Autonomous Control of a Differential Thrust Micro ROV

by

Wei Wang

A thesis
presented to the University of Waterloo
in partial fulfillment of the requirements for the degree of
Master of Applied Science
in
Mechanical Engineering

Waterloo, Ontario, Canada, October, 2006

I hereby declare that I am the sole author of this thesis. This is the true copy of the thesis, including any required final revisions, as accepted by examiners.

I understand that my thesis may be made electronically available to the public.

Wei Wang

Abstract

Underwater vehicles that use differential thrust for surge and yaw motion control have the advantage of increased maneuverability. Unfortunately, such vehicles usually don't have thrusters/actuators to control the lateral movements. Hence, they fall into the underactuated vehicle category.

The goal of the work in this thesis is to develop an autonomous control system for a differential thrust underwater remotely operated vehicle (ROV) to track predefined position trajectories. This is challenging because the mathematical model for underwater vehicles is highly nonlinear and the environmental disturbances are usually strong and unpredictable. These factors make the design of the control system very difficult.

In this work, we use the VideoRay Pro III micro ROV as the test platform, on which we design an autonomous control system. We first present the development and analysis of a hydrodynamic model of the VideoRay Pro III using both analytical and experimental approaches. Based on this model, a state estimator is then designed using the unscented Kalman filter, which yields better estimates of the system states and their uncertainty level in a highly nonlinear system than the commonly used extended Kalman filter. In the controller design, the integrator backstepping technique is used to achieve a Lyapunov stable trajectory tracking controller based on the work by A. P. Aguiar *et al.* [2]. We extended their work by further considering the quadratic drag terms in the vehicle's hydrodynamic model. The sliding mode control is used to design the bearing and depth controller.

Finally, the autonomous control system is validated by simulation and experimental tests. It is shown that the VideoRay Pro III is able to track the predefined trajectory within error range of 0.5 meters.

Acknowledgements

I would first like to thank my supervisor, Professor Christopher M. Clark, for giving me a chance, for helping me to get started on such an interesting problem, and for providing me support and advice during the course of this project.

I would also like to thank Professor Jan Paul Huissoon and Professor John J. McPhee for agreeing to read this thesis. I would like to thank Mr. Terry Ridgeway for helping me with the fluid experiments, based on which the coefficients of the dynamical model of the VideoRay Pro III are determined. I would like to thank all the colleagues in LAIR who have given me a great deal of advice and suggestion.

Finally, I would like to thank my family for their unchanging confidence and support.

Without all these assistance and support, this work would not even have been accomplished.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

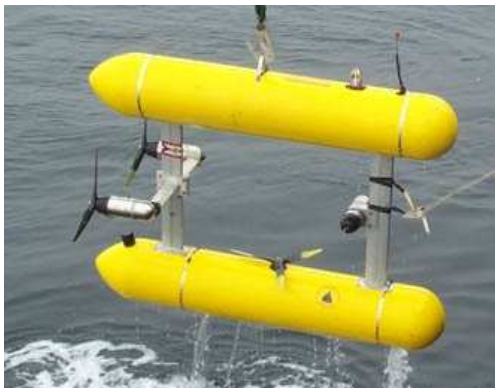
Remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs) have been applied in a wide variety of areas. Recently, there has been a trend to use smaller autonomous underwater vehicles, both tethered and untethered, in rivers, lakes and oceans. The potential uses for AUVs include: scientific (oceanography, geology, geophysics, ...), environmental (waste disposal monitoring, wetland surveillance, ...), commercial (oil and gas, submerged cables, harbours, ...), military (minehunting, tactical information gathering, smart weapons, ...) and other applications where their endurance, economy and saftey can replace divers.

During a mission, an AUV is expected to carry sensors, such as scanning sonar, bathymetry, bottom profiler, etc., track a certain planned trajectory, and even make on-line decisions allowing for mission reconfiguration.

Currently, there are many AUVs being deployed for research purposes and even for carrying out missions. For example, the SeaBED AUV is built by the Woods Hole Oceanographic Institute (WHOI) for geologic and benthic-habitat studies of the sea floor. The SeaBED AUV is constructed of two horizontal cylinders configured one above the other, connected by struts. It is designed to precisely navigate survey tracks at altitudes as low as 2.5 m above the sea floor. Another example is the Phoenix AUV and ARIES AUV developed by Naval Postgraduate School, for studies relating to the design of control system, navigational accuracy, mission planning, etc. These two AUVs are shown in Figure 1.1.

Ultimately, AUVs like SeaBED or ARIES may be left entirely unattended awaiting

ocean events or conducting repeated surveys.



