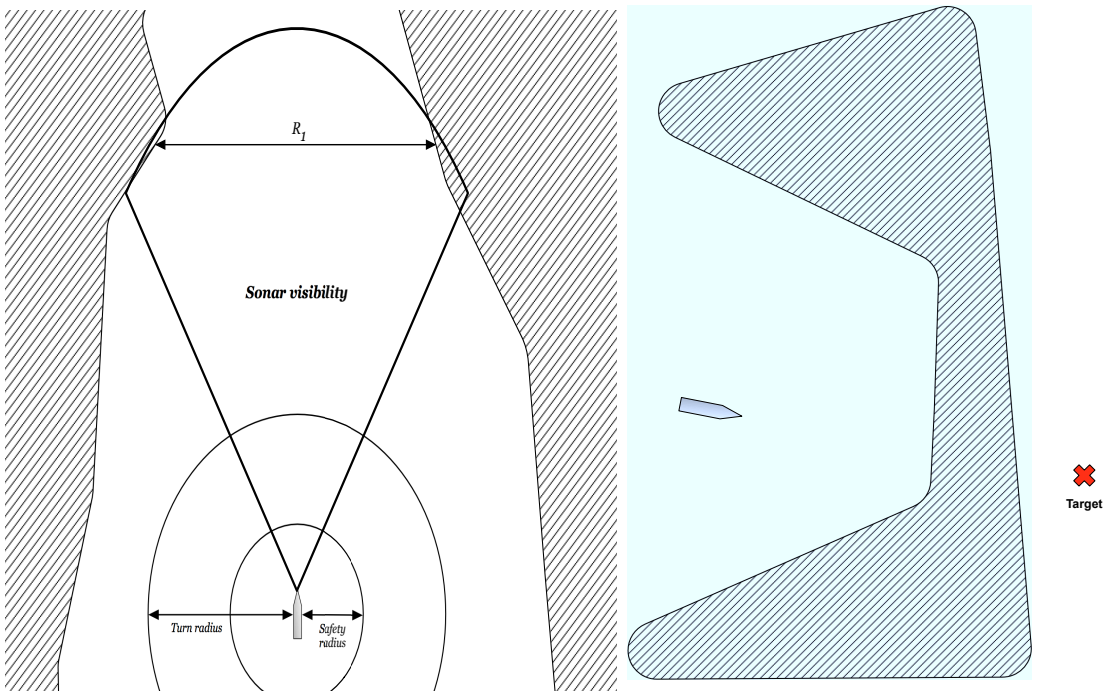
Figure 5.1: The vehicle with its critical parameters concerning collision avoidance.

In Figure 5.1 an outline of the sonar field and the critical parameters to be considered is presented. The turn radius is the space the vehicle needs to take a 180° turn, or go upwards in a spiral. The turn radius will differ drastically from a ROV to an AUV, depending on actuation, size and shape. The safety radius is a predefined space around the vehicle that should not intersect with an obstacle.

The information obtained from the sonar is represented in a map. If a scenario in the horizontal plane is considered, such a map will be a 2-dimensional grid, consisting of cells with some value representing the probability of a obstacle occurring in the position corresponding to the cell. The software translating the sonar signals into such a map will not be discussed in this thesis, but it will be assumed that building this map is feasible, which is a fair assumption. The map will continuously be updated as the vehicle is moving through the environment. Only a part of the map is considered by the local algorithm, which is called the active window. Information in the map outside this window could be deleted as the vehicle passes that environment, if that information is not wanted for off-line purposes at a later stage. The map is sent as an input to the collision avoidance system.

5.2.2 Supervisory System

Local collision avoidance algorithms do all have the drawback that by considering only the immediate area they can cause the vehicle to get trapped. To deal with this problem several



(a) A canyon situation. (b) A trap situation.

Figure 5.2: Critical situations for local collision avoidance algorithms.

methods can be applied, where the general idea is to have some supervisor that is able to detect trap situations and other situation where the local algorithm fails.

In Antonelli et al. (2001) such a supervisory system for an AUV is proposed. The supervisory system is designed to detect critical situations, which has been divided into states that should be activated if the situation corresponding to the state is detected. The states are defined as *Trap*, *Canyon* and *Nominal*. The trap state is activated if a trap situation is detected, and a different procedure is chosen by the supervisor to guide the vehicle out of the trap. The method used for this state in Antonelli et al. (2001) is a wall following method (WFM). The canyon state is activated if the vehicle is approaching confined space, where the situation is critical if the confined area is smaller than twice the vehicles turn radius. This situation is critical because a safe path after entering such a canyon can not be guaranteed. If such a situation is detected, the simple way to avoid the situation becoming critical is to defer from entering the canyon and turning towards the last observed safe direction. The third state is the nominal state, which implies that nothing is wrong, and the algorithm is run as normal.

Detection of the two critical situation can be done different ways, but for the canyon situation it is quite straight-forward to detect. The trap situation can be done in different ways, which also depends on which algorithm is used. A condition that with absolute certainty implies a trap is if the speed is zero, this will however almost never be the case. In most cases a trap situation will give a cycle behaviour or an oscillatory behaviour around some local minima. A simple way, but also conservative test is to compare the direction of travel χ with the direction of the target χ_t. In Borenstein & Koren (1989) the criteria $|\chi_t - \chi| > 90^\circ$ is proposed as a conservative condition for detecting trap situations. Another method is to monitor the diversions from the target, if the diversion within some limit changes from left to right or vice versa, this implies a trap situation.

A proper supervisory system is crucial for a collision avoidance system, especially if local collision avoidance algorithms are applied. Global collision avoidance algorithms are usually not sufficient either when moving in a dynamic environment, which will be addressed later in

this chapter.

5.3 Local Collision Avoidance Algorithms

Local collision avoidance algorithms are in general reactive algorithms, which demands little computational time, but has the major drawback that they cannot guarantee an optimal path. These algorithms consider only the environment close to the vehicle, which the name insinuates.

5.3.1 The Dynamic Window Approach

The Dynamic Window Approach was first proposed by Dieter Fox, Wolfram Burgard and Sebastian Thrun in Fox et al. (1997). The concept behind this approach was to design the algorithm based on the velocity and acceleration constraints of the robot they were working with. The avoidance algorithm is derived directly from the vehicles motion dynamics and hence not only generates inputs that avoids collision, but the inputs should also be physically feasible for the vehicle.

It is assumed that the translational and angular velocities (u, r) are constant within a given time-interval. The method also only considers movement along arcs, where the center of an arc at time-interval i is given as

$$x_i^c = -\frac{u_i}{r_i}\sin\theta(t_i) \tag{5.1}$$

$$y_i^c = \frac{u_i}{r_i}\cos\theta(t_i) \tag{5.2}$$

where u_i and r_i are consistent with Table 2.1. The radius of the arc r^c is given as

$$r_i^c = \frac{u_i}{r_i} \tag{5.3}$$

Search Space

The search space of pairs (u_i, r_i) is denoted as V_r and three different restrictions are set on the search space, that the velocities are admissible (V_a), attainable (V_s) and that they lie within a *dynamic window* (V_d). The search space can then explicitly be formulated as

$$V_r = V_s \cap V_a \cap V_d \tag{5.4}$$

where the admissible velocities and the dynamic window are given by

$$V_a = \left\{ (u,r) \,\middle|\, u \leq \sqrt{2 \cdot dist(u,r) \cdot \dot{u}_b} \wedge r \leq \sqrt{2 \cdot dist(u,r) \cdot \dot{r}_b} \right\}, \tag{5.5}$$

$$V_d = \left\{ (u,r) \,\middle|\, u \in [u_a - \dot{v}\Delta t, u_a + \dot{v}\Delta t] \wedge r \in [r_a - \dot{r}\Delta t, r_a + \dot{r}\Delta t] \right\}. \tag{5.6}$$

V_s is simply all the possible velocity pairs (u, r) the vehicle can obtain.

Optimizing velocities

For an optimal selection of velocity pairs, the pair (u, r) in the search space V_r that maximizes the objective function

$$G(u,r) = \sigma(\alpha \cdot heading(u,r) + \beta \cdot dist(u,r) + \gamma \cdot velocity(u,r)), \tag{5.7}$$

should be selected. Maximizing the objective function ensures a) that the vehicle keeps direction towards the target through the $heading$ function, b) high velocity through the $velocity$ function and c) collision avoidance through the $dist$ function. The different functions are weighted by the parameters α, β and γ. The function σ is a smoothing function, which can be implemented as a low-pass filter.

5.3.2 The Artificial Potential Field Approach

Using potential fields in collision avoidance schemes was first proposed by Oussama Khatib in Khatib (1985) and Khatib (1986). The intent was to design low level control components for robots that were able to react fast to a dynamic and complex environment. The method is known for fast implementation and its simple, but elegant nature. The philosophy of the approach is best described as in Khatib (1986):

> *The manipulator moves in a field of forces. The position to be reached is an attractive pole for the end effector and obstacles are repulsive surfaces for the manipulator parts.*

Since Khatib (1986) considered this approach for a robot arm, some explanation is needed. If we consider a ship at sea starting at a point A moving towards a point B, then B is the position to be reached and hence acts as an attractive pole on the ship. Obstacles in the area between A and B, such as islands, offshore constructions and other vessels acts as repulsive forces on the ship. This will make the obstacles "push" the ship away from them, while point B drags the ship towards it.

If $\boldsymbol{p}_d$ denotes the target position and a single obstacle is denoted as O, the objective of moving to the target point while avoiding the obstacle O can be achieved by exposing the vehicle to an artificial potential field

$$\boldsymbol{U}_{art}(\boldsymbol{p}) = \boldsymbol{U}_{\boldsymbol{p}_d}(\boldsymbol{p}) + \boldsymbol{U}_O(\boldsymbol{p}). \tag{5.8}$$

The force generated by this potential field can be written as (Khatib, 1986)

$$\boldsymbol{F}^*_{art} = \boldsymbol{F}^*_{\boldsymbol{p}_d} + \boldsymbol{F}^*_O. \tag{5.9}$$

Attracting Force

In (5.9) $\boldsymbol{F}^*_{\boldsymbol{p}_d}$ represents the attracting force, which makes $\boldsymbol{p}$ reach the target $\boldsymbol{p}_d$. This corresponds to the proportional term in a PD servo. The attractive potential field is given as

$$\boldsymbol{U}_{\boldsymbol{p}_d}(\boldsymbol{p}) = \frac{1}{2} k_p (\boldsymbol{p} - \boldsymbol{p}_d)^2, \tag{5.10}$$

where k_p is the position gain. The forces ensuring vehicle motion towards the target point and stability are

$$\boldsymbol{F}^*_{\boldsymbol{p}_d} = -k_p(\boldsymbol{p} - \boldsymbol{p}_d) - k_v \dot{\boldsymbol{p}}, \tag{5.11}$$

where k_v is the velocity gain. In order to ensure that the velocity vector points toward the target, and that the velocity is limited to a specified value U_{max}, (5.11) is rewritten where the desired velocity vector is specified as

$$\dot{\boldsymbol{p}}_d = \frac{k_p}{k_v}(\boldsymbol{p}_d - \boldsymbol{p}). \tag{5.12}$$

The attraction force vector can now be rewritten, with the inclusion of servo control and velocity limitation as

$$\boldsymbol{F}^*_{\boldsymbol{p}_d} = -k_v(\dot{\boldsymbol{p}} - \kappa \dot{\boldsymbol{p}}_d), \tag{5.13}$$

where

$$\kappa = \min\left(1, \frac{U_{max}}{\sqrt{\dot{\boldsymbol{p}}_d^T \dot{\boldsymbol{p}}_d}}\right). \tag{5.14}$$

This will make the vehicle travel at the specified maximum speed at a straight line towards the target, except for when the vehicle enters repulsive potential regions.

Repulsive Forces

The potential field generated by an obstacle should increase towards infinity as the vehicle approaches the surface of the obstacle, while it should be negligible beyond a given region. This to avoid undesirable effects to the vehicle when it is outside the obstacles vicinity. The artificial potential field based on the shortest distance to an obstacle is proposed in Khatib (1986).

$$\boldsymbol{U}_O(\boldsymbol{p}) = \begin{cases} \frac{1}{2}\lambda\left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)^2 & \text{if} \rho \leq \rho_0 \\ 0 & \text{if} \rho > \rho_0 \end{cases} \tag{5.15}$$

The limitation distance of the potential field is represented by ρ_0 while the shortest distance to the obstacle is represented by ρ, λ is a constant gain. When selecting ρ_0 the selected speed U_{max} should be taken into account in addition to the vehicles deceleration ability. The force repulsing the vehicle with respect to an obstacle O_i is given by

$$\boldsymbol{F}_{O_i}^* = \begin{cases} \frac{1}{2}\lambda\left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2}\frac{\partial \rho}{\partial \boldsymbol{x}} & \text{if} \rho \leq \rho_0 \\ 0 & \text{if} \rho > \rho_0 \end{cases} \tag{5.16}$$

where $\frac{\partial \rho}{\partial \boldsymbol{x}}$ denotes the partial derivative vector of the distance from the vehicle to the obstacle,

$$\frac{\partial \rho}{\partial \boldsymbol{x}} = \left[\frac{\partial \rho}{\partial x}\frac{\partial \rho}{\partial y}\frac{\partial \rho}{\partial z}\right]^T. \tag{5.17}$$

For several obstacles the total repulsive force becomes

$$\boldsymbol{F}_O^* = \sum_i \boldsymbol{F}_{O_i}^*. \tag{5.18}$$

By inserting the calculated terms into (5.9), the desired forces are calculated. For a fully actuated vehicle the forces could be fed to the control allocation, which would distribute the forces properly to fulfil the control objective. For underactuated vehicles the direction of the resultant force should act as a heading command, and the force could be inserted into the control allocation as forces in surge.

Limitations of the Algorithm

Potential field methods for collision avoidance have several drawbacks, which are discussed more extensively in Koren & Borenstein (1991). There are four main drawbacks

- Trap situations due to local minima, which can end up in the vehicle getting trapped. The repulsive forces cancels out the attracting force ending up in either a stand still, or a limit cycle.
- No passage between closely spaced obstacles. The repulsive force could become greater than the attracting force either ending up in a limit cycle or with the algorithm choosing another route.

- Oscillations in the presence of obstacles. The vehicle has to move quite close to an obstacle to experience a repulsive force, which could create oscillations in the presence of several obstacles.

- Oscillations in narrow passages. The vehicle could start to oscillate from wall to wall, when it does not experiences repulsive force from both walls simultaneously.

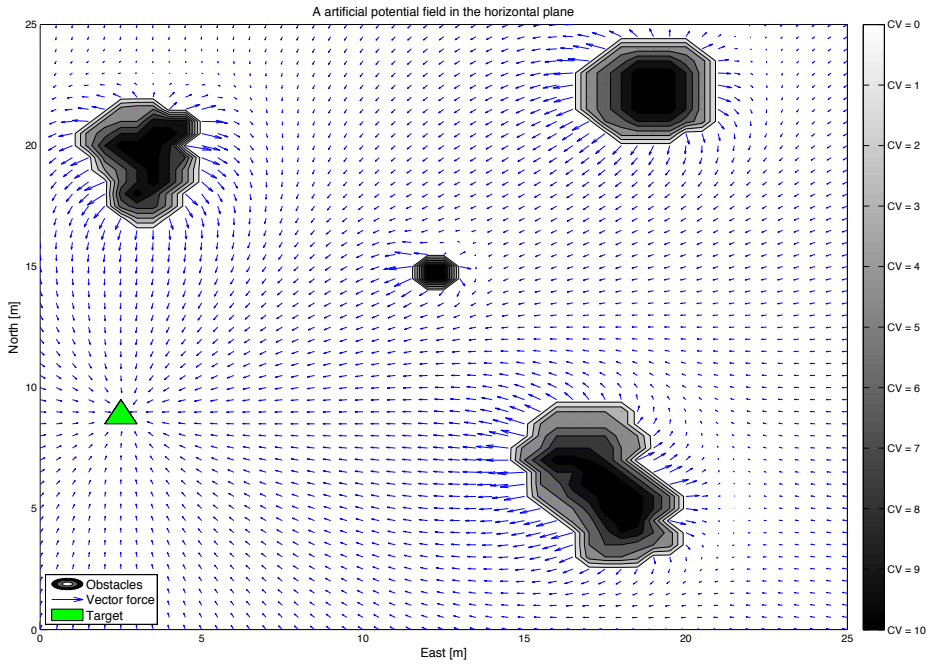


Figure 5.3: An artificial potential field for a region with several obstacles represented by the gray areas.

The Virtual Force Field (VFF) Method

The VFF method is an extension or version of the potential field method for collision avoidance. The method uses a histogram grid $\boldsymbol{C}$ which is a two-dimensional Cartesian grid. The method is presented in Borenstein & Koren (1989). Each cell (i, j) in the grid are given a certainty value $C(i, j)$ that indicates the level of certainty that the cell is occupied by an obstacle. Each cell in the grid has a corresponding virtual force that is applied to the vehicle, which is inversely proportional to the distance between the cell and the vehicle and also proportional to the certainty value $C(i, j)$. Unoccupied cells will not generate any force, while a high degree of certainty corresponds to a force of greater magnitude. The certainty values will account for sensor uncertainties and other shortcomings when generating the search space. The magnitude of the force from a cell is given as

$$\boldsymbol{F}(i,j) = \frac{F_{cr} C(i,j)}{d^2(i,j)} \left[\frac{x_i - x_0}{d(i,j)} \hat{x} + \frac{y_i - y_0}{d(i,j)} \hat{y} \right], \tag{5.19}$$

where F_{cr} is the repelling force constant, $d(i, j)$ is the distance between the vehicle and the cell (i, j), (x_0, y_0) are the current coordinates of the vehicle and (x_i, y_i) are the coordinates of the

cell (i, j). The resultant repulsive force is the sum of the repulsive forces of all individual cells

$$\boldsymbol{F}_r = \sum_{i,j} \boldsymbol{F}(i, j). \tag{5.20}$$

The attracting force from the target point is given by

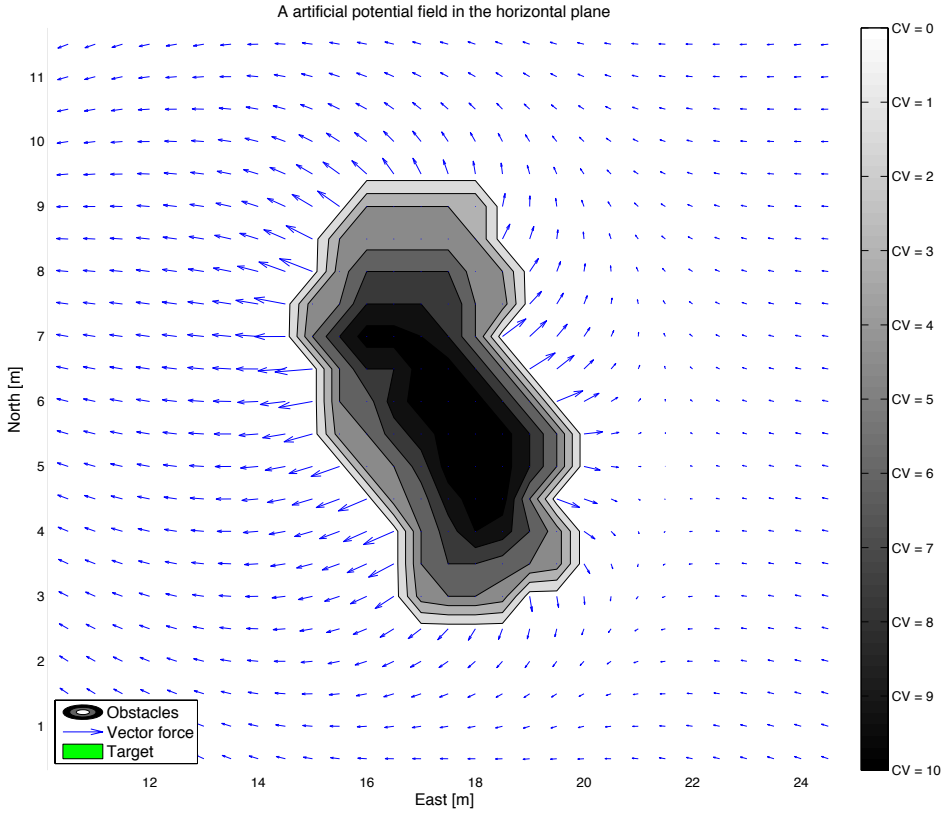


Figure 5.4: The behaviour of the potential field near an obstacle.

$$\boldsymbol{F}_t = F_{ct}\left[\frac{x_t - x_0}{d(t)}\hat{x} + \frac{y_t - y_0}{d(t)}\hat{y}\right] \tag{5.21}$$

where F_{ct} is the attracting force constant, $d(t)$ is the distance between the target and the vehicle and (x_t, y_t) are the coordinates of the target point. The total force on the vehicle is given by

$$\boldsymbol{R} = \boldsymbol{F}_r + \boldsymbol{F}_t. \tag{5.22}$$

In Figure 5.3 a potential field is demonstrated using the VFF method. The dark areas represents obstacles, and the color spectrum represents the certainty values given in the color bar to the right. The vector represented in each cell is calculated based on the vehicle being at that location. The length of the vector is corresponding to the magnitude of the force, while the direction of the vector corresponds to the direction to which the force should be applied. The interesting part of the map is around the obstacles, where it can be seen that the vector is pointing away from the obstacles despite the direction is away from the target which is the green triangle in Figure 5.3. The behaviour close to an obstacle is shown in Figure 5.4 which is a zoom-in on the map shown in Figure 5.3. The behaviour becomes more apparent here, with the direction always away from

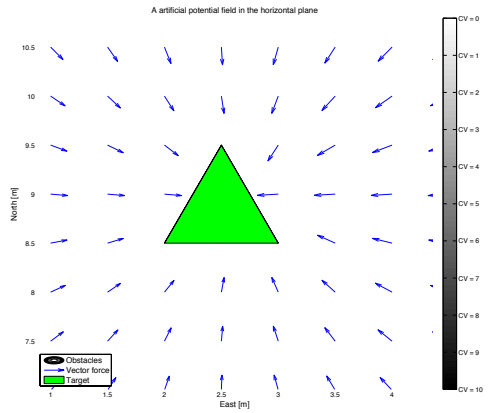


Figure 5.5: The behaviour of the potential field near the target.

the obstacle. The problem with local minima can also be seen here in the coordinates around $(5, 21)$ were it can be seen that the force goes towards zero. In Figure 5.5 the behaviour around the target can be seen, only to verify that the forces converge to the target.