(a) SeaBED AUV (courtesy of WHOI)

(b) ARIES AUV (courtesy of NPS)

Figure 1.1: (a) SeaBED (b) ARIES

AUVs like SeaBED usually use differential thrust to control their motion, especially in the horizontal plane. Compared to using one propulsion thrust and several hydro surfaces such as fins and rudder, differential thrust from propellers provide more advantanges in terms of maneuveribility[1]. For instance, AUVs with differential thrust are able to turn with smaller radius, and even turn on spot without having to have a certain forward moving speed.

In this work, an autonomous control system of a differential thrust AUV will be disscussed. The control system will be verified and validated by computer simulations, and field tests using a VideoRay Pro III ROV.

1.1 Autonomous Control for VideoRay Pro III ROV

Among many requirements that make an AUV autonomous, is the ability to track a predifined trajectory, which might be parameterized with time. There has been a great deal of research on trajectory tracking problem for land vehicles, using various positioning systems, such as GPS, for navigation. Although much success has been achieved for the land

[1] *maneuverability* can be defined as the capability of the vehicle to carry out specific maneuvers [9]

vehicles trajectory tracking, it is still a problem for underwater vehicles. In the underwater world, vehicle positioning is much more difficult because several issues must be addressed including positioning accuracy, bandwidth, and possible time delay for underwater communication.

In this work, we develop a control system for an AUV to track a predefined trajectory using an underwater acoustic positioning system. With the ability to track trajectories, the AUV will be able to carry out missions on its own without human intervention. One of the possible missions, for example, would be deep sea survey consisting of a descent to a predetermined position on the sea floor followed by the execution of a series of lawn mower-like tracklines above the sea floor, and finally an ascent back to the surface ship for recovery.

The vehicle used in this research is a VideoRay Pro III micro ROV. It is a system designed for intensive, underwater operations. It has an open architecture that accommodates a wide variety of tools and sensors. The VideoRay Pro III system consists of a control console, a submersible robot and a tether deploying mechanism. The control console has a video display, joystick controls for horizontal and vertical movement, and a computer control interface. The submersible robot has two horizontal thrusters and one vertical thruster for its motion control, a pressure sensor for measuring depth, and a compass for measuring orientation. It also has an accessory connector allowing for field integration of various instruments and sensors, as shown in Figure 1.2.

Figure 1.2: VideoRay Pro 3 system

The Pilot tracking system from Desert Star systems was used for the underwater positioning system. The Pilot tracking system uses short base line (SBL) technology to track small ROVs. The system consists of a control console connected to a PC running DiveTerm software, and three cabled sonar transducers. The sonar transducers are lowered over the side of the surface vessel or dock. The best tracking performance of this acoustic positioning system is a nominal of ± 0.15 m RMS. The accuracy of the target position depends on the distance between the surface station transducers and the distance between the target and the transducers. The Pilot tracking system is shown in Figure 1.3.

Figure 1.3: Pilot acoustic positioning system.

To allow the robot to track a predefined trajectory, an autonomous control system was developed, the structure of which is shown in Figure 1.4. In this system block digram, we have an input trajectory which could be generated by some high level mission planning algorithm, a controller that outputs desired control parameters such as surge force and yaw torque to drive the vehicle to follow the trajectory, a sensor system consisting of an underwater positioning system and compass, and an observer/estimator providing full state estimation including position and velocity information for the controller. Note that for the observer/estimator to generate accurate state estimation, an accurate dynamic model of the VideoRay Pro III is required.