5.3.3 The Vector Field Histogram Approach

The apparent shortcomings of the VFF method led to the work on the Vector Field Histogram (VFH), where the focus was to overcome some of the limitations experienced with the VFF method. The method was proposed in Borenstein & Koren (1991).

The method also starts with expressing the environment around the vehicle in a Cartesian histogram grid $\boldsymbol{C}$. The content off each cell in the active region of the histogram grid $\boldsymbol{C}^*$ is mapped into the corresponding sector of a polar histogram $\boldsymbol{H}$.

The Polar Histogram

The polar histogram is comprised of n angular sectors of width α, where each sector has a corresponding value h_k that represents the polar obstacle density in the sectors direction. The way the different cells are mapped into the appropriate sector is by calculating the obstacle vector of each cell for which the direction is given by

$$\beta_{i,j} = \tan^{-1}\left(\frac{y_j - y_0}{x_i - x_0}\right), \tag{5.23}$$

where (x_0, y_0) are the coordinates of the vehicle and (x_i, y_j) are the coordinates of the current cell. The magnitude of the vector is given by

$$m_{i,j} = \left(C^*_{i,j}\right)^2 (a - bd_{i,j}), \tag{5.24}$$

where a and b are positive constants, $C^*_{i,j}$ is the certainty value of cell (i, j) and $d_{i,j}$ is the distance between the vehicle and the current cell. The parameters a and b are chosen such that

$$a - bd_{max} = 0, \tag{5.25}$$

where d_{max} is the distance to the cell furthest away from the vehicle. Cell (i, j) belongs to sector

$$k = INT(\beta_{i,j}/\alpha). \tag{5.26}$$

The polar obstacle density h_k for sector k can then be calculated as

$$h_k = \sum_{i,j} m_{i,j}. \tag{5.27}$$

Because the sectors are based on a grid, the mapping into sectors may cause errors depending on the grid versus cell size. For this reason a smoothing function is applied to the polar obstacle density. The function is defined as (Borenstein & Koren, 1991)

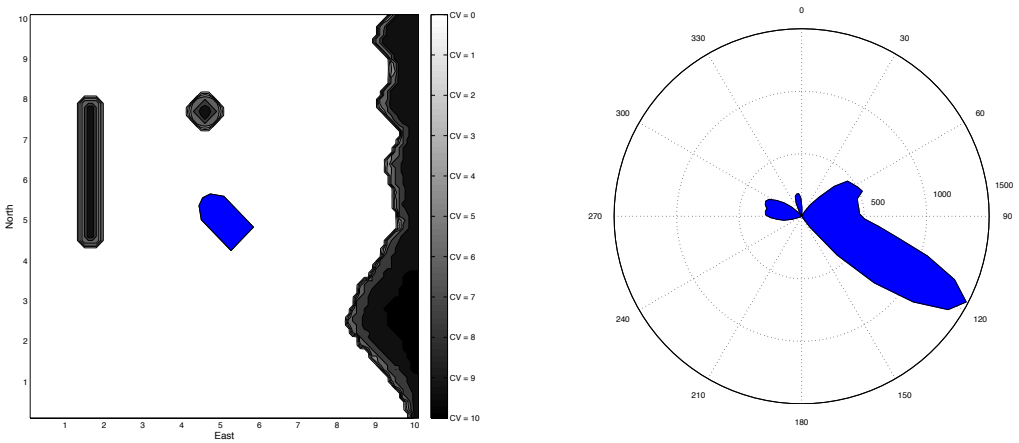
$$h_k^* = \frac{h_{k-l} + 2h_{k-l+1} + \cdots + lh_k + \cdots + 2h_{k+l-1} + h_{k+l}}{2l+1} \tag{5.28}$$

Steering and Velocity Control

The smoothed polar histogram can now be used to determine the steering angle and as will also be shown it can be used to determine speed reduction in case of obstacles ahead in the steering direction. From the smoothed histogram in Figure 5.7b it can be seen that there are peaks and valleys. The threshold which is the red line in Figure 5.7 defines which valleys are *candidate valleys*. If a valley consist of PODs below the threshold, the valley is considered as a candidate for travel. The valley which direction is nearest the direction to the target k_{targ} is chosen for travelling through. The border sector within the selected valley that is closest to the direction of the target is denoted as k_n while the far border is denoted as k_f. Border sectors denote the first and last sector of the valley. The desired course angle is chosen as

$$\chi_d = \frac{k_n + k_f}{2}. \tag{5.29}$$

For valleys that are very wide, only a part of the valley is considered defined by the parameter s_{max}. In the case of a valley wider than s_{max} the far border is chosen as $k_f = k_n + s_{max}$ and the desired course angle is again chosen as in (5.29). The algorithm is not very sensitive to the threshold, but it should be noted that by choosing a too high value for the threshold could result in the algorithm not identifying an object, while a too low threshold will result in the algorithm not considering narrow passages.



(a) Example of an obstacle scenario.

(b) Smoothed polar obstacle density histogram in polar form.

Figure 5.6: An example of the Vector Field Histogram approach.

The speed assignment should be set to a constant value U_{max} at the beginning of a run. When moving in sectors where there are obstacles ahead, a speed reduction should occur. The sector which the current direction of travel lies within is denoted $h_c^{'}$ and $h_c^{'} > 0$ indicate that an obstacle lies ahead of the vehicle. The speed reduction can then be implemented as

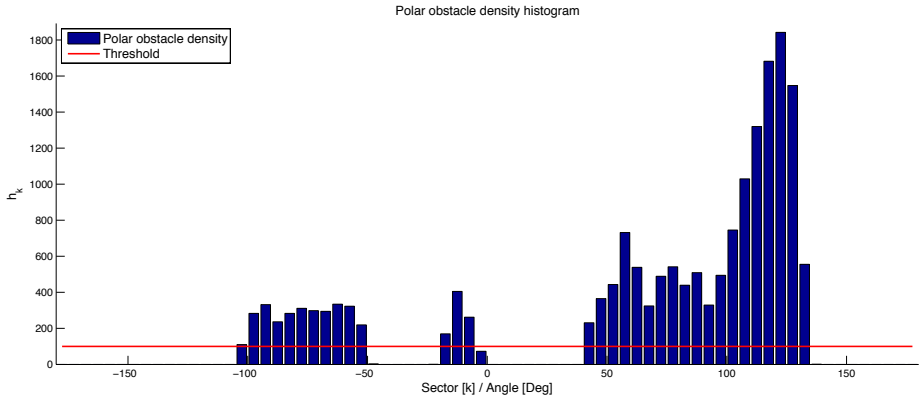
$$U^{'} = U_{max}\left(1 - \frac{h_c^{''}}{h_m}\right) \tag{5.30}$$

where

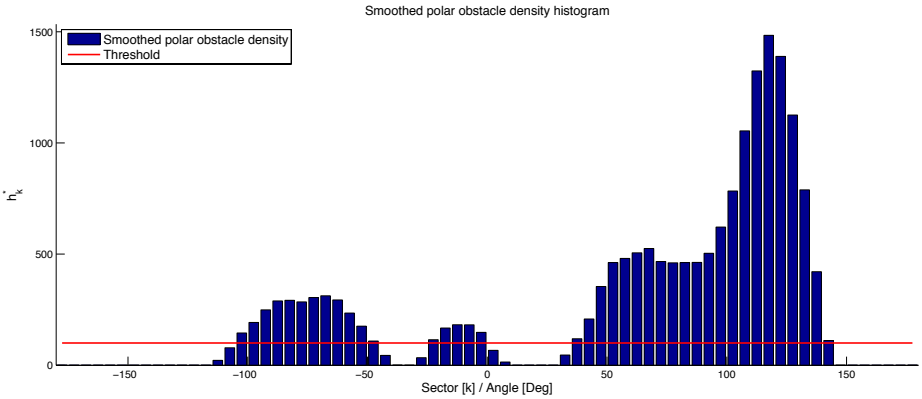
$$h_c^{''} = \min(h_c^{'}, h_m) \tag{5.31}$$

and h_m is a constant that gives a sufficient speed reduction. Non-zero forward speed is guaranteed from (5.31). To demonstrate the VFH algorithm, consider the scenario given in Figure 5.6a. The obstacle is represented by the certainty values for the cells, indicated by the color bar.

The smoothed obstacle density histogram for the scenario in Figure 5.6a is presented in Figure 5.6b. There are three 'leaves' in the histogram which corresponds to the three obstacles, where the wall to the right in the Figure 5.6a is the dominating part to the right in the histogram. The histogram is theoretically easier to analyse in a normal histogram. In Figure 5.7 both the polar obstacle density histogram and the smoothed histogram is presented in (a) and (b) respectively.



(a) Polar obstacle density histogram.



(b) Smoothed polar obstacle density histogram.

Figure 5.7: By smoothing the polar histogram some additional sectors are excluded, which adds some robustness with respect to not moving too close to obstacles.

The difference between the two histogram is defined by (5.28), where in this instance the

smoothing parameter $l = 2$. As can be seen by investigating the two figures, some additional sectors are excluded from the valleys by smoothing the histogram.

For each of the three sectors the desired course angle is calculated as in (5.29), and the course which is closest to the direction of the target is chosen as the desired course angle. The velocity is calculated from (5.30). The direction of the target is easily calculated as

$$k_{targ} = \text{atan2}\left(\frac{y_{targ} - y(t)}{x_{targ} - x(t)}\right), \tag{5.32}$$

where the position of the target is $\boldsymbol{p}_{targ} = [x_{targ}, y_{targ}]$.

5.4 Global Collision Avoidance Algorithms

As has been previously discussed, there are several drawbacks with local methods, especially that they are very unlikely to succeed in finding an optimal path in the presence of obstacles. Global collision avoidance algorithms are at a higher hierarchical level, and are more likely to find a path in the presence of obstacles. The global methods are practical for off-line path planning based on existing information of the environment. The main reason for this is the high computational time, and thus impractical for on-line computation.

The main reason for the global methods having a higher success rate than the local methods are that they consider a larger map, which includes the entire path to the target, when generating a feasible path. Several methods exists for this purpose, and most of them are search algorithms. Such algorithms can be used for several objectives. Some of the most common algorithms will be mentioned in the following.

5.4.1 Shortest Path Algorithms

Shortest path or least cost algorithms originates from computer science, where such algorithms are wide spread. The objective is to solve the shortest path problem. The problem is solved by finding the path between two vertices in a graph, such that the sum of the weight given to the edges between all vertices connecting the the start and end vertices is minimized. For more detailed description of these types of algorithms, see Cormen et al. (2001). To clarify, vertex is a node or an intersection, an edge is the line between vertices while the graph can be seen as a map.

Dijkstra's Algorithm

Dijkstra algorithm solves the single-source shortest paths problem for a weighted, directed graph, with non-negative weights on the edges. The edges in the graph are given weights, and the sum of the weights on all edges between the start vertex and some other vertex is the cost of the path between them. For the application considered in this thesis, the weight of the edges would be a distance or time, depending on what should be minimized. If the objective is to find the shortest path between the start node and an end node, then the algorithm would simply be stopped after finding this particular path.

A* Algorithm

The A* (A star) algorithm is an extension of Dijkstra's algorithm. Especially the A* algorithm is known for getting better performance than the Dijkstra's algorithm by using heuristics. The algorithm uses a best-first search algorithm to find the shortest path between two nodes in a graph. It is based on the cost between the starting node and the current node and an heuristic

estimate of distance left to the goal. The A* algorithm is widely used, mostly because of high performance and its accuracy. The A* algorithm is complete, which means that if there exists a solution, the solution will be found. If no solution is found, then it is guaranteed that no solution exists.

Several other methods of the same nature such as breadth-first, depth-first, D* and Bellman Ford algorithm exists, but will not be elaborated any further.

5.4.2 Rapidly-Exploring Random Trees

The Rapidly-Exploring Random Tree (RRT) was introduced by Steve LaValle in LaValle (1998). The method is designed to solve path planning problems, in a sub-optimal manner. A tree is generated as the algorithm explores the configuration space. As opposed to other trees based on random walks, the RRT algorithm is not biased towards places not yet visited. The RRT does not only generate a path of workspace coordinates, but actually generates the states during the path. This means that the path will in addition to finding a path in the free configuration space, generate a feasible path with respect to the vehicles limitations. The RRT explores the configuration space faster than the shortest path algorithms mentioned above, but the RRT is not a complete method. This means that a solution will not guarantee an optimal path. In Loe (2008) an implementation of the RRT method in a hybrid collision avoidance algorithm for an UUV can be investigated.

5.4.3 Hybrid Collision Avoidance Algorithm

The apparent drawbacks with both global methods and local methods has led to research into combining global and local methods, in the hope of getting the "best of both worlds". The global method guarantees finding a path to the goal, while the local method guarantees that the vehicle will not collide in a dynamic environment. The concept has been investigated within the Unmanned Surface Vehicle (USV) community, where there are much stricter rules for being able to operate. In Larson et al. (2006) a hybrid approach to collision avoidance for USVs is suggested. It should be noted that in literature, global methods often are referred to as deliberate methods while local methods often are referred to as reactive or responsive methods.

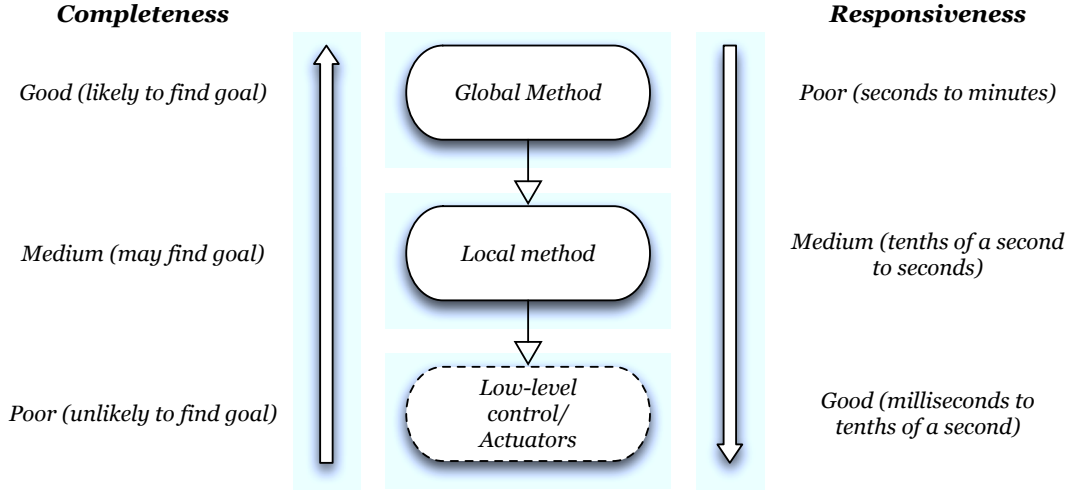


Figure 5.8: The hierarchy for hybrid collision avoidance. The chance of finding a optimal path increases upwards in the hierarchy, while the chance of avoiding dynamic, or sudden, obstacles increases downwards in the hierarchy.

The motivation for a hybrid approach is quite nicely demonstrated in Figure 5.8, where the outline is adopted from Loe (2008). Completeness was mentioned in the shortest path algorithms, which says something about the chance of finding an optimal path. Responsiveness is based on the algorithms chance of reacting when a collision is imminent.

Chapter 6

Collision Avoidance System for ROV Minerva

A collision avoidance system is implemented in the control system for ROV Minerva. This chapter will present the theory which has been applied, implementation considerations and assumptions that have been made. Some scenarios will be simulated to present the functionality of the system. Full scale experiments have been conducted to verify the system, from which results will be presented.

6.1 The Collision Avoidance System

The collision avoidance system is hierarchical placed within the guidance system, which can be seen in Figure 6.1. The collision avoidance system supplements the already existing guidance system and will take action in case obstacles are detected. The signal flow within the collision avoidance system is presented in Figure 6.2. The map is generated based on sonar data, while the

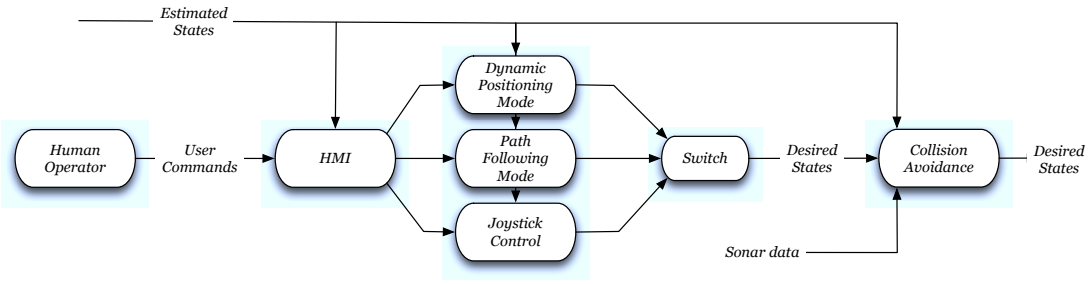


Figure 6.1: The collision avoidance system is located in the end of the signal flow within the guidance system.

map is limited by the current position. Once the map is generated, the obstacle detection block determines whether there are obstacles within the map. The anti-collision algorithm generates desired states based on the new information received from the obstacle detection. If obstacles are detected, the new desired states are sent to the controller. If not, the desired states generated by the guidance system are sent straight through. The collision avoidance system should not intervene with the defined guidance missions unless it is absolutely necessary. The collision avoidance does not guarantee a successful mission on its own, but is designed to guarantee that the vehicle does not collide, independent of the mission. A global collision avoidance algorithm has not been implemented for this work, the work done in this thesis could be a an indication of whether such an approach should be followed or not.

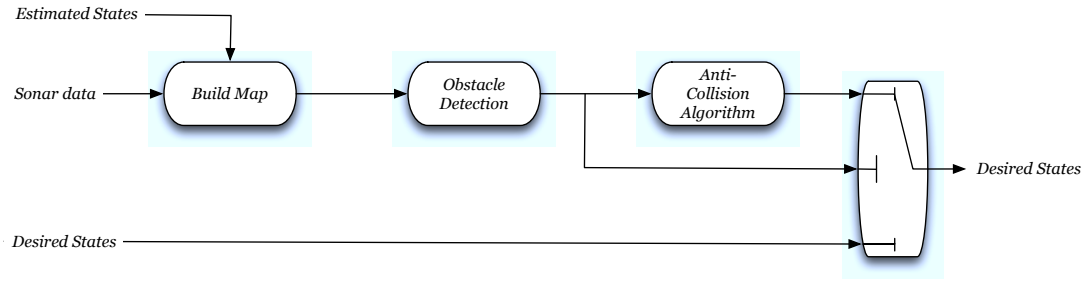


Figure 6.2: The collision avoidance algorithms should take over only if obstacles are detected within the vehicles vicinity.

The collision avoidance system is for now implemented as a stand alone guidance system, which means that it is not integrated,as desired, as described in Figure 6.1. This is during development, when the collision avoidance system is robust and redundant, then it should be implemented as described by Figure 6.1 and 6.2.

6.1.1 Map Generator and Obstacle representation

Since the ROV Minerva does not have a forward looking sonar or equivalent, a virtual map is considered. A map is generated by the author beforehand, representing an area in which the ROV is supposed to operate and avoid virtual obstacles. The map is represented using a grid of integers, where the cells are weighted from 0 to 10, where 0 represent no obstacles and 10 represent an obstacle with absolute certainty. Each cell represents a 0.5×0.5 metre area. The precision of the map, and consequently of the algorithm, would increase if the size of the cells is decreased.

This pre-generated map replaces the sonar input in Figure 6.1 and Figure 6.2. The *Build map* block consist of generating a sub-map based on the current position of the ROV. The sub-map consists of a grid of 20×20 cells which corresponds to 10×10 metres. The sub-map will be denoted as the active window, as it is the part of the map that is considered by the VFH algorithm.

6.1.2 The Vector Field Histogram Algorithm

The collision avoidance algorithm which has been chosen for implementation is the Vector Field Histogram method as previously described in Section 5.3.3. The algorithm has been chosen because of its intuitive nature, and since it has been shown to take care of some of the problems experienced with general potential field methods.

A big challenge with the VFH method is tuning. The method does not consider the physical limitations of the vehicle, and more importantly, it does not consider the size and shape of the vehicle. For this reason, a lot of testing needs to be done in order to tune the algorithm for the specific vehicle. For ROV Minerva this is simpler than for larger vehicles and vehicles with less actuation and slower dynamics. Since the ROV Minerva is fully actuated within its work space, and also has fast dynamics, it is possible to obtain fairly good results without too much time consumed on tuning.

The map is divided into sectors of size $\alpha = 5$ which produces 72 sectors. Each of the cells generate a repulsing force based on the certainty value assigned to the cell. The parameters are

chosen or calculated as follows

$$\begin{aligned} \alpha &= 5, && (6.1)\\ d_{max} &= \sqrt{2}(5^2+5^2), && (6.2)\\ a &= 10, && (6.3)\\ b &= \frac{a}{d_{max}}, && (6.4)\\ s_{max} &= 12, && (6.5)\\ l &= 2, && (6.6)\\ h_m &= 100, && (6.7) \end{aligned}$$

where s_{max}, l and h_m are considered tunable parameters. An additional tunable parameter has been added when calculating the force in order to have some tuning possibilities. The repulsive force generated by each cell is calculated as in 5.24.

$$m_{i,j} = \left(C^*_{i,j}\right)^2 (a - bd_{i,j}) \tag{6.8}$$

The magnitude of the force is dependent on the distance between the cell and the vehicle, where the criteria in 5.25 ensures that the cell furthest away generates zero force, while the force increases linearly for the cells in between. The second dependency is the certainty value $C^*_{i,j}$, which is an integer between 0 and 10. The gain $M_c > 0$ has been introduced as follows

$$m_{i,j} = M_c \left(C^*_{i,j}\right)^2 (a - bd_{i,j}), \tag{6.9}$$

which gives the possibility of investigating how the algorithm is dependent on the certainty values in the map. This value will initially be set to 100 which will make the certainty value range from 100 to 1000. This will assure that the cells which are occupied actually are considered as obstacles.

Since the collision avoidance system for now is a stand alone guidance system, an extension has been added to the VFH algorithm. The algorithm does not consider a scenario where no obstacles are found, which will make the algorithm fail. To cope with this, a simple check is done, where if the active window is obstacle free, the steering direction is chosen as the direction to the target. A more important extension which has been made, is for the scenarios where there are obstacles, but not in the path of the ROV, for instance if the obstacles are behind the ROV. As it is now, the algorithm would chose a steering direction based on obstacles being in the active window. If the object is behind the ROV, it would actually try to stay close to the object in order to avoid colliding with it. A simple way to avoid this is implemented. If s_{max} sectors are free to each side of the direction of the target, the path towards the target is considered clear. So the ROV will follow the outline of an obstacle until the path towards the target is clear with 2_{smax} sectors.

Way-points

The path the ROV should follow consists of way-points, where the target always is the next way-point in the way-point database. The way-points will be chosen based on what type of behaviour is wanted from the algorithm. Switching between way-points has been implemented as a circle of acceptance criteria as in (3.33). The acceptance radius is chosen as $R = 0.5$ metres.

6.1.3 Reference Models

One of the drawbacks with the VFH algorithm is that the heading assignment will experience steps of varying magnitude. Step inputs are in general not desirable for the controller. In order to

avoid step inputs reference models for heading and velocity have been implemented to smooth out the desired signals.

As explained previously, the last cruise in May, was shortened by a damaged fibre-optic cable. This also ruined the plans of testing the collision avoidance system with reference models implemented in the full scale system. This means that all the full scale results have step inputs, which will be addressed more in Section 6.3.

Reference models have however been implemented in the Matlab system, such that simulation with this feature has been done, which will give an indication of the effect of being able to operate with smooth heading signals instead of steps. The heading reference model is implemented as a 3rd order reference model with natural frequency $\omega_\psi = 0.8$ and critical damping $\zeta_\psi = 1$. The surge reference model is implemented as a 2nd order reference model with $\omega_u = 0.8$ and $\zeta_u = 1$. Both reference models are implemented such as described in Section 3.1.2.

6.1.4 Implementation

The collision avoidance system is, as the rest of the guidance system, implemented in Labview for the full scale control system. The simulation based system is implemented in Matlab/Simulink, and it is also possible to do Hardware-in-the-Loop (HIL) testing. The simulations are run with a model of the ROV Minerva. The model is developed by Marianne Kirkeby, and a thorough description of the model is found in Kirkeby (2010a).

6.2 Simulation-based Experiments

In order to test the VFH algorithm, several simulations have been done to verify its functionality and tuning parameters. Simulations does "always" reveal nice results, since the scenarios are made to show some particular situation. This is also the case for these simulations, they are done to give the reader a general idea about how the collision avoidance system implemented works. Limitations will also be addressed. The simulations are done in Matlab/Simulink, specifically set up for each scenario.

6.2.1 Step Inputs in Heading vs Smooth Heading Reference

As explained, the reference models have not been successfully tested during full scale tests, but the difference between using reference models as opposed to not using them will be demonstrated through simulations. Only one obstacle is considered for the first simulations. The collision avoidance system has to take action to ensure that the ROV does not collide with the obstacle. The trajectory of the ROV is presented in Figure 6.3. The ROV initially sets course towards the target, but when the obstacle enters the active window of the collision avoidance algorithm, the algorithm calculates steering directions such that the ROV travels around the obstacle and is manoeuvred safely to the target destination.

The heading during the test can be seen in Figure 6.4. The heading assignment has not been filtered so as expected there are several steps in desired heading, off different magnitude, which the ROV is unable to follow closely. The objective is reached despite this, due to the ROV Minervas fast dynamics.

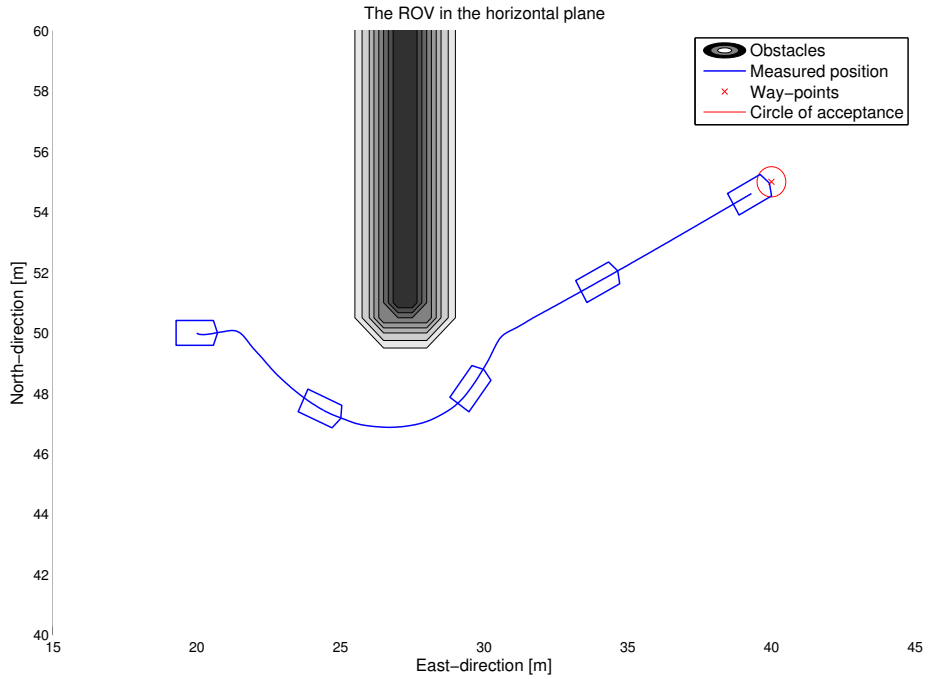


Figure 6.3: Horizontal view of the ROV moving around an obstacle.

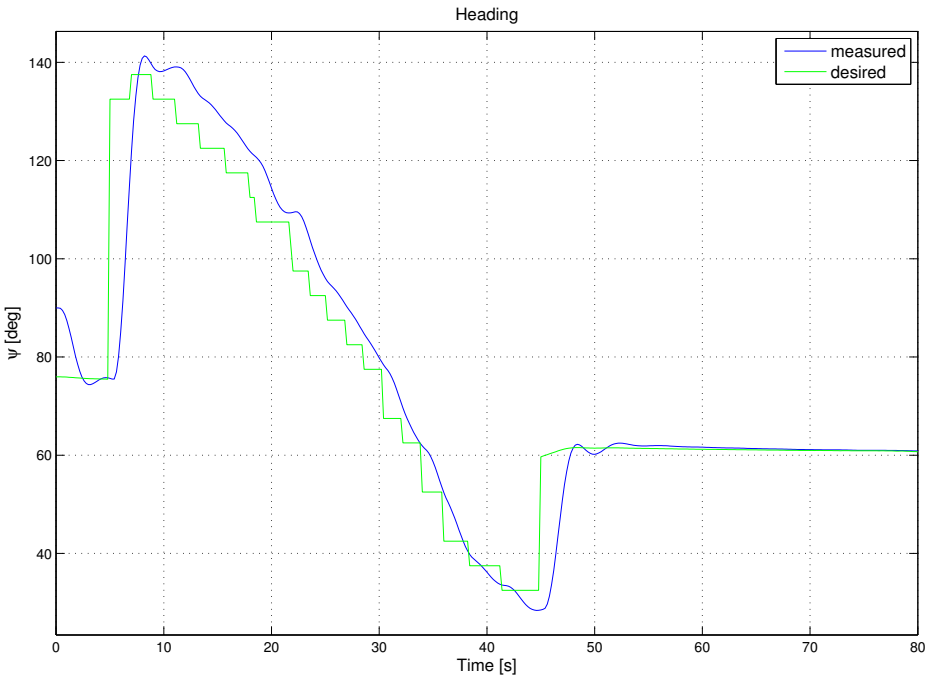


Figure 6.4: Heading for the ROV while moving around an obstacle.

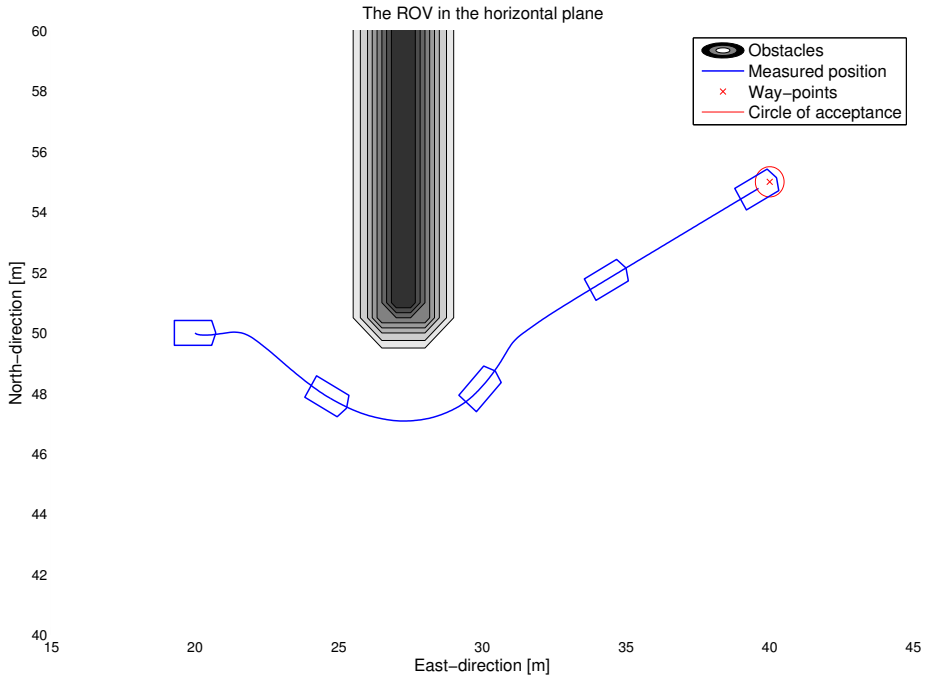


Figure 6.5: Horizontal view of the ROV moving around an obstacle, with smooth heading reference.

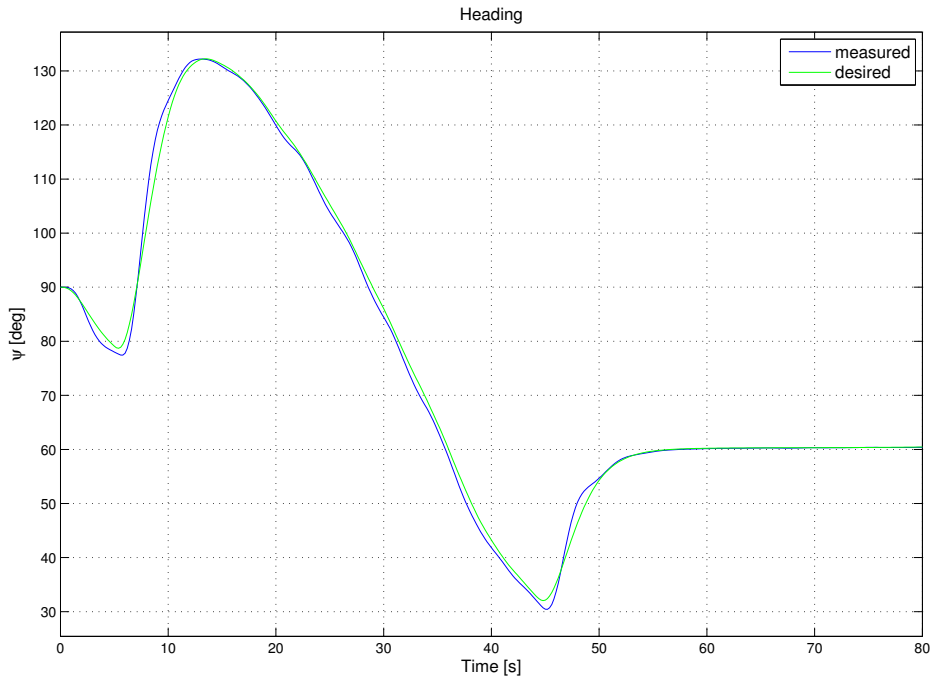


Figure 6.6: Heading for the ROV while moving around an obstacle, with smooth heading reference.

From this point on all simulations will be done with heading references that have been fil-

tered through a 3rd order reference model. In Figure 6.5 the trajectory of the ROV can be seen for the same scenario as above. The trajectory does not reveal a particularly difference from the previous simulation, but if the heading reference in Figure 6.6 is considered the difference is actually quite prominent. The ROV is able to follow the desired heading very closely, and no overshoots and oscillations of a substantial order occur, which makes the mission in itself both less energy consuming, and safer.

6.2.2 Selecting the Maximum Sectors Parameter

The results revealed in the two first simulation shows that the ROV is moving quite close to the obstacle. The parameter which is the most significant in that respect is the maximum considered sectors parameter, s_{max}. To recapitulate, this parameter decides how many sectors should be considered if the valley is to big to consider the entire valley. So in the scenarios simulated above, this is set to $s_{max} = 12$ as previously stated. This means that the steering direction will be $s_{max}/2 = 6$ sectors into the valley, which again corresponds to $6\alpha = 30$ degrees. The maximum number of sectors considered is increased to $s_{max} = 18$ such that the steering direction will be $9\alpha = 45$ degrees away from the obstacles, which should increase the distance the ROV is keeping to the obstacles. This assessment of the s_{max} parameter has been purely theoretical. In practice the angle from the obstacle direction will be dynamic, such that it should only be considered as a reference for tuning. The same scenario is simulated again with the new configuration and the ROV can be seen in Figure 6.7, while the heading is seen in Figure 6.8.

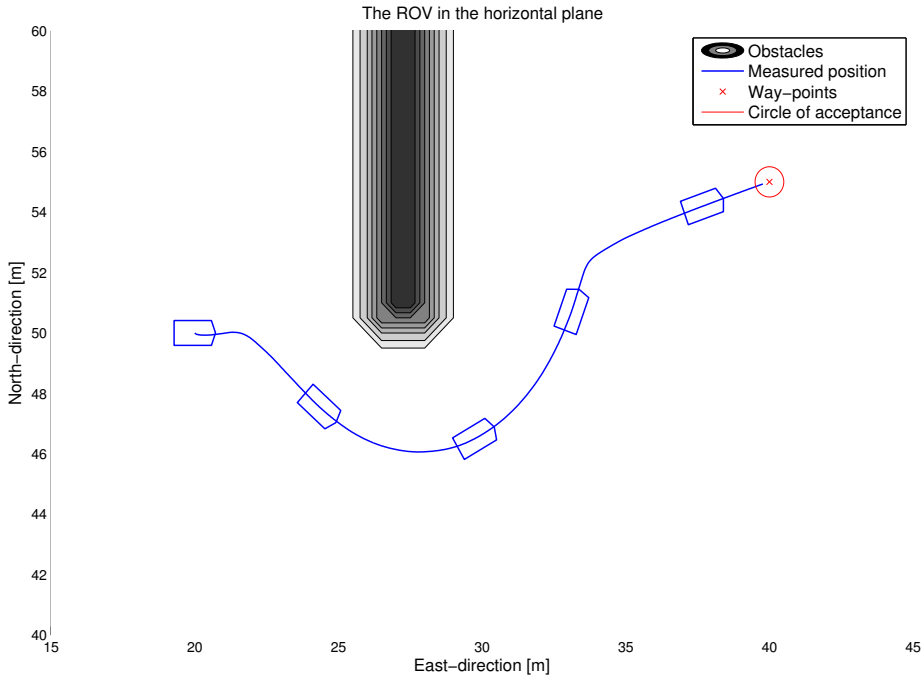


Figure 6.7: Horizontal view of the ROV while moving around an obstacle with increased $s_{max} = 18$.

The ROV is keeping the obstacle at a safer distance, which was the intended result. The heading demonstrates that the ROV is taking longer time and has higher maxima and lower minima in heading which actually increases the ROVs ability to follow the reference signal as well. The parameter s_{max} will be set to 18 for the remaining simulations.

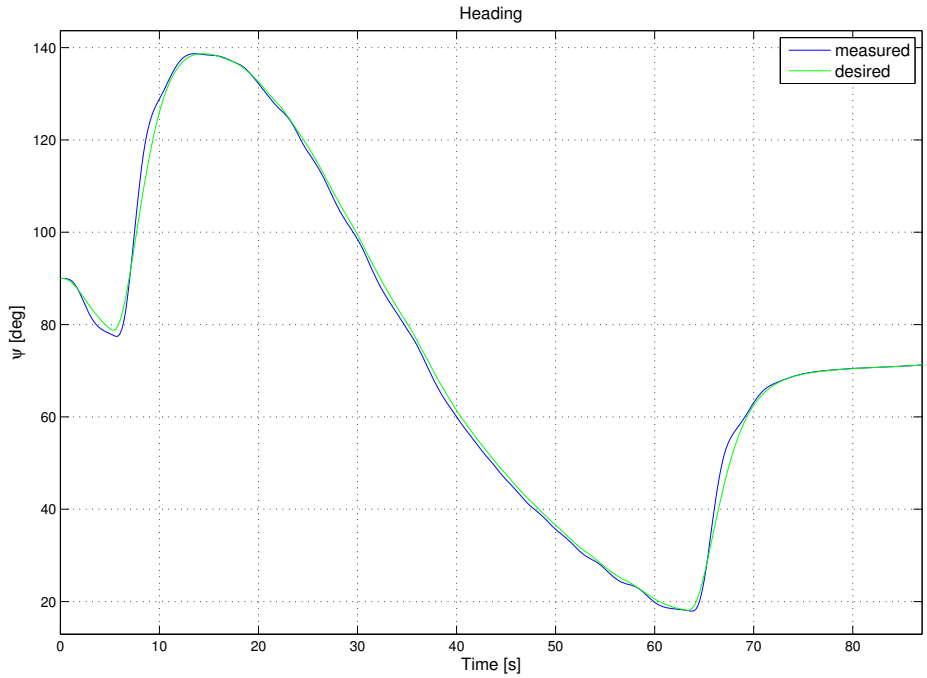


Figure 6.8: Heading for the ROV while moving around an obstacle with increased $s_{max} = 18$.

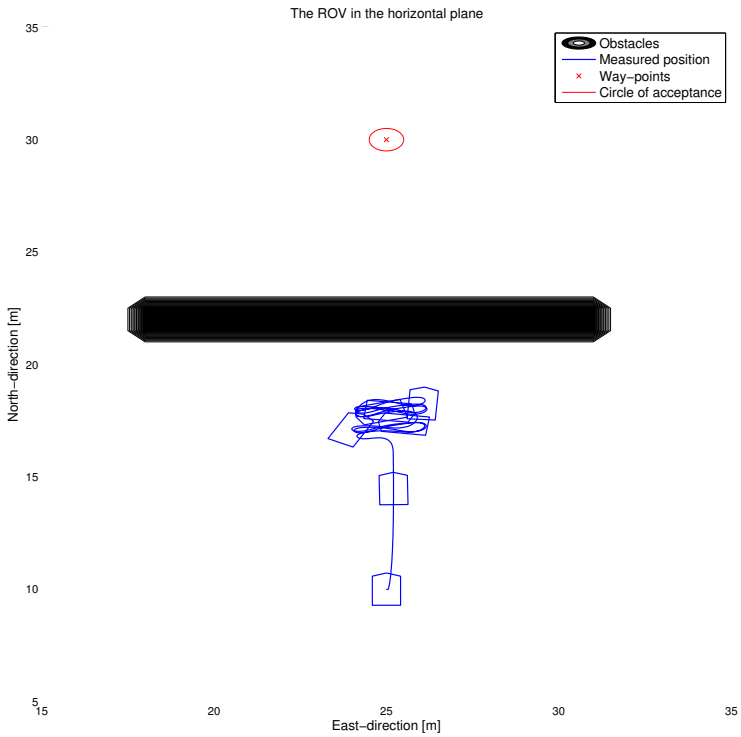


Figure 6.9: The ROV in a trap situation.