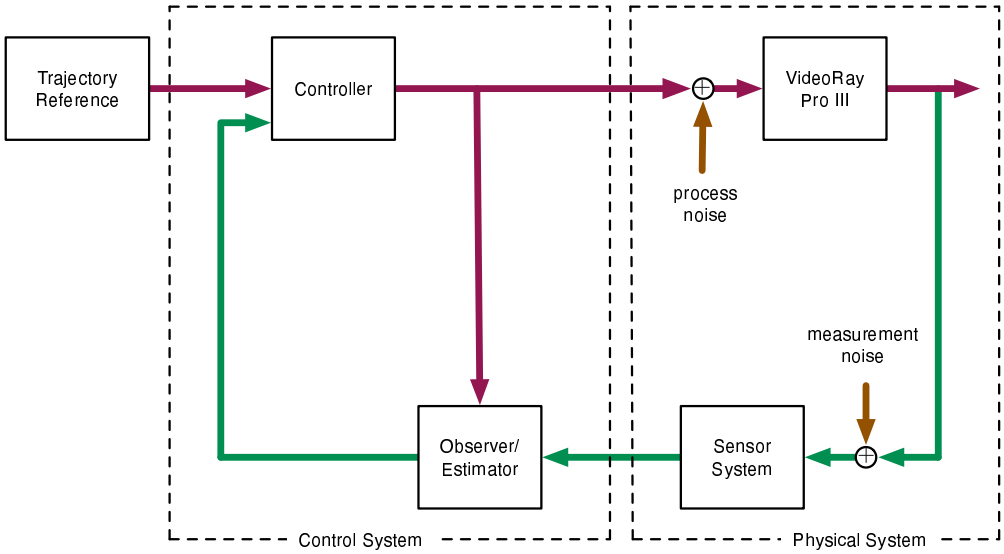


Figure 1.4: Autonomous control system configuration

1.2 Organization of this Work

Chapter 2 is devoted to the development of a six-degree of freedom vehicle dynamical model. The model developed is based on theoretical and existing empirical hydrodynamic work. It is expected that the parameters for the model will require tuning because the influence of the tether is not considered in the model. Our focus here is on the model structure.

In Chapter 3, the state observer (or estimator) is discussed. A state observer is crucial to the realization of this autonomous system. For the controller to work properly, full state feedback including the vehicle's position and velocity is required. However, only the position information could be obtained with the Pilot acoustic positioning system and the compass. The observer uses model-based predicted states, and fuses them with the acoustic positioning system observations and the compass heading observations. This results in a full state estimation. The well known observer/estimator for state estimation is the Kalman filter, which works quite well in linear systems. Since underwater vehicles exhibit high nonlinearity in their models, the Kalman filter, even the extended Kalman

filter, will not work well in this situation. Therefore, the unscented Kalman filter (UKF) will be used to solve this problem.

In Chapter 4, the analysis and development of a tracking controller for the horizontal plane trajectory tracking is discussed. The use of sliding mode control for depth and bearing control is also presented

In Chapter 5, simulation results of the vehicle following a circular trajectory are presented using the full non-linear six-degree of freedom model. The experimental results of the vehicle following a straight line trajectory are also presented and discussed for evaluating the tracking performance.

Finally, this work is concluded by considering the limitations of the trajectory tracking controller and suggesting directions for future work.

Chapter 2

Modeling of VideoRay Pro III

2.1 Introduction

An accurate dynamic model is required for autonomous control of an underwater vehicle. Accurate dynamic models are crucial to the realization of precision autopilots, AUV simulators and for prediction of performance [28, 31]. However, the modeling and control of underwater vehicles is difficult. The governing dynamics of underwater vehicles are fairly well understood, but they are difficult to handle for practical design and control purposes [23, 5]. The problem includes many nonlinearities and modeling uncertainties. These hydrodynamic and inertial nonlinearities are present due to coupling between the degrees of freedom [9]. For example, currents usually exist in the underwater environment which become coupled with the direction of motion. The presence of these non-linear dynamics requires the use of a numerical technique to determine the vehicle response to thruster inputs and external disturbances over the wide range of operating conditions.

In general, modeling techniques tend to fall into two categories [10]:

1. Predictive methods based on either Computational Fluid Dynamics or strip theory, and

2. Experimental techniques.

The predictive methods calculate the vehicle's dynamic motion parameters from the vehicle's design. It has the advantages of low cost, being easy to implement, and being

able to carry out even before the vehicle has been built, but it has the disadvantage of less accuracy.

In contrast, experimental techniques are usually carried out by using towing tank testing and more recently by system identification methods, and have the advatange of being more accurate. They are usually more costly as well.

Figure 2.1: VideoRay Pro III

These two techniques will be used to build a hydrodynamic model for the VideoRay Pro III ROV. The VideoRay Pro III is a small inspection-class micro ROV, with hundreds of units in operation around the world (see Figure 2.1). It is designed for underwater exploration at maximum depth of 500 feet (152 meters). The basic system includes the submersible itself, an integrated control box, a tether deployment system, and a tool kit. The vehicle is neutrally buoyant and hydrostatically stable in the water due to its weight

distribution. It is equipped with a system of sensors including front facing and rear facing cameras, depth gauge and compass. The vehicle has three control thrusters, two for horizontal movements and one for vertical movements.

The vehicle is considered as a 6 degrees of freedom (DOF) free body in space, namely *surge*, *sway*, *heave*, *pitch*, *roll* and *yaw* motions. The control of the vehicle is only available in the surge, heave, and yaw motion. Equal and differential thrust from the horizontal thrusters provide control in surge motion and yaw motion respectively. The heave motion is controlled by the vertical thruster. Therefore, the VideoRay Pro III falls into the category of underactuated vehicles because the dimension of the control vector is less than the degrees of freedom.

In this chapter, a dynamic model of the VideoRay Pro III micro ROV is presented, using both strip theory and/or experimental techniques. In determining the model parameters, a series of experiments were performed and a system identification method is used. Finally, a decoupled dynamical model of the vehicle is described to faciliate the design of trajectory tracking controllers.