6.2.3 Trap Situation

As discussed in Section 5.3.3, trap situations is one of the most common reason for unsuccessful mission completion. A "watch dog" should be implemented in order to raise a flag once a trap situation is detected. For this thesis, such an implementation has not been done, but to demonstrate a trap situation a simulation designed to invoke a trap has been simulated. The result can be seen in Figure 6.9

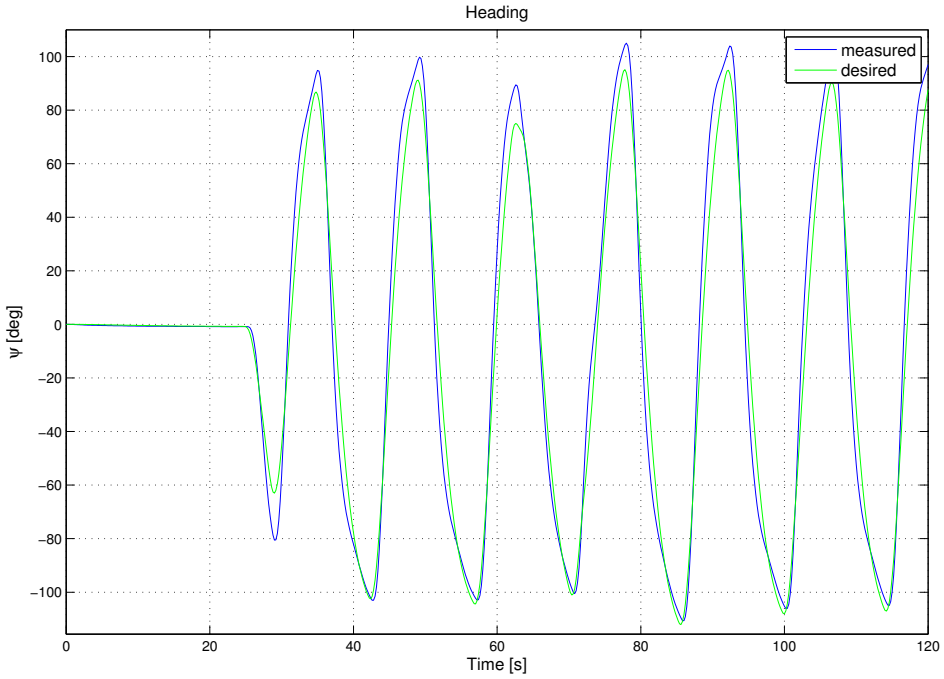


Figure 6.10: Heading for the ROV while in a trap situation.

The ROV is not able to find a way around the obstacle, due to the fact that after going to west for a short time, it will turn and go back since that direction is closer to the direction of the target. This will happen again when it is moving towards east and so fort. This can also clearly be seen by looking at the heading in Figure 6.10. A simple watch dog would be to monitor which direction the algorithm chooses first, which is set as the direction of travel. If this is violated at some point, a flag should be raised, that indicates a trap situation, and a new path must be generated. For this particular scenario, a simple solution would be to raise a flag and introduce a new way-point, for instance 10 metres west of the current position, when the algorithm tries to move eastwards after initially moving towards west. When it reaches this point, it should try heading for the initial target again. This could be an iterating procedure depending on the length and size of the particular obstacle.

For a ROV with an operator, the iterating process is generally not a problem, because the operator could at any time abort the mission and return to support vessel. For an AUV this is different, in that case a supervisor should calculate energy consumption such that the AUV is not searching along an obstacle for an undetermined amount of time. At some point, it should start searching towards the east of the initial trap situation for a solution, or in case of shortage of energy mission should be aborted and retrieval mode should be activated.

6.2.4 Travelling Between Two Obstacles

The scenarios considered so far have consisted of only one obstacle. In some cases the vehicle might have to be guided between two or more obstacles which makes things a little more advanced. In Figure 6.11 the ROV can be seen guided between two obstacles. In this scenario the maximum sectors parameter s_{max} will not effect the ROV is moving through the opening, as the number of sectors of the opening is lower than s_{max}. The distance to the obstacles can not be increased, as it only would push the ROV towards the other obstacle. The algorithm is guiding the ROV nicely through the opening, and thereafter to its destination.

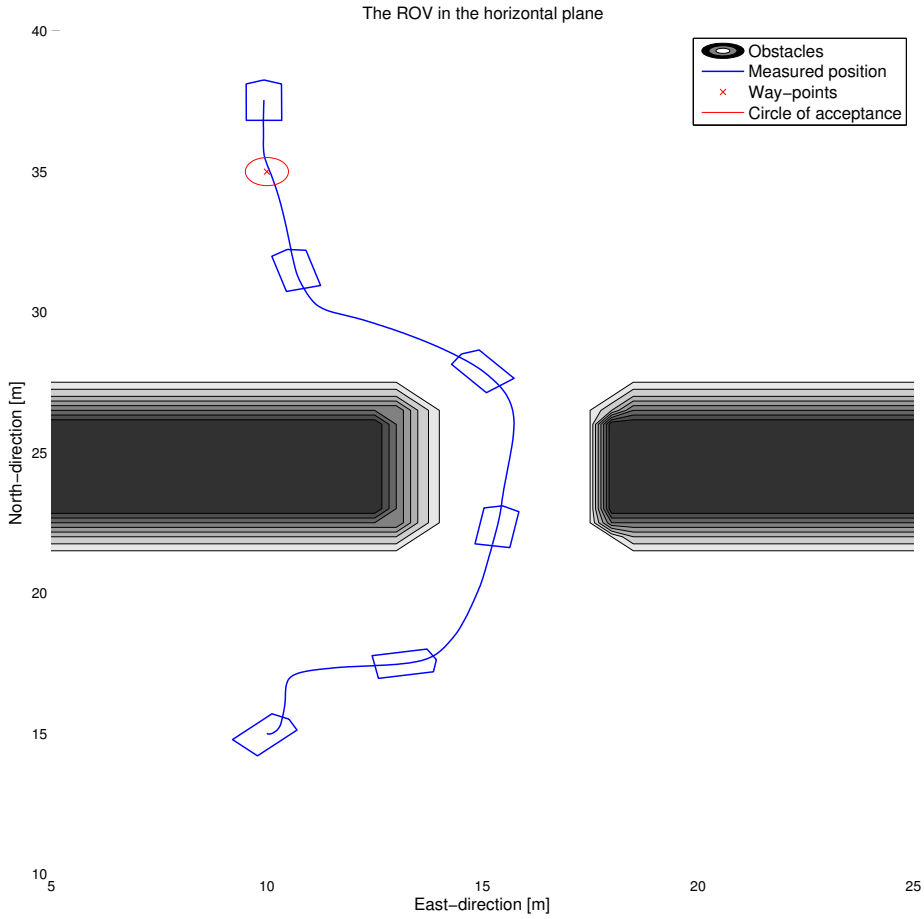


Figure 6.11: Horizontal view of the ROV moving through an opening between two obstacles.

The situation is critically dependent on the heading, if the ROV is unable to keep the desired the heading, a collision would likely occur. The heading can be seen in Figure 6.12, and since the reference is smooth the ROV is able to keep the desired heading which in turn makes sure that the ROV is safely guided to its destination. It should be noted that the way-point is the first way-point. Once the ROV enters the circle of acceptance, the ROV is travelling towards the next way-point.

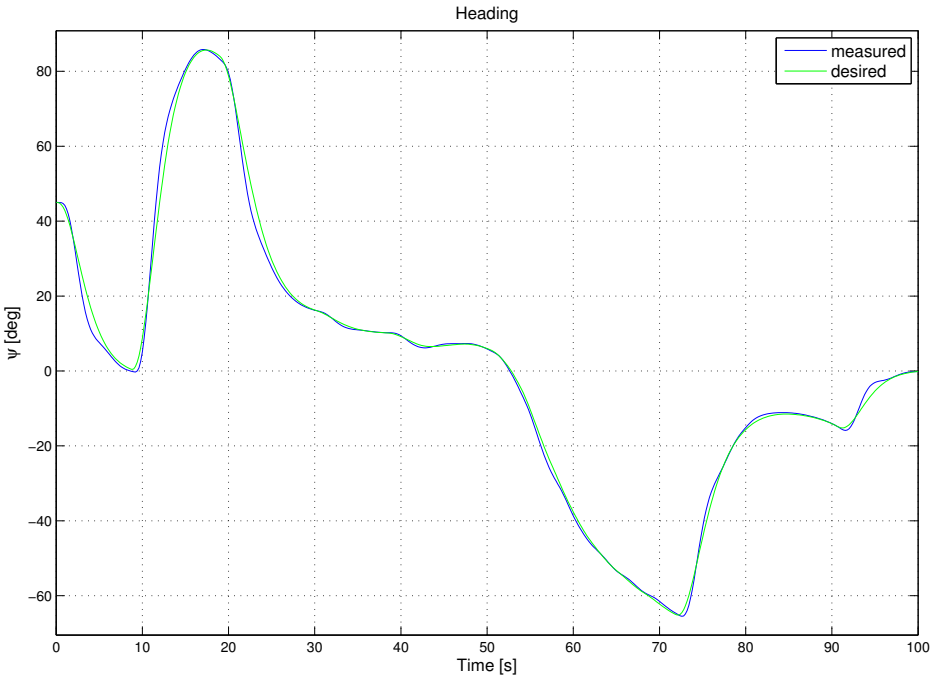


Figure 6.12: The heading of the ROV while travelling between two obstacles.

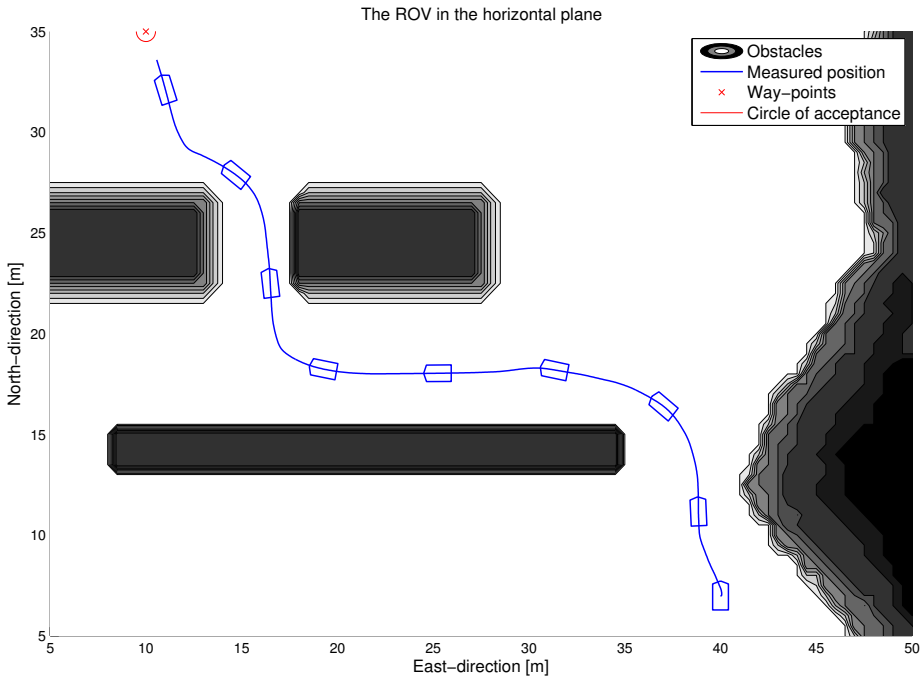


Figure 6.13: The ROV travelling through narrow corridors.

6.2.5 Narrow Corridors and Several Obstacles

A nice property of the VFH algorithm is that it does not provoke oscillations when moving through narrow corridors and in the presence of several obstacles. This will be demonstrated in the last two simulations.

In Section 6.1.2 a parameter M_c was introduced and set to $M_c = 100$. This has as shown in the above simulations worked well. Now however the ROV is travelling through densely occupied environment which has shown that by using to high certainty values will make the algorithm to tight. What is meant by this is that it will for instance in some cases not consider a valley or see the valley as much narrower than it is due to the filtering of the histogram. For the remaindering simulations the certainty values will be considered as they are in the map, i.e. $M_c = 1$.

Figure 6.13 shows a scenario where the ROV has to move through narrow corridors and also exits the corridor through a "door". The ROV is able to avoid colliding with the obstacles, but the closest distance to an object is approximately 0.5 metres, which if a safety radius is considered, probably would not have been allowed. A modification of the algorithm should be made such that it does not only consider the number of sectors in a valley, but also calculates the width of the valley to be able to rule out some valleys if they are too narrow. Considering that the ROV is 1.44 metres long, it needs at least 0.72 metres at each side in order to do a 180 degree turn and go back. This is obviously not feasible, it should at least have 2 metres to each side to such a manouver. A safety radius for the ROV Minerva should probably be higher than 3 metres, in order to be safe in case of high speed currents.

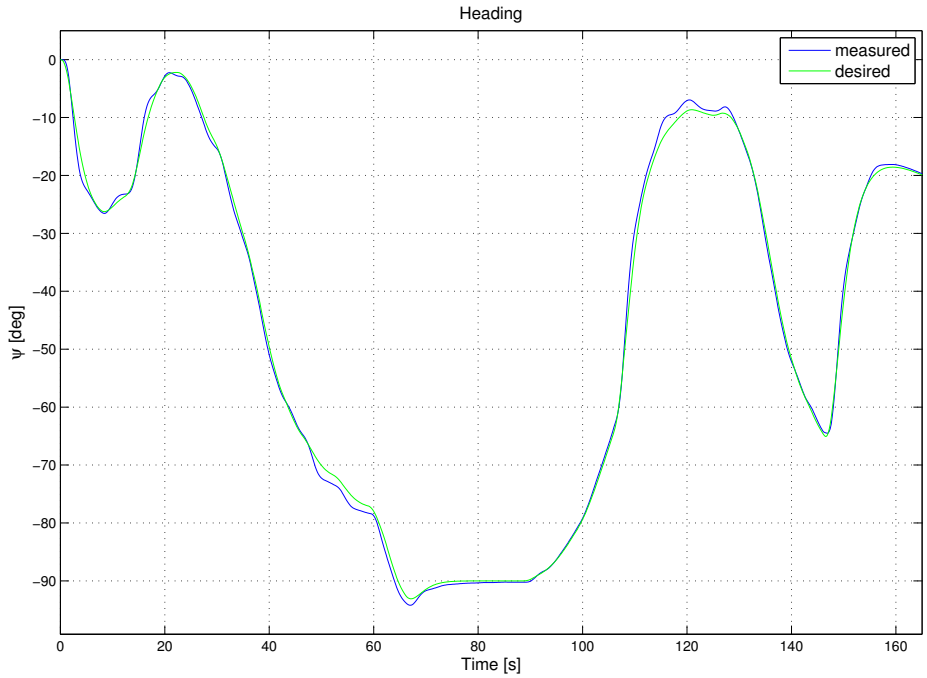


Figure 6.14: The heading while travelling through narrow corridors.

An even more difficult scenario is given to the ROV in Figure 6.15, but it is also here able to avoid collision. It should be noted that the algorithm would fail in this scenario if not several way-point had been defined which the local algorithm is able to find a path between. This demonstrates implicitly that in some cases, if the path is not made up of feasible way-points, the algorithm will fail.

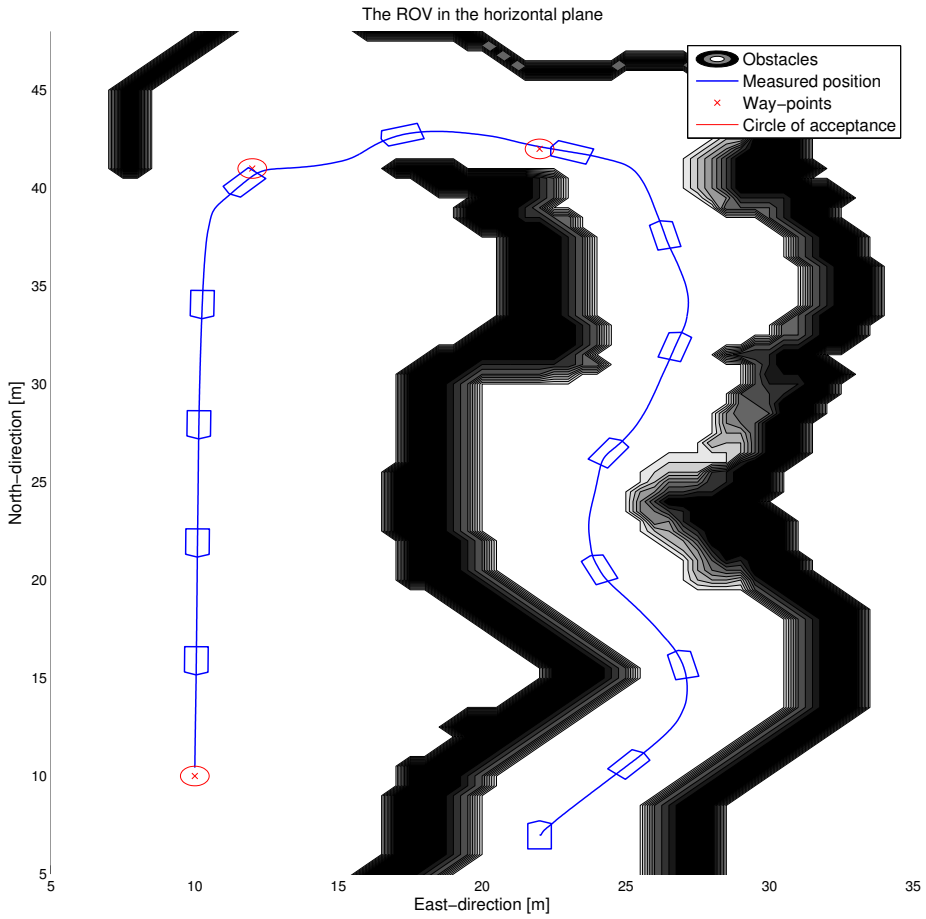


Figure 6.15: The ROV travelling through a canyon-like environment.

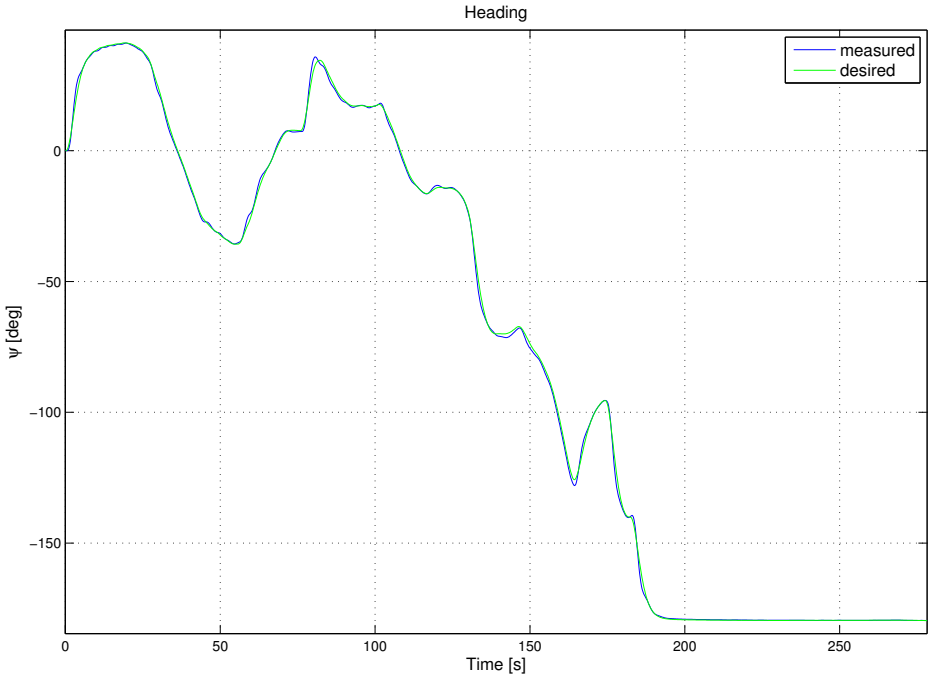


Figure 6.16: Heading while travelling through a canyon-like environment.

6.3 Full Scale Experiments with ROV Minerva

Initial full scale testing of the collision avoidance system was conducted in May 2011 in Trondheimfjorden with ROV Minerva deployed from RV Gunnerus. The objective with the experiments is to find out if the VFH algorithm could work with the full scale ROV, and whether this approach should be investigated further or abandoned. The tests consists of one or two obstacles that the ROV Minerva should avoid. The surge speed for all the full scale tests on collision avoidance is set to 0.3 m/s. The experiments on collision avoidance was done with $s_{max} = 12$ and $M_c = 100$ as was initially considered. The biggest difference will be the closeness to the obstacles for wide valleys.

6.3.1 Scenario 1

The first experiment which is presented considers exactly the same scenarios as for the first three simulations in the last section. There is one obstacle which is between the ROV and its destination. In order for the ROV to get to the destination the collision avoidance system has to generate heading assignments such that the ROV travels around the obstacle. The behaviour of the ROV can be seen in Figure 6.17. A trajectory as seen in Figure 6.7 is the expected ideal trajectory, and the full scale trajectory is close to that. Obviously the signals are more distorted than for the simulation as expected, but generally the result is quite satisfactory. The ROV is within 2 metres from the obstacle. This distance should be increased if the s_{max} parameter is increased as demonstrated in 6.2.2.

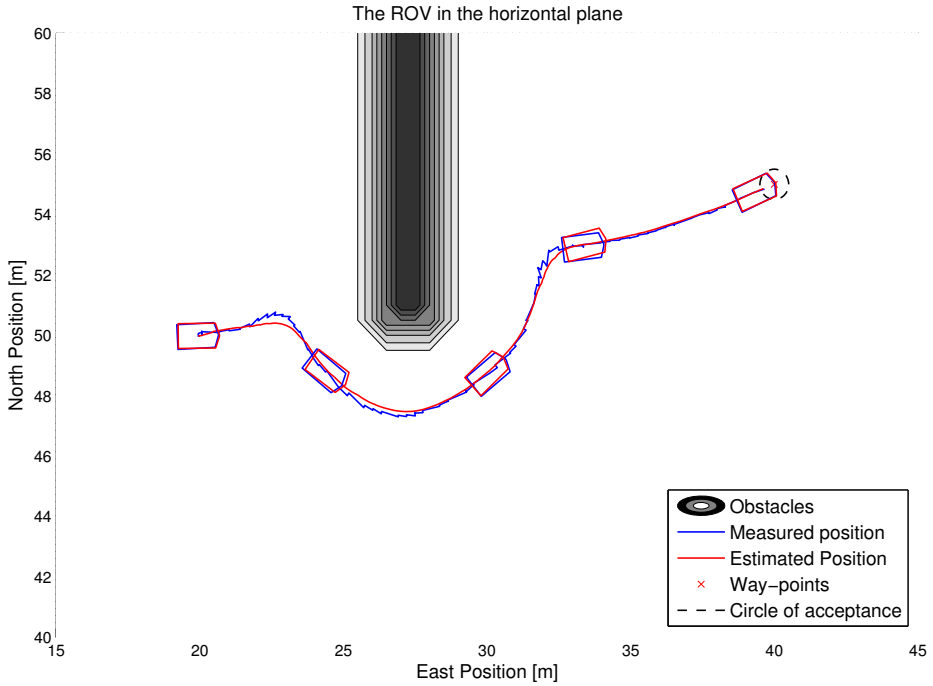


Figure 6.17: Full scale experiment where ROV Minerva is travelling around an obstacle.

The heading is a point of particular interests as the heading reference consists of steps of different magnitudes, as seen in the first simulation. This should generally make the motion of the ROV less stable. The heading can be seen in Figure 6.18 and as expected the ROV is not able to follow the largest steps, but the smaller steps is handled well. Especially the steps at approximately 6 and 49 seconds is hard on the ROV.

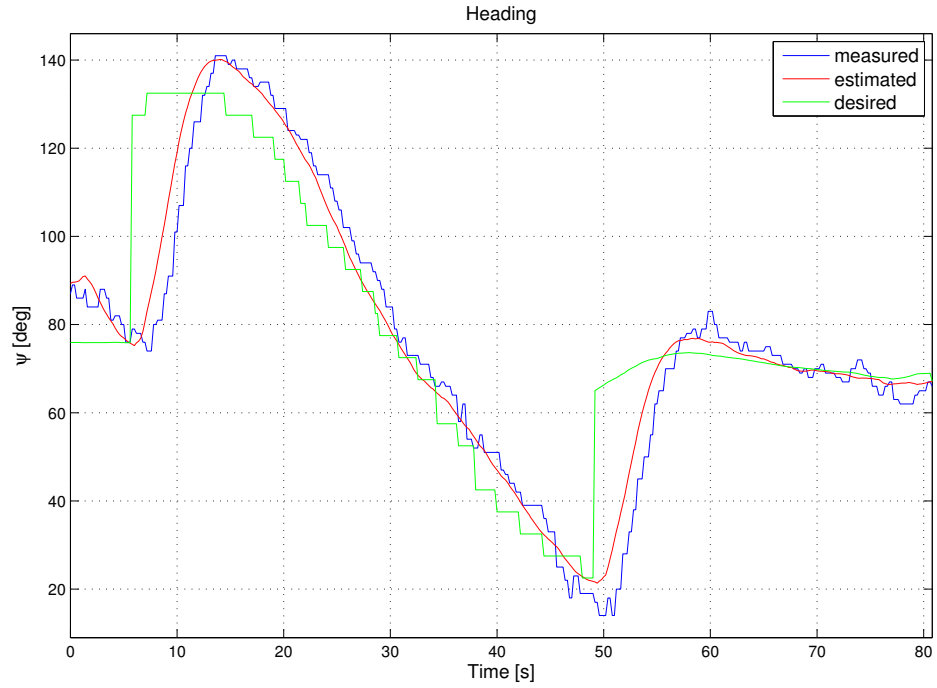


Figure 6.18: Heading while moving around an obstacle.

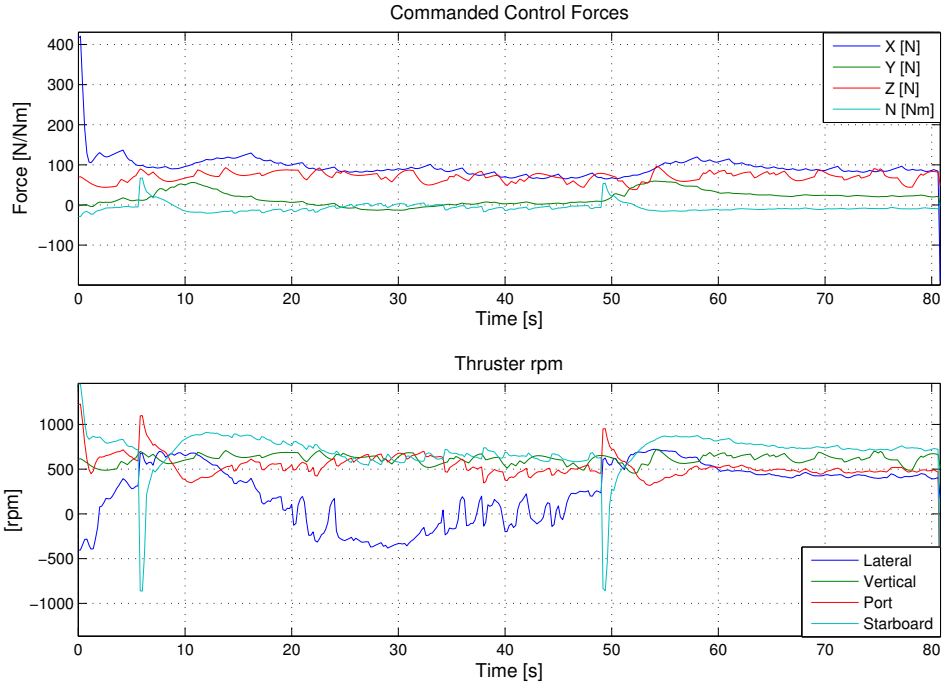


Figure 6.19: Thruster RPM while moving around an obstacle.

Wear and tear can be seen by the input on the thrusters in Figure 6.19, where the starboard thruster gets massive step inputs at the same point at which the large steps in heading is occurring. This is because the ROV needs to counteract the lateral motion induced by the fast

changing heading. The inputs stay within the limits of the thruster, which in Chapter 1 was described to be 1450 rpm. Complete results for this full scale experiments can be investigate in Appendix A.3

6.3.2 Scenario 2

The second test is a little more challenging for the collision avoidance system, where it has to guide the ROV through a narrow space between two obstacles. The scenario is the same as the scenario simulated in Figure 6.11. The trajectory of the ROV is presented in Figure 6.20. The ROV can be seen going very close to the left obstacle, and is grazing the tip of the obstacle. This means that this test actually failed its main objective, to avoid collision. But there is information to be gotten from all experiments, failed or succeeded. If we take a closer look it can be seen that the ROV is oscillating when going through the opening, which in turn imperils the ROV. If the heading in Figure 6.21 is investigated, it can be seen that the heading overshoots the desired heading by 10 degrees for estimated heading and almost 20 degrees for measured heading. This is the main reason why the experiment failed, since it can be seen that right after entering the corridor the ROV is seen turning towards the left obstacle, this is due to an overshoot in heading. This could be improved by having filtered heading references, which might would have made the test successful.

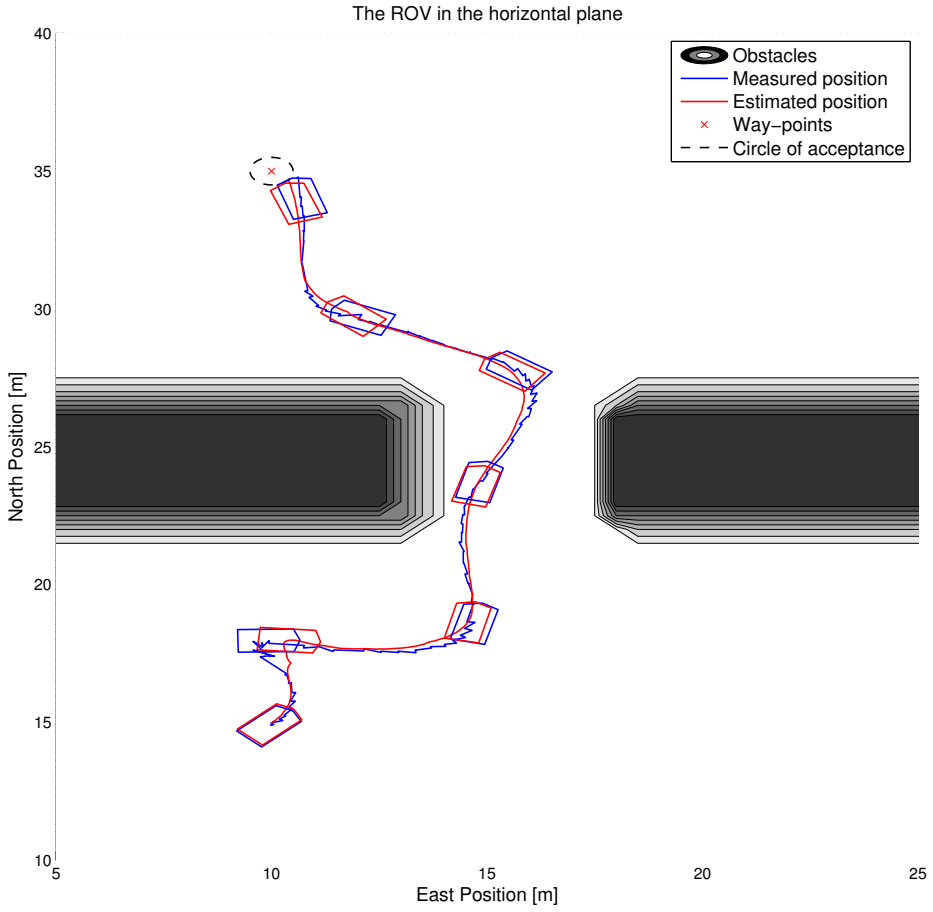


Figure 6.20: Full scale experiment where ROV Minerva travels between two closely spaced obstacles.

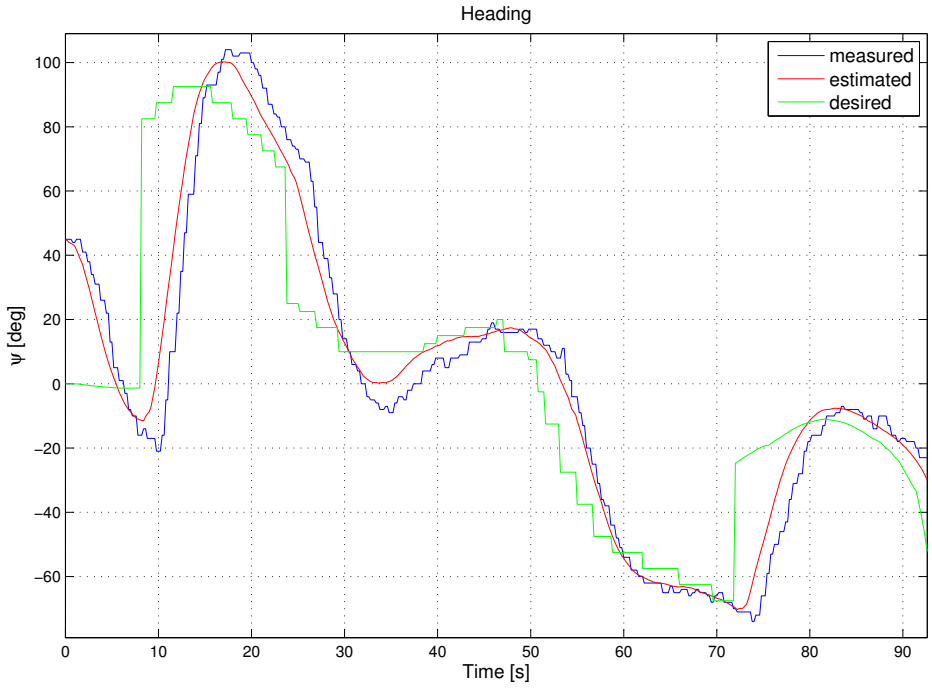


Figure 6.21: Heading while travelling between closely spaced obstacles.

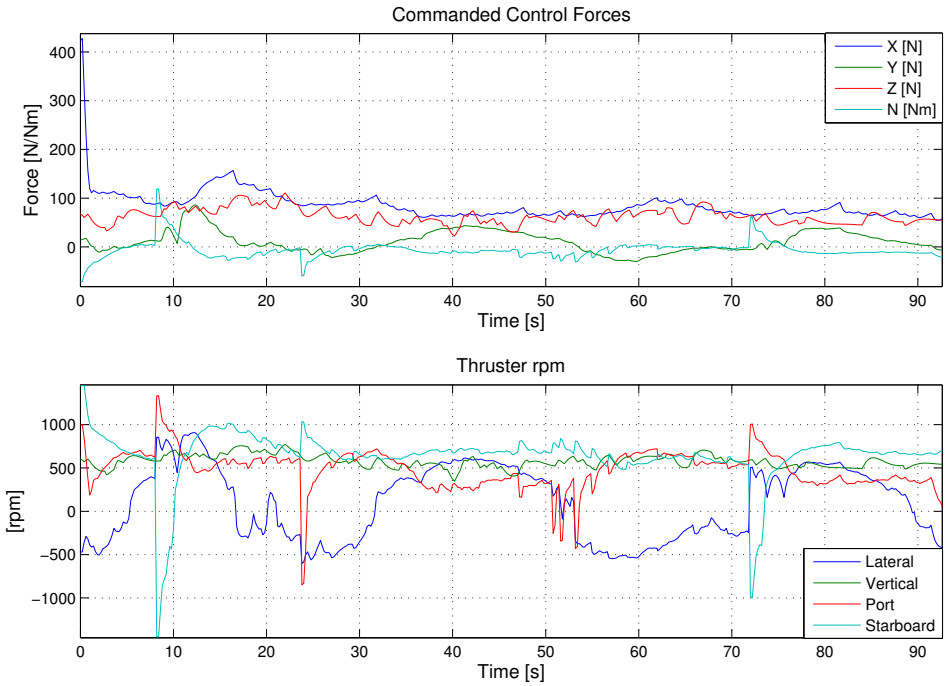


Figure 6.22: Thruster RPM while travelling between closely spaced obstacles

The tendency seen in the first scenario with the heading steps giving the ROV impossible thrust inputs is being confirmed with this test. The thrust inputs is shown in Figure 6.22. It can be seen that the starboard thruster even is saturated at approximately 8 seconds, while the con-

nection between heading steps and steps in starboard and port thruster is clear for this scenario as well. As discussed in Section 6.2.5, the width of the opening should perhaps been considered too narrow for the ROV to travel through.

6.3.3 Scenario 3

In the final full scale result the ROV is once again challenged with a scenario where it optimally should travel between two obstacles. The trajectory of the ROV is shown in Figure 6.23. The algorithm could choose to make the ROV travel around the "islands", but as desired, the algorithm chooses the valley between the two "islands". As opposed to the previous test, this mission must be classified as successful. It takes a route which is more or less through the middle of the valley, which is the intuitively preferred route. The ROV is initially not noticing any of the obstacles due to the active window size of 5 metres, and is therefore not correcting steering direction with respect to the obstacles until the first obstacle, the obstacle at the bottom, enters the active window. When the first obstacle enters the active window the steering direction is changed towards right of the initial heading. When the second obstacle then enters the active window, the steering direction is corrected to go approximately in the middle of the two obstacles. When the coast is clear, the ROV can safely steer towards its destination.

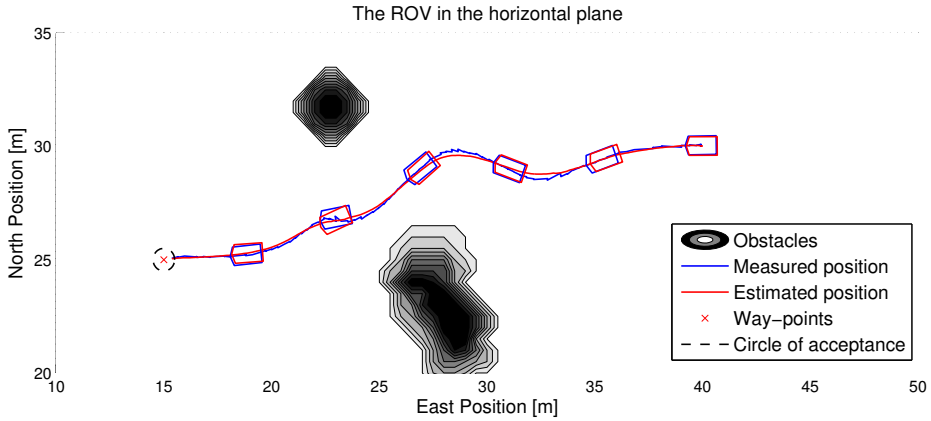


Figure 6.23: The ROV is guided between two "islands" by the collision avoidance system.

The heading is shown in Figure 6.24 and the thrust input in Figure 6.25. The apparent connection between step inputs in heading and steps for the starboard and port thruster input is again noticed. As explained in Section 6.1.2 the algorithm will make the ROV follow an obstacle until s_{max} sectors are free to each side of the direction of the target. This is the reason for the step in heading at 57 seconds. The ROV wants to go south after coming through the passage, but at 60 seconds the free path criteria is fulfilled, and a second step to change the heading towards the destination can be seen. This problem could be less apparent with smoothed heading references, but ideally a better solution to this problem should be found.

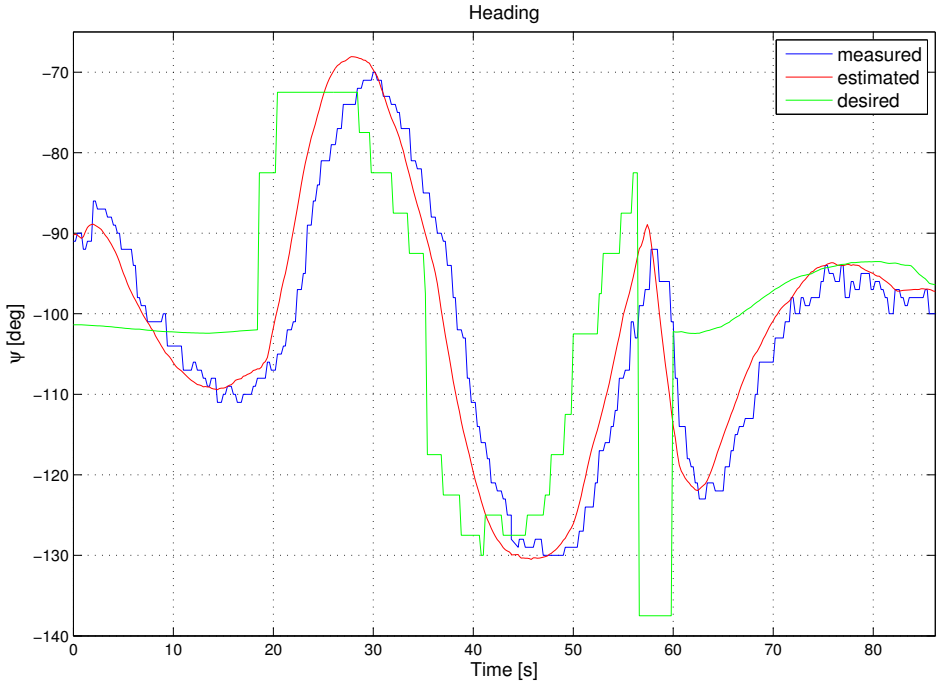


Figure 6.24: The heading while the ROV is travelling between two "islands".

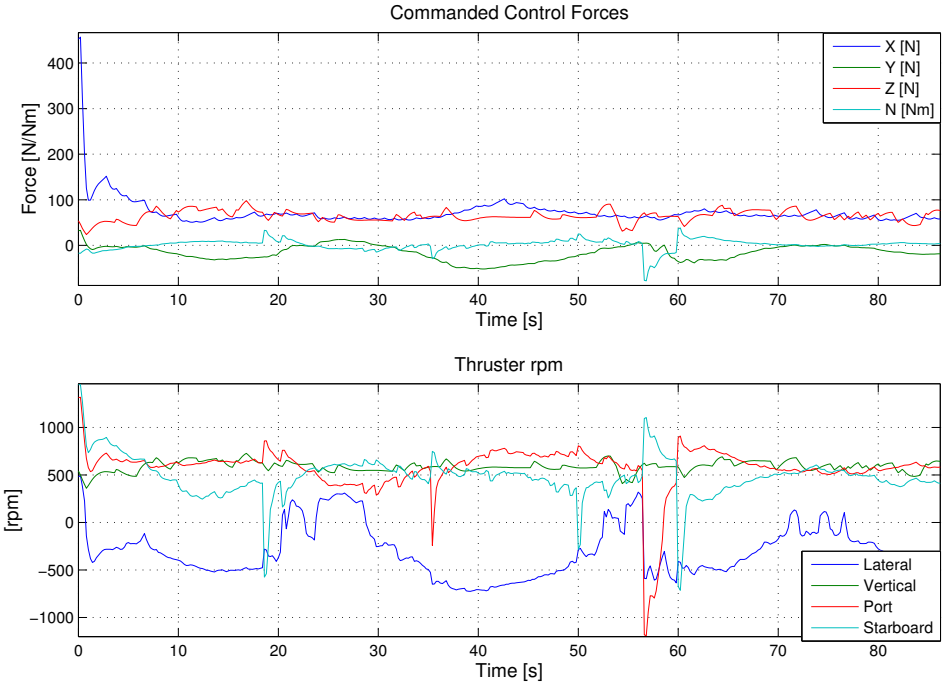


Figure 6.25: The thrust inputs while the ROV is travelling between two "islands".

6.4 Discussion

Several results have been presented in this chapter, based both on simulation based experiments and full scale experiments. The simulation results have as mentioned been made to demonstrate and highlight positive and negative aspects of the system. The full scale tests have been made to see if the collision system has potential to be further developed with respect to the ROV Minervas dynamics.

The simulations have shown that the VFH algorithm can be difficult to tune, especially since it does not take into account the physical limitations or the size and shape of the vehicle. It has, however, been shown that if tuned correctly, the collision avoidance system will work nicely within its limitations. The problem with the trap situation has been demonstrated and discussed, where also possible solutions to the problem have been discussed. Simulations in clustered environment have been done to show that the ROV does not experience oscillations in such scenarios which is a nice quality of the VFH algorithm.

The full scale experiments have especially highlighted the problems that arise if the steering assignment given by the VFH algorithm is not smoothed. Despite this problem, the full scale experiments have been successful, and should perform quite well when full scale tests is done with a reference model for the steering assignment. This has been implemented but has unfortunately not been tested full scale yet.

The system has been purely based on a local approach to collision avoidance. A global extension to the existing collision avoidance system could be tested in the future, based on predefined maps of a mapped area. Also a hybrid approach to collision avoidance would be interesting to explore. The field of collision avoidance is an important part when working with autonomous underwater vehicles, and the field should be further explored with the ROV Minerva to get a fully implemented collision avoidance system working.

Chapter 7

Concluding Remarks

A lot of work has been done during this master thesis, spanning over quite large research fields. Some of the parts has been quite successful, while other parts have had some shortcomings. In this final chapter, the work done will be evaluated and recommendations for further work will be presented.

7.1 Conclusion

This thesis has given a description of different guidance principles and guidance strategies. In particular the principle of following a path has been dealt with. Several aspects of following a path have been discussed. It has been shown that representing a path in different manners can help in optimizing path traversal with respect to different criteria. It has been shown that when travelling along arcs, it is important to reduce forward speed such that the vehicle do not drift to much. Reducing forward speed will reduce total energy consumption of the mission.

A path following scheme has been implemented for ROV Minerva, and full scale experiments have been conducted in the Trondheimsfjord. The experiments show that the functionality of the scheme works well for different patterns. Especially, paths represented using straight lines and circular arcs have been considered. Using this principle the ROV Minerva has been shown following lawnmower patterns with high precision. Different strategies for keeping cross-track error low has been investigated, and the conclusion is that the best way to do this is to reduce forward speed while travelling along arcs.

A description of collision avoidance aspects has been given. Relevant theory within this field has been presented, and relevant considerations have been made. A collision avoidance system based on the Vector Field Histogram algorithm has been implemented for use with the ROV Minerva. Simulations of the algorithm has shown its strength, but also addressed its weaknesses. It has been made clear that parameter tuning is difficult with respect to the physics of the vehicle. Initial full scale experiments with ROV Minerva show that initial work with the VFH algorithm has a lot of promise.

7.2 Recommendations for Further Work

During the work of this spring, several new ideas have come to mind, both improvements to the existing work and new concepts. Sometimes a lot of work is done just to rule out a concept or some method, this is something which also has been experienced in this work. Based on the experiences done, some recommendations for future work on guidance and collision avoidance will be presented. Most of the suggestions will be connected to the work in progress through AUR-Lab

7.2.1 Future Work on ROV Minerva

A lot of hydrodynamic experiments have already been conducted for ROV Minerva. Still, there are a lot of parameters which still is unknown. As the control system increases in complexity, the need for more accurate modelling of ROV Minerva increases. On the topic of energy efficiency, it is important to have good estimates of the propeller coefficients and losses connected to the propellers. For doing more advanced guidance and collision avoidance systems, it should be looked into acquiring new sensors, and especially sonars. This would make the ROV be able to follow walls, and also avoid colliding with walls.

7.2.2 Future Work on Guidance

With respect to guidance, there is a lot of topics that have emerged during the work done in this thesis. Topics concerning mission planning, and especially within feasible mission planning is an area which is immense, since a lot of considerations should ideally be made. Work on this is as far as I know planned also for the future. With respect to AUVs, energy and autonomy is critical issues which increases the complexity of guidance systems. This means that even more robust and redundant guidance systems should be investigated. This will also overlap with collision avoidance.

If the work done in this thesis should be continued, it should be focus on overlap with control theory. Controller designs made specifically for path following would probably increase the performance of this feature.

7.2.3 Future Work on Collision Avoidance

Further investigation and testing of both the VFH algorithm and other algorithms should be conducted. The limitations should be evaluated and possible solutions should be tested. The VFH algorithm might be extended to be more robust, both with respect to the ROV Minervas physics, and also with respect to the algorithms drawbacks.

It is possible that some kind of hybrid collision avoidance scheme could be useful for an underwater vehicle with little knowledge about its environment. In order to do this, additional knowledge about global collision avoidance algorithms should be acquired.

The collision avoidance scenarios demonstrated in this thesis have been based on virtual obstacles. For future work on collision avoidance on ROV Minerva, it should be mounted with a forward looking sonar. This would require further work on mapping the sonar data into a map readable by a collision avoidance algorithm. In order to have a working collision avoidance system for ROV Minerva, this is crucial.

References

Antonelli, G. (2006). *Underwater Robots*. Springer-Verlag, 2 ed.

Antonelli, G., Chiaverini, S., Finotello, R., & Schiavon, R. (2001). Real-time path planning and obstacle avoidance for rais: an autonomous underwater vehicle. *Oceanic Engineering, IEEE Journal of*, *26*(2), 216 –227.

Borenstein, J., & Koren, Y. (1989). Real-time obstacle avoidance for fact mobile robots. *Systems, Man and Cybernetics, IEEE Transactions on*, *19*(5), 1179 –1187.

Borenstein, J., & Koren, Y. (1991). The vector field histogram - fast obstacle avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, *7*, 278–288.

Breivik, M. (2010). *Topics in Guided Motion Control of Marine Vehicles*. Ph.D. thesis, Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, Department of Engineering Cybernetics.

Breivik, M., & Fossen, T. (2004). Path following of straight lines and circles for marine surface vessels. In *6th IFAC Control Applications in Marine Systems (CAMS)*, (p. 65–70).

Breivik, M., & Fossen, T. I. (2008). Guidance Laws for Autonomous Underwater Vehicles. In A. V. Inzartsev (Ed.) *Underwater Vehicles*, chap. 4. I-Tech.

Candeloro, M. (2011). *Design of observers for dp and tracking of rov minerva with experimental results*. Master's thesis, Università Politecnica delle Marche/Norwegian University of Science and Technology.

Chardard, Y., & Copros, T. (2002). Swimmer: final sea demonstration of this innovative hybrid auv/rov system. In *Underwater Technology, 2002. Proceedings of the 2002 International Symposium on*, (pp. 17 – 23).

ConmpactRIO (2011). CompactRIO.
URL `http://www.ni.com/compactrio`

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). *Introduction to Algorithms, Second Edition*. McGraw-Hill Science/Engineering/Math.
URL `http://www.worldcat.org/isbn/0070131511`

Dukan, F., Ludvigsen, M., & Sørensen, A. J. (2011). Dynamic positioning system for a small size rov with experimental results.

DVL (2011). Teledyne RDI Worhorse Doppler velocity log.
URL `http://www.rdinstruments.com/navigator.aspx`

Engelhardtsen, Ø. (2007). *3D AUV Collision Avoidance*. Master's thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics.

Fernandes, D. d. A., Kørte, S. Ø., Dukan, F., Ludvigsen, M., & Sørensen, A. J. (2011). Exploring limitations and constraints of guidance systems for uuvs with experimental results. In *Submitted: OCEANS '11 MTS/IEEE KONA, Hawaii*.

Fossen, T. I. (2010). *Guidance and Control of Marine Craft*. Norwegian University of Science and Technology.

Fox, D., Burgard, W., & Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics and Automation*, *4*(1).

Jakuba, M., Yoerger, D., & Whitcomb, L. (2007). Longitudinal control design and performance evaluation for the nereus 11,000 m underwater vehicle. In *OCEANS 2007*, (pp. 1 –10).

Khatib, O. (1985). The potential field approach and operational space formulation in robot control. In *Proc. of the 4th Yale Workshop on Applications of Adaptive Systems Theory*, (pp. 208–214). Yale University, New Haven, CT, USA.

Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, *5*(1), 90–98.

Kirkeby, M. (2010a). *Comparison of controllers for dynamic positioning and tracking of rov minerva*. Master's thesis, Norwegian University of Science and Technology, Department of Marine Technology.

Kirkeby, M. (2010b). Dynamic Positioning of ROV Minerva. Tech. rep., Norwegian University of Science and Technology, Department of Marine Technology.

Koren, Y., & Borenstein, J. (1991). Potential field methods and their inherent limitations for mobile robot navigation. In *IN PROC. IEEE INT. CONF. ROBOTICS AND AUTOMATION*, (pp. 1398–1404).

Kørte, S. Ø. (2010). Guidance and Control of a Remotely Operated Vehicle (ROV). Tech. rep., Norwegian University of Science and Technology, Department of Marine Technology.

Labview (2011). Labview.
URL `http://www.ni.com/labview/`

Larson, J., Bruch, M., & Ebken, J. (2006). Autonomous navigation and obstacle avoidance for unmanned surface vehicles. In *SPIE Proc. 6230: Unmanned Systems Technology VIII*, (pp. 17–20).

LaValle, S. M. (1998). Rapidly-exploring random trees: A new tool for path planning. TR 98-11, Computer Science Dept., Iowa State University.

Loe, Ø. (2007). Collision avoidance concepts for marine surface craft. Tech. rep., Norwegian University of Science and Technology, Department of Engineering Cybernetics.

Loe, Ø. (2008). *Collision Avoidance for Unmanned Surface Vehicles*. Master's thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics.

Ludvigsen, M. (2010). *An ROV Toolbox for Optical and Acoustical Seabed Investigations*. Ph.D. thesis, Norwegian University of Science and Technology, Faculty of Engineering Science and Technology, Department of Marine Technology.

Ludvigsen, M., & Ødegård, Ø. J. (2004). Fullskala Thrust-test av Minerva. Tech. rep., Norwegian University of Science and Technology, Department of Marine Technology.

Ludvigsen, M., & Ødegård, Ø. J. (2006). Sammenfatning av hydrodynamiske forsøk med rov minerva. Tech. rep., Norwegian University of Science and Technology, Department of Marine Technology.

MRU6 (2011). Kongsberg Seatex MRU6.
URL `http://www.km.kongsberg.com`

Pettersen, K., & Fossen, T. (2000). Underactuated dynamic positioning of a ship-experimental results. *Control Systems Technology, IEEE Transactions on*, *8*(5), 856 –863.

Refsnes, J. E. G. (2008). *Nonlinear Model-Based Control of Slender Body AUVs*. Ph.D. thesis, Norwegian University of Science and Technology, Faculty of Engineering Science and Technology, Department of Marine Technology.

Roberts, G. (2008). Trends in marine control systems. *Annual Reviews in Control*, *32*(2), 263 – 269.

Silicon-Sensing-Systems (2011). Silicon rate sensor by Silicon Sensing Systems.
URL `http://www.siliconsensing.com`

Skjetne, R. (2005). *The Maneuvering Problem*. Ph.D. thesis, Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, Department of Engineering Cybernetics.

Skjetne, R., Fossen, T. I., & Kokotovic, P. V. (2004). Robust output maneuvering for a class of nonlinear systems. *Automatica*, *40*(3), 373 – 383.

Sonar-MS1000 (2011). Kongsberg single beam scanning sonar - MS 1000.
URL `http://www.km.kongsberg.com`

Standardi, L. (2011). *Signal Processing and Change Detection Applied to ROV in DP System*. Master's thesis, Università Politecnica delle Marche/Norwegian University of Science and Technology.

Svendby, E. (2007). *Robust Control of ROV/AUVs*. Master's thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics.

Sørensen, A. J. (2011). *Marine Control Systems: Propulsion and Motion Control of Ships and Ocean Structures*. Norwegian University of Science and Technology, Department of Marine Technology.

Sørensen, A. J. (2011). A survey of dynamic positioning control systems. *Annual Reviews in Control*, *35*(1), 123 – 136.
URL `http://www.sciencedirect.com/science/article/pii/S1367578811000095`

Sørensen, A. J., & Refsnes, J. (2009). Towards supervisory-switched control of hybrid underwater vehicles. *International Journal of the Society for Underrwater Technology*, *28*(4), 141–150.

Truong, P. T. (2011). *Navigation of ROVs and basic signal processing*. Master's thesis, Norwegian University of Science and Technology, Department of Marine Technology.

Wikipedia (2011). Dynamic positioning, visited june 7th, 2011.
URL `http://en.wikipedia.org/wiki/Dynamic_positioning`

Yoerger, D., Newman, J., & Slotine, J. J. (1986). Supervisory control system for the jason rov. *IEEE Journal of Oceanic Engineering*, *11*(3), 392–400.

Appendix A

Complete Results from a Selection of Full Scale Experiments

For studying the full scale results closer, some of the full scale experiments will be presented with all the figures available.

A.1 Arc Radius of 2 Metres

With Constant Forward Speed

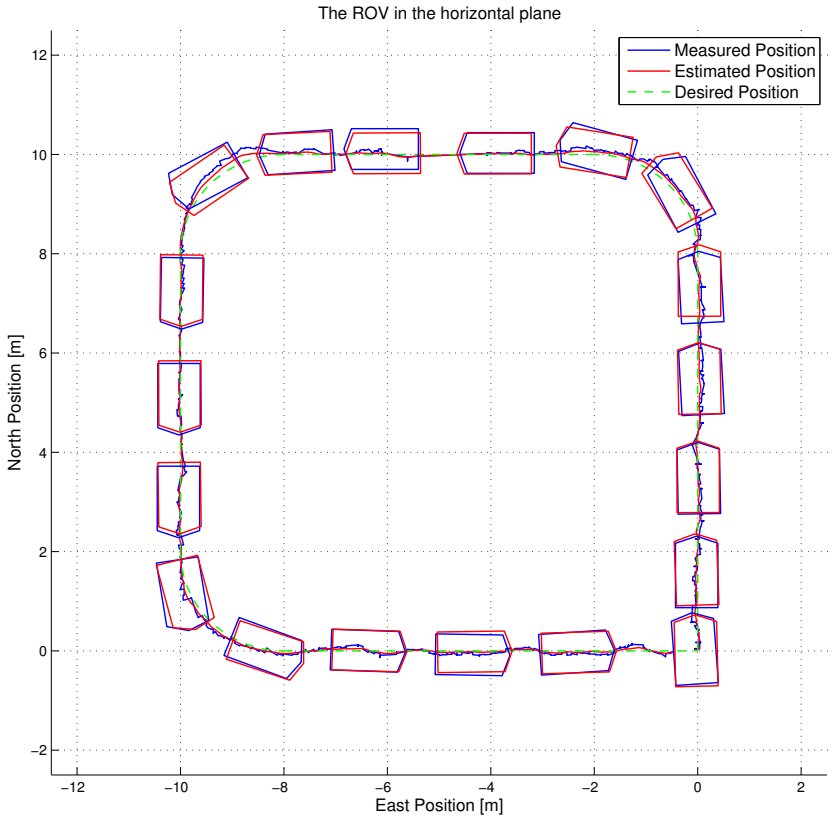


Figure A.1: The ROV Minerva is following a square with arc radius of 2 metres in the corners.

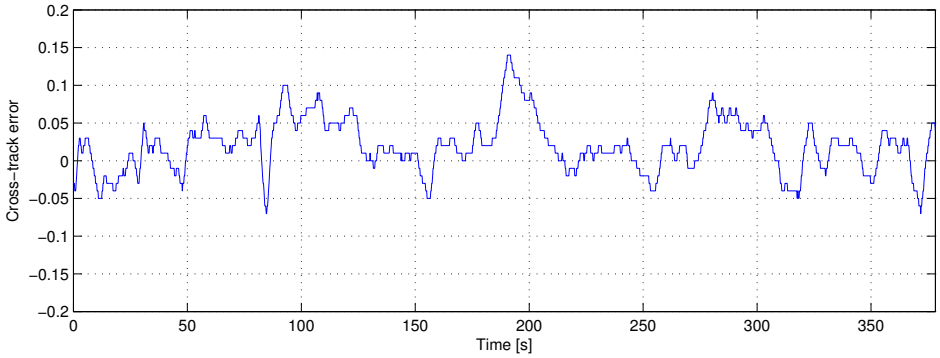


Figure A.2: The cross-track error stays tightly within 0.14 metres at all times.

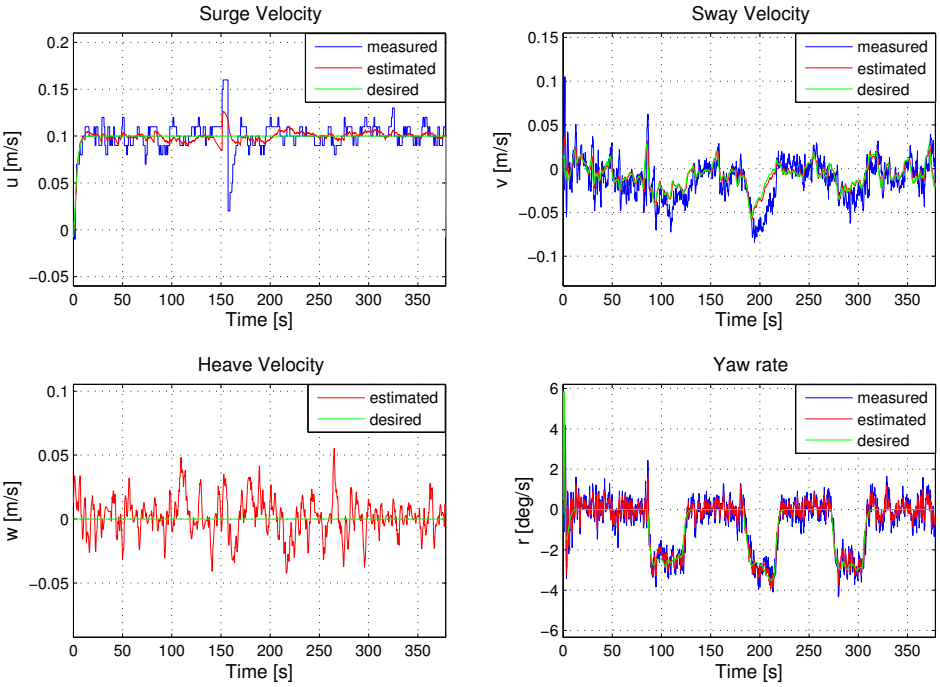


Figure A.3: The surge velocity is constant at 0.1 m/s. The sway velocity is proportional to the cross-track error, and very low motion is needed to control the cross-track error. The yaw-rate is zero during the straight lines, while it is approximately constant during turning.

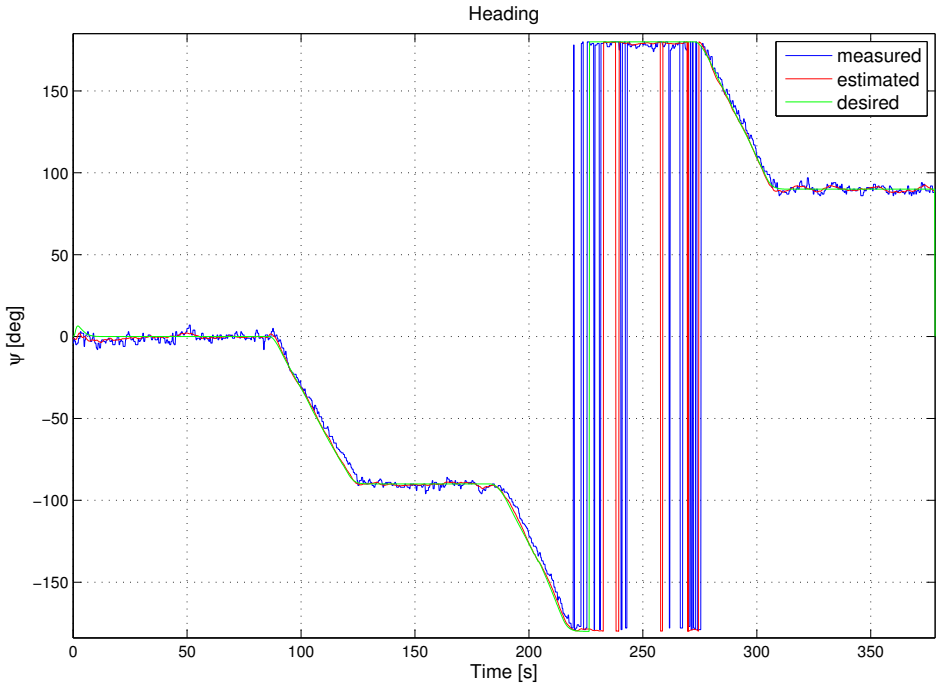


Figure A.4: The heading indicates that the ROV follows the direction of the path nicely.

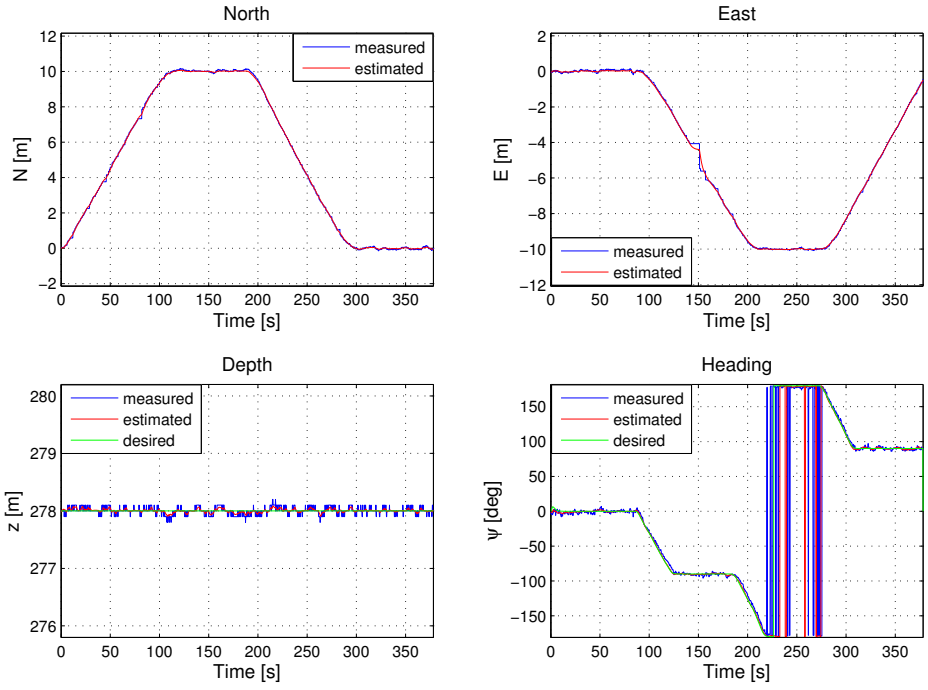


Figure A.5: A small HiPaP drop-out at approximately 150 seconds can be seen in the East plot. The depth is kept tightly to the desired value with a maximum deviation of 0.2 metres.

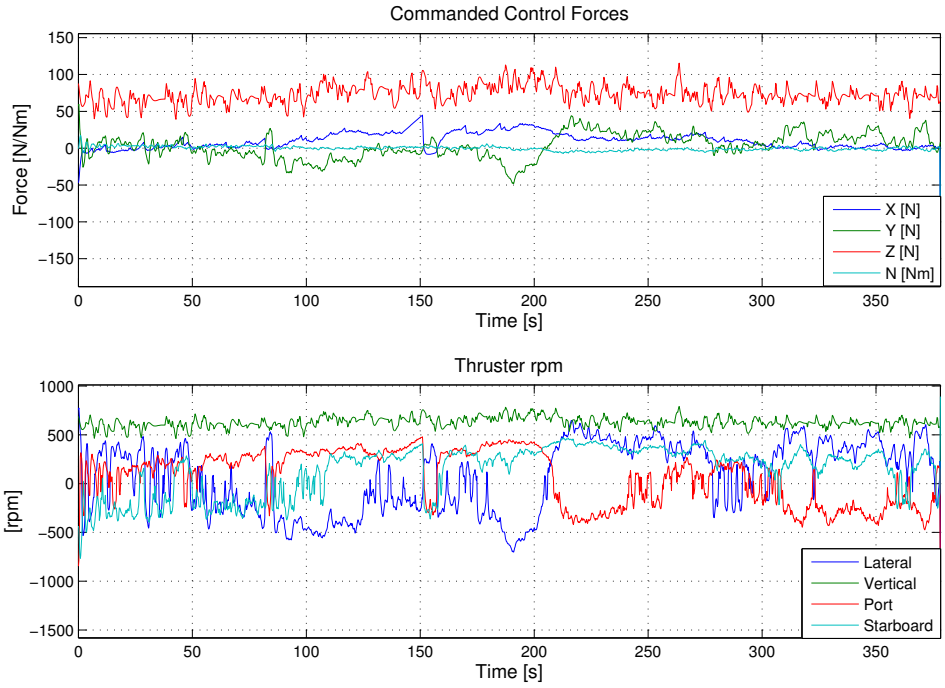


Figure A.6: The thruster inputs stays well within the saturation limits of $-1450, 1450$ RPM.

With Speed Assignment

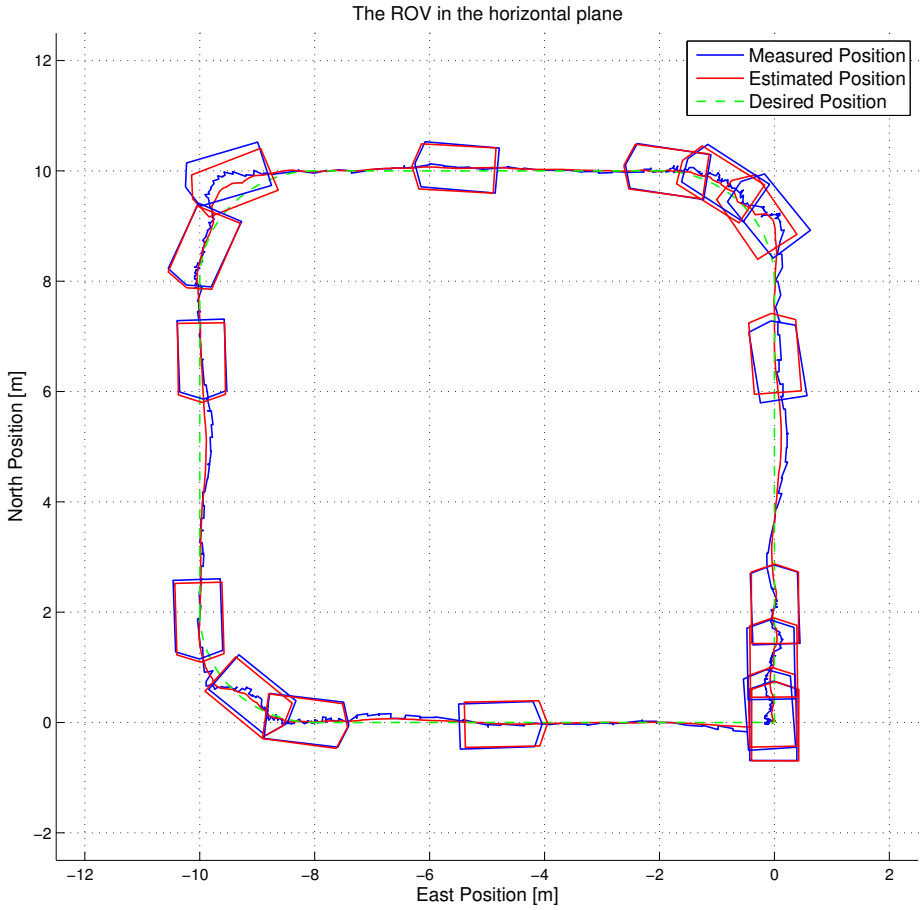


Figure A.7: The ROV is following a square with arc radius of 2 metre arc radius. The reduced speed can seen by the ROV outline being plotted more frequently during turns.

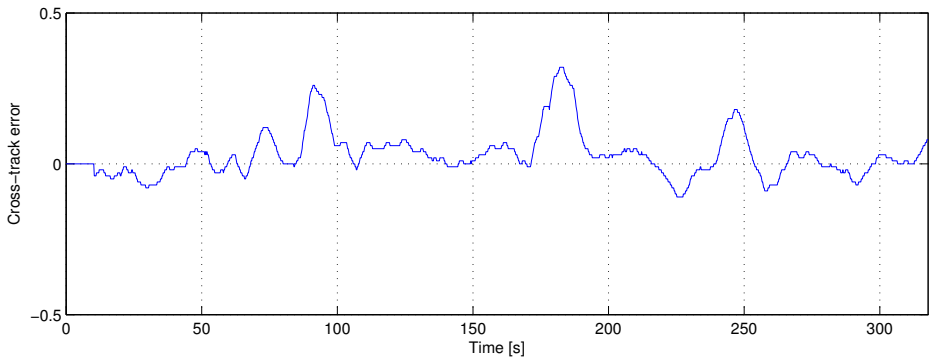


Figure A.8: The cross-track have increases for a short time when entering the arcs, but as the speed is reduced the cross-track error also is reduced.

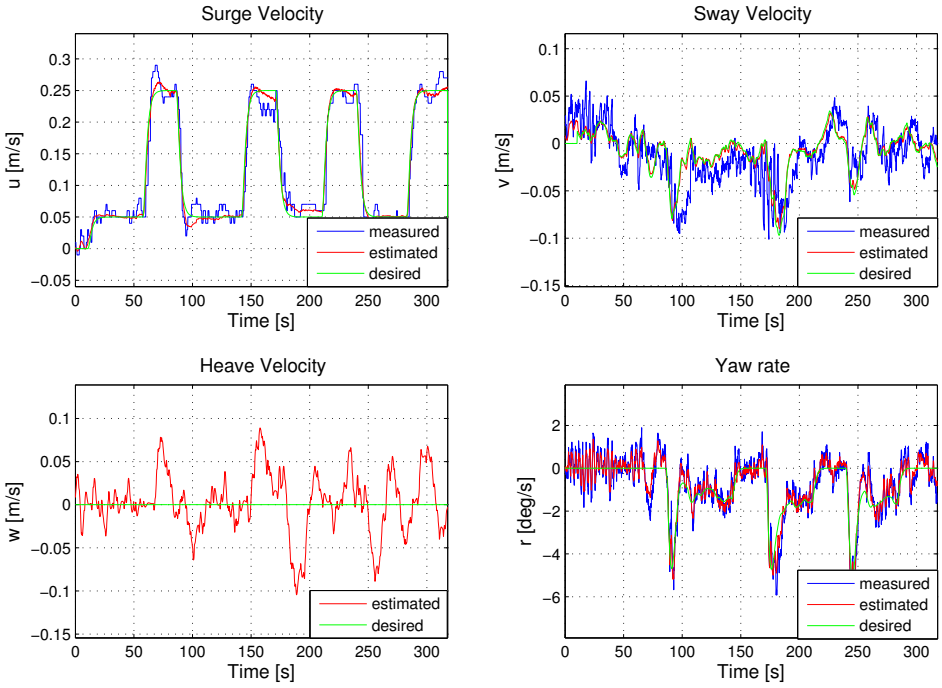


Figure A.9: The surge velocity can be seen to increase and decrease between 0.25 m/s and 0.05 m/s. The sway velocity and yaw-rate is oscillating somewhat, probably due to the acceleration and deceleration in surge.

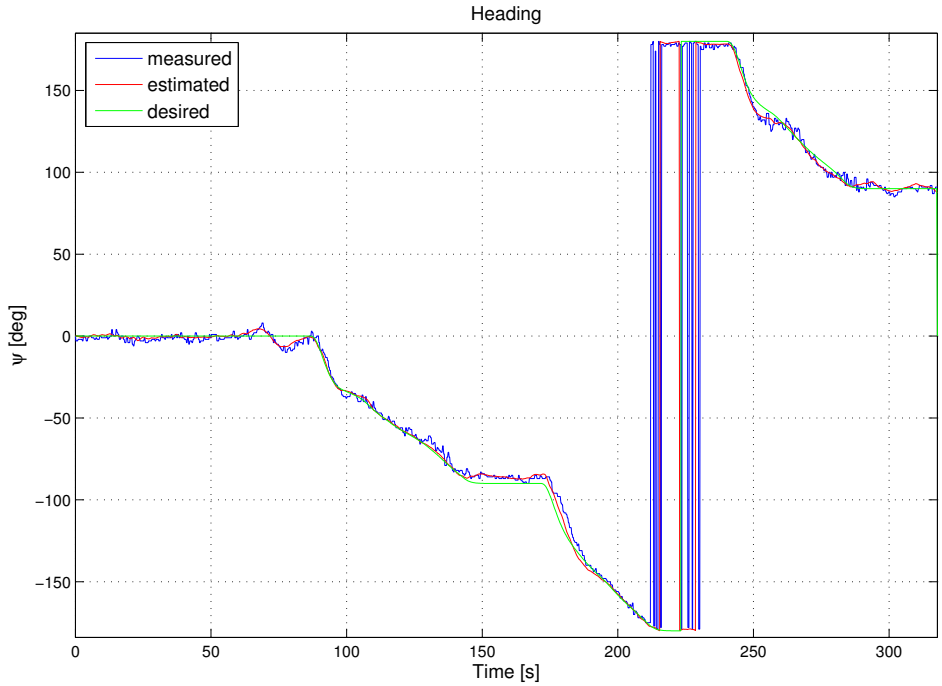


Figure A.10: The heading is also oscillating somewhat, but generally stays within 5 degrees of the desired signal.

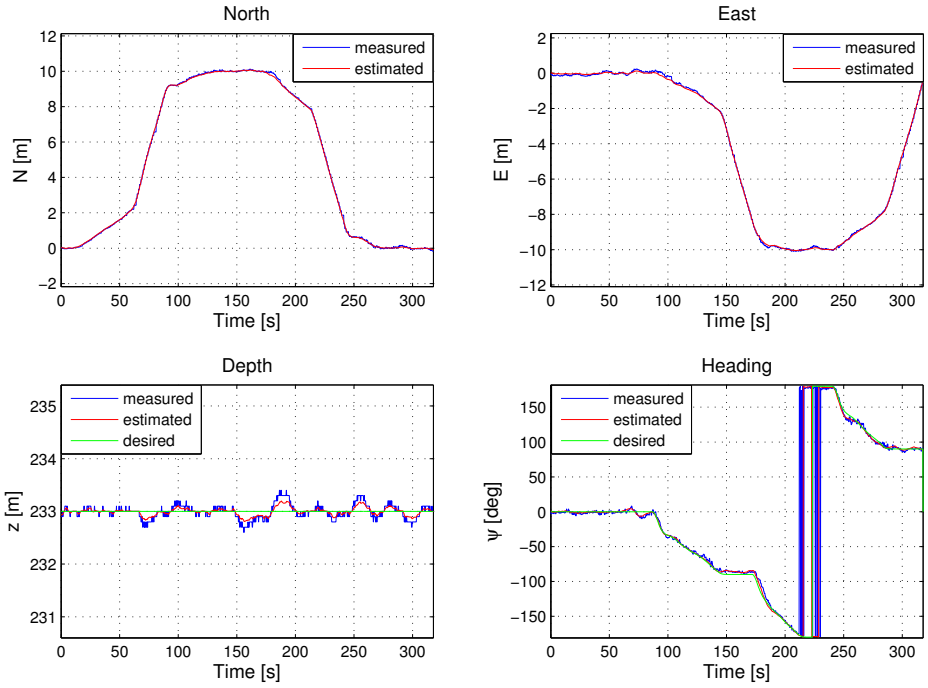


Figure A.11: In North and East, the increase and decrease in surge velocity can be seen by the changing slope. The depth is regulated at 233 metres.

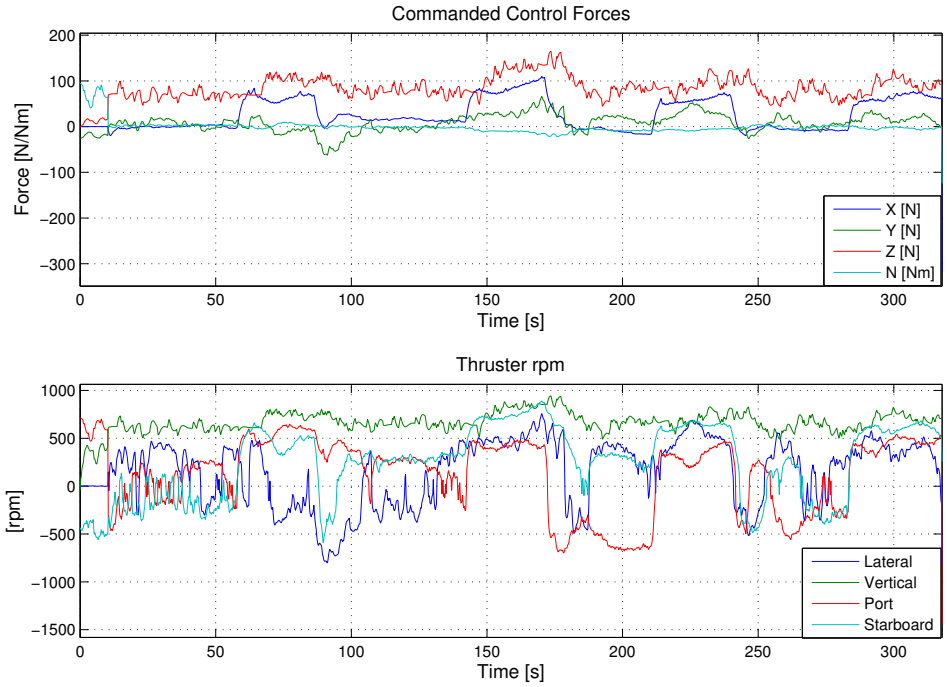


Figure A.12: The horizontal thrusters are doing all the work, while the vertical thrusters are only used to keep the ROV at the desired depth. The values are well within the limits.

A.2 Lawnmower Pattern with 10 Metre Line Distance

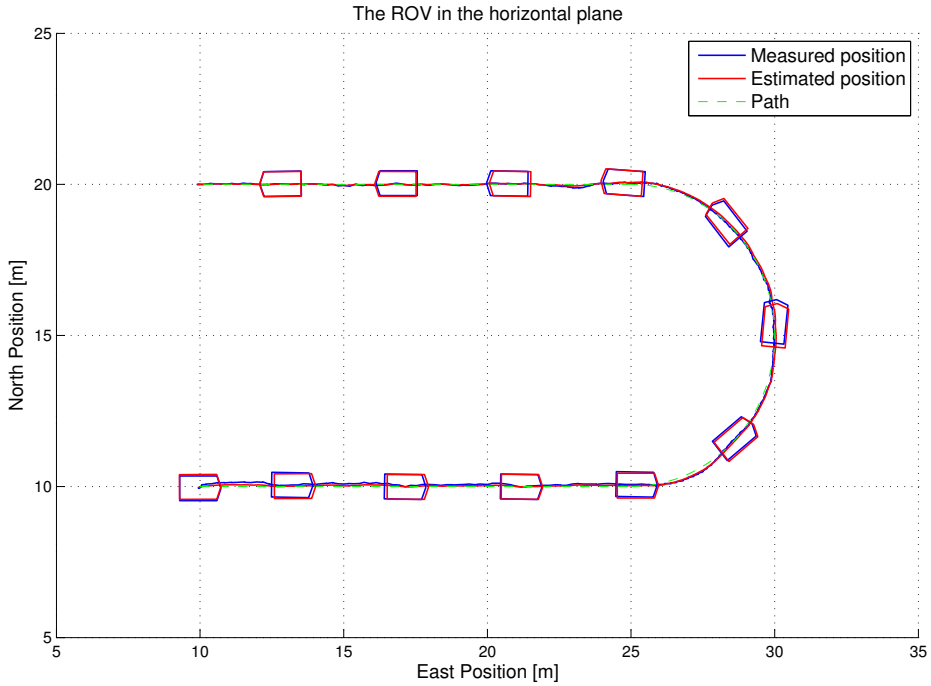


Figure A.13: As expected when the forward speed is set to 0.2 m/s, the ROV is able to follow both the lines, which is the main objective, and the circle.

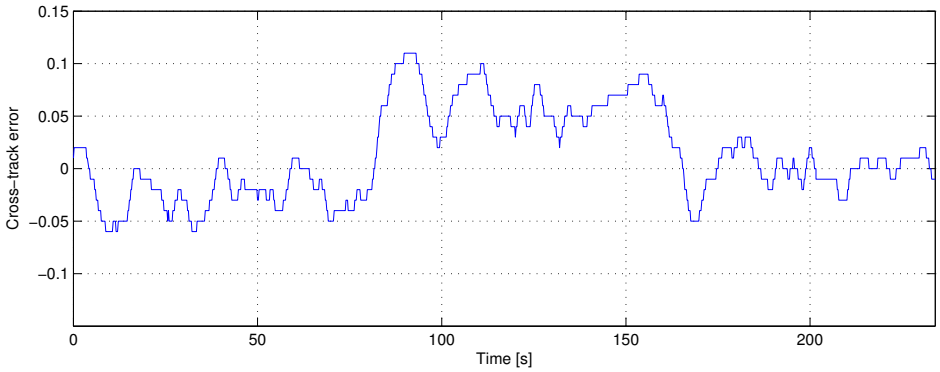


Figure A.14: The cross-track error stays nicely within 0.11 metres.

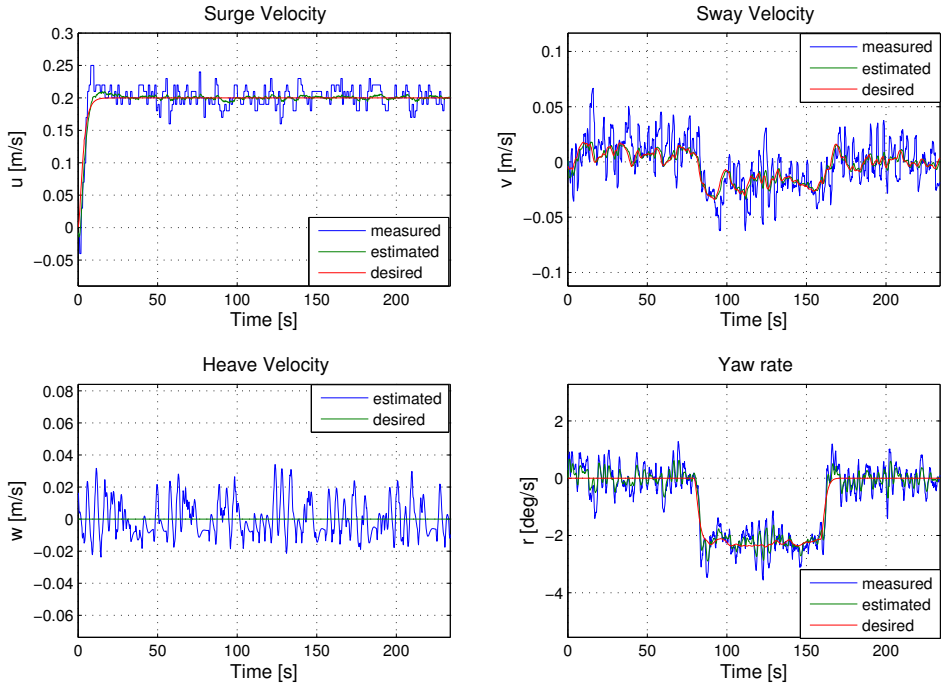


Figure A.15: Due to MRU measurements, the velocities are now very close to the desired. The yaw-rate can be seen to be very close to a constant value around -2 degrees per second during the turn.

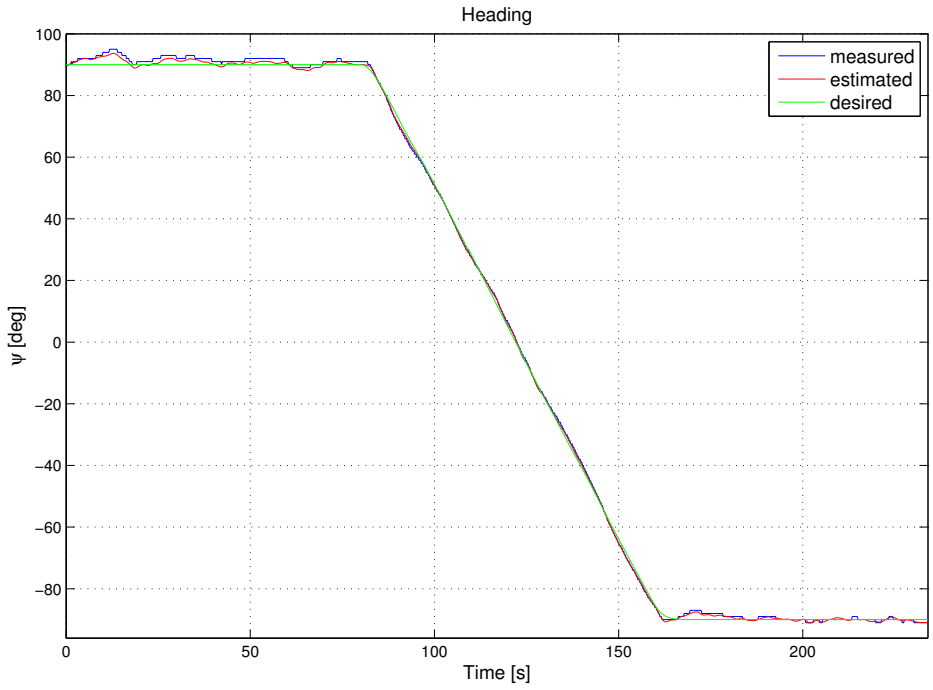


Figure A.16: The heading tracks the desired heading very closely.

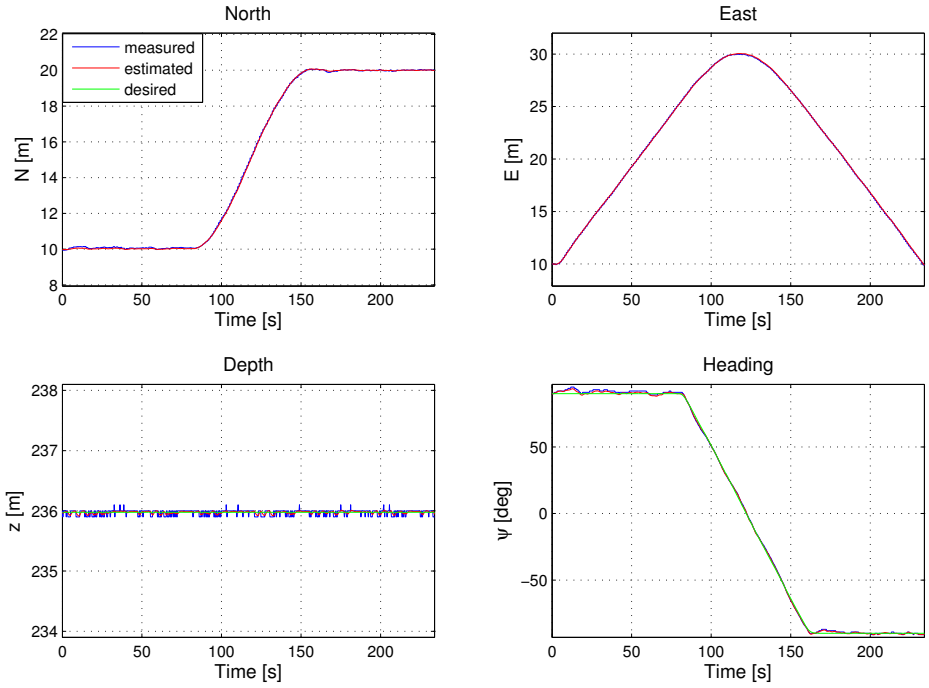


Figure A.17: The position is generally very smooth, and the depth is kept tightly at 236 metres.

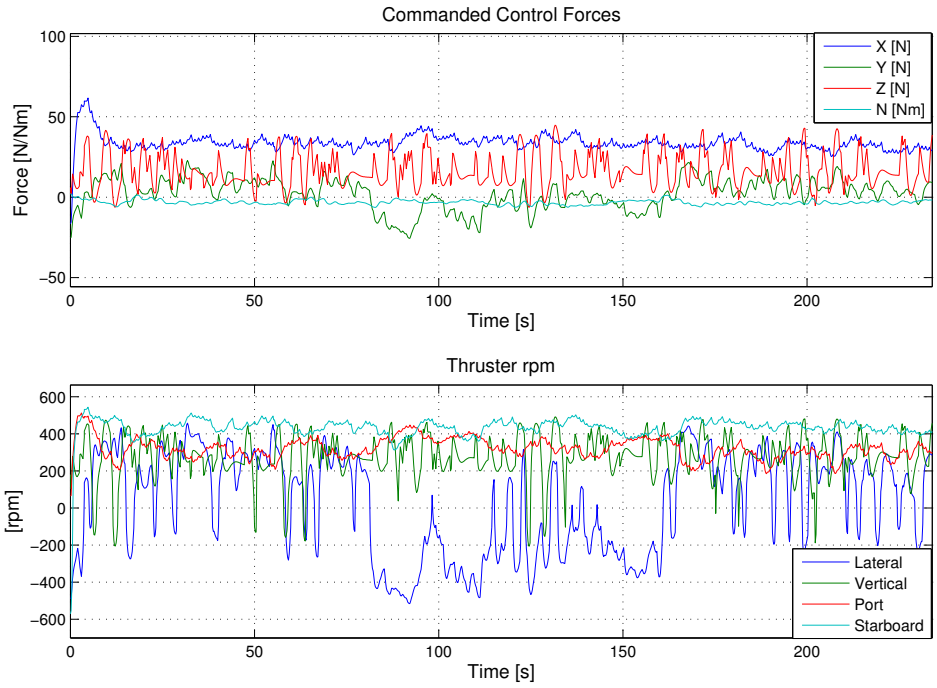


Figure A.18: The thrust inputs are generally quite low and it can be seen that the lateral thruster get increased thruster inputs during the turn.

A.3 Collision Avoidance From Scenario 1

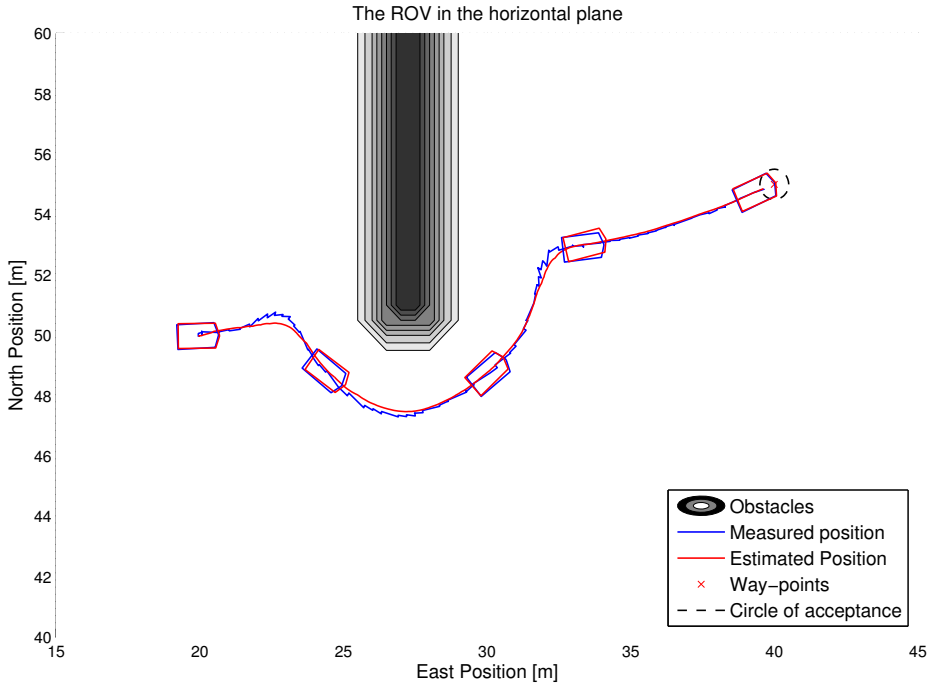


Figure A.19: Full scale experiment where ROV Minerva is travelling around an obstacle.

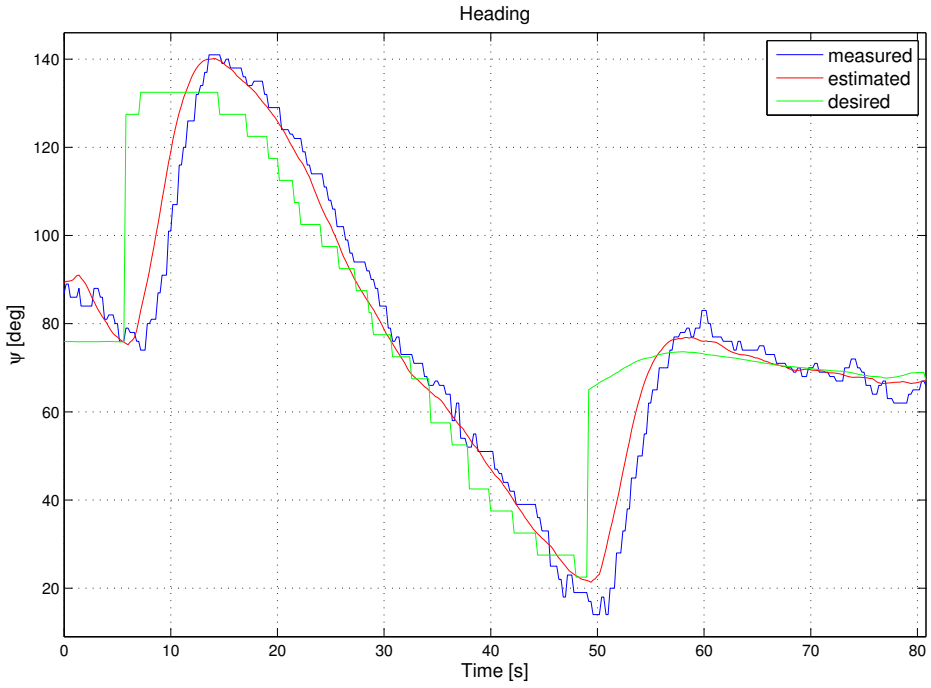


Figure A.20: The ROV has problems tracking the desired heading due to major steps, but follows quite nicely for smaller steps.

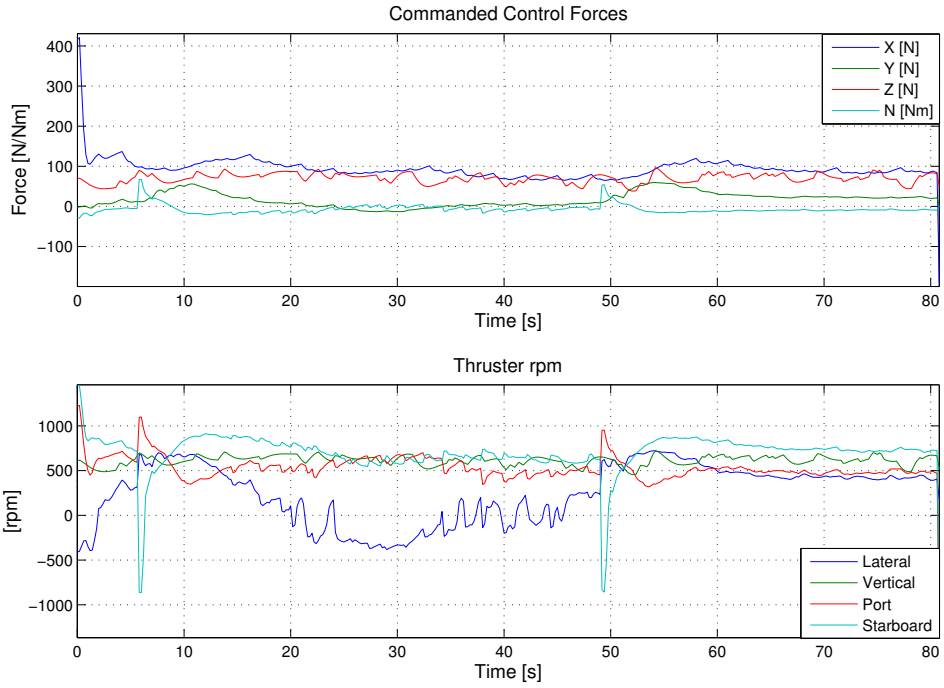


Figure A.21: The thrust is heavily influence by the step inputs in heading. This is especially seen around 6 and 49 seconds for the starboard thruster.

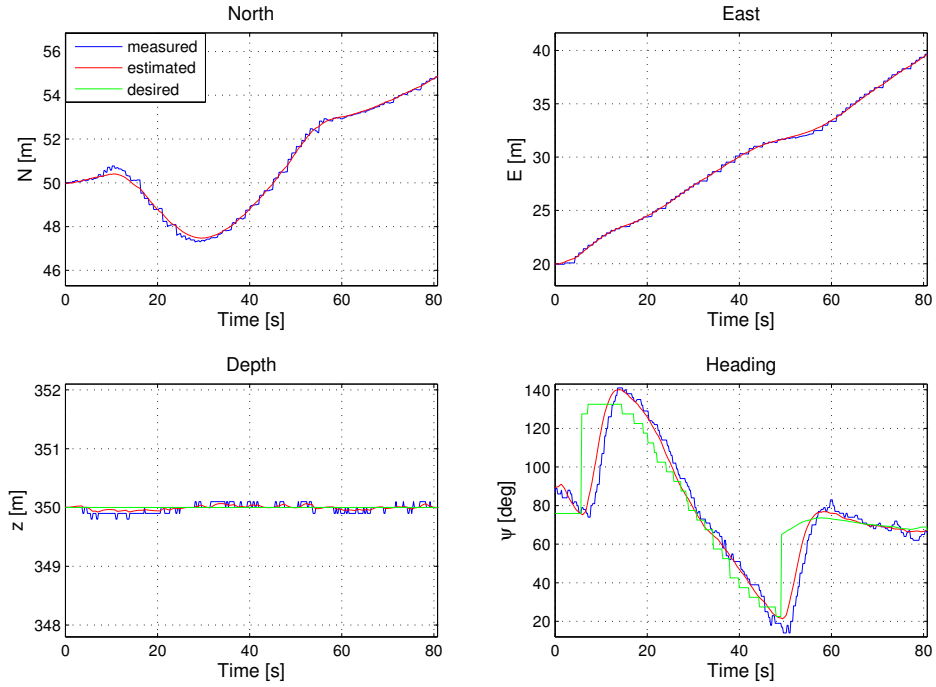


Figure A.22: The depth is kept nicely at 350 metres.

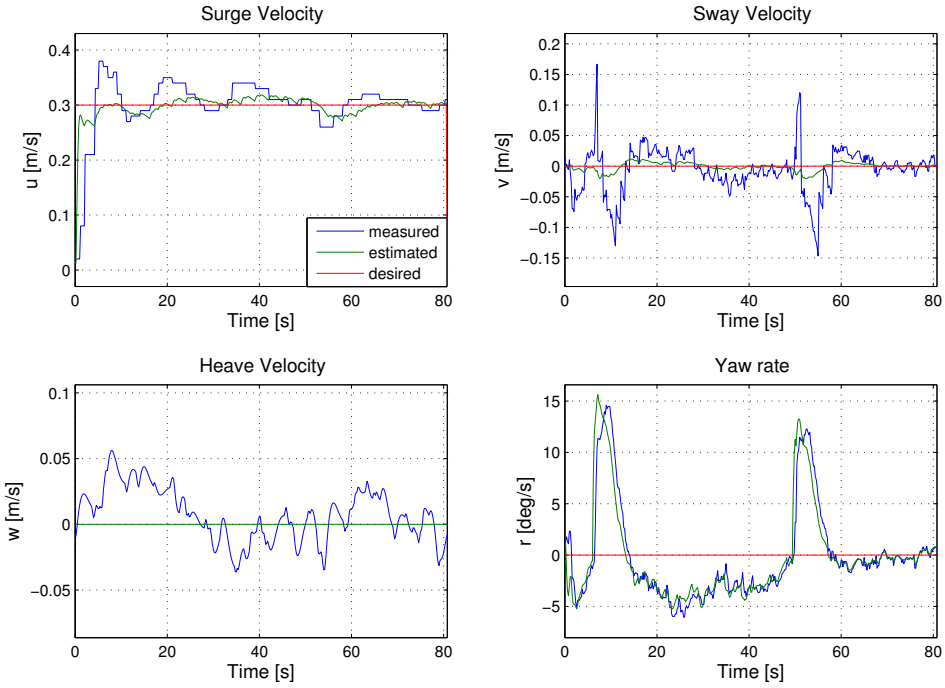


Figure A.23: The forward speed is constant at 0.3 m/s, which means that the steering assignment in the VFH algorithm has not been activated during this experiment. Due to lack of smooth velocity signals, the surge velocity oscillates somewhat when getting a step input. The yaw-rate is heavily influenced by the heading step inputs.

Appendix B

Program Listing

The CD that is attached to the thesis includes the MATLAB scripts for running the simulations presented in this thesis. It requires MATLAB with C compiler and the MSS toolboxes which are available on the CD. In addition a pdf copy of this thesis is included.

The simulation are dependent on that all the toolboxes are properly loaded, so MATLAB should be started by opening MATLAB itself, not a MATLAB file. Since some of the simulink models are different, I would recommend to close MATLAB and restart it between simulating the different scenarios. The simulations with respect to guidance strategy needs to run until an error occurs due to get correct timing to completion of path. Within these scenarios plotting scripts should be ran manually after running the simulation. The "force_calc.m" script will calculate the forces and effect and store the data in the text file "calc.txt". To run the simulations, please run the files called "Run_..".

Guidance simulations folders:

- Guidance strategy ① - Stopatwaypoint.
- Guidance strategy ② constant forward speed - circ_constant_speed.
- Guidance strategy ② reduced forward speed on arcs - circ_reduced_speed.
- Guidance strategy ③ - Normal_zeroradius.
- Guidance strategy ④ constant forward speed - circout_constantspeed.
- Guidance strategy ④ reduced forward speed on arcs - circout_constantspeed.

The simulations done on collision avoidance will be numbered according to appearance in the thesis. So the folder "Scenario 1" corresponds to the first simulation presented in the thesis and so fort.