2.2 Vehicle Dynamics

2.2.1 6-DOF Reference Frames

For marine vehicles, the 6 degrees of freedom are conventionally defined as *surge, sway, heave, roll, pitch* and *yaw*. Two reference frames are used to describe the vehicle states, one being the inertial frame (or earth-fixed frame), and the other being the local body-fixed frame with its origin coincident with the vehicle's center of mass and the three axes in the vehicle's *surge*, *sway* and *heave* directions. (see Figure 2.2)

2.2.2 Kinematics

The motion of the vehicle in 6 degree of freedom, *surge, sway, heave, roll, pitch* and *yaw*, are described by the following vectors [9]:

$\eta_1 = [x,\ y,\ z]^T$: position in inertial frame;

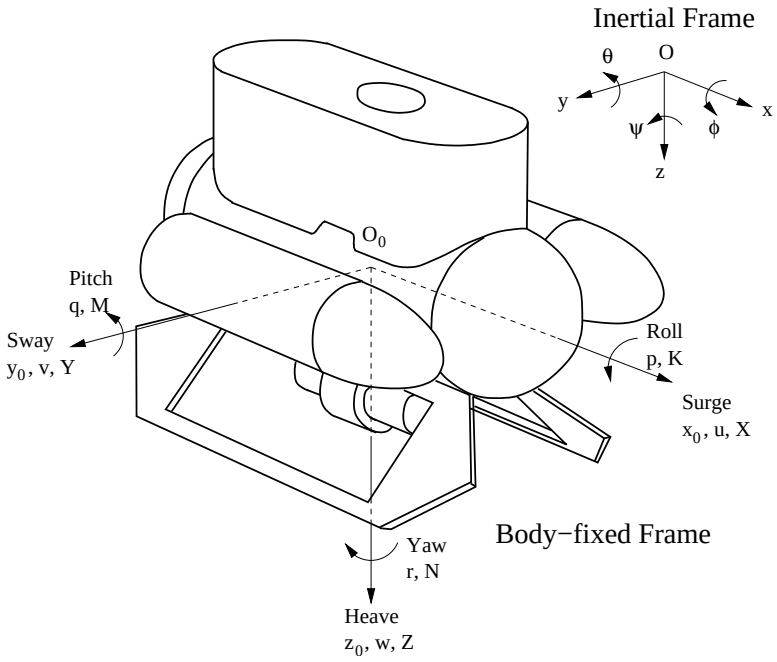


Figure 2.2: Body-fixed and inertial reference frames

$\eta_2 = [\phi,\ \theta,\ \psi]^T$: orientation (Bryant angles) describing the relation from the inertial frame to the body-fixed frame;
$\eta = [\eta_1^T,\ \eta_2^T]^T$: position and orientation in inertial frame;

$\nu_1 = [u,\ v,\ w]^T$: translational velocities in body-fixed frame;
$\nu_2 = [p,\ q,\ r]^T$: rotational velocities in body-fixed frame;
$\nu = [\nu_1^T,\ \nu_2^T]^T$: translational and rotational velocities in body-fixed frame;

$\tau_1 = [X,\ Y,\ Z]^T$: forces acting on the vehicle in body-fixed frame.
$\tau_2 = [K,\ M,\ N]^T$: moments acting on the vehicle in body-fixed frame.
$\tau = [\tau_1^T,\ \tau_2^T]^T$: forces and moments acting on the vehicle in body-fixed frame.

The translational velocity of the vehicle expressed in the body-fixed frame is expressed as translational velocity in the inertial frame through the following transformation [9]

$$\dot{\eta}_1 = J_1(\eta_2)\nu_1, \tag{2.1}$$

where

$$J_1(\eta_2) = \begin{bmatrix} \cos\psi\cos\theta & -\cos\psi + \cos\psi\sin\theta\sin\phi & \sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi \\ \sin\psi\cos\theta & \cos\psi\cos\phi + \sin\psi\sin\theta\sin\phi & -\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix}$$

The rotational velocity of the vehicle expressed in the body-fixed frame is expressed as rotational velocity in the inertial frame through the following transformation [9]

$$\dot{\eta}_2 = J_2(\eta_2)\nu_2, \tag{2.2}$$

where

$$J_2(\eta_2) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix}$$

Note that J_2 above is singular for pitch angle $\theta = \pm 90°$, where the inverse transformation does not exist. However, the VideoRay Pro III is unlikely to pitch near $\pm 90°$ due to the weight distribution. For this reason we choose to define the transformation matrices J_1 and J_2 in terms of the familiar and widely used Bryant angles.

2.2.3 Rigid Body Dynamics

The translational and rotational motion of a general rigid body with six degrees of freedom is formulated as follows [9]:

$$m\left(\mathring{v}_0 + \omega \times v_0 + \mathring{\omega} \times r_G + \omega \times (\omega \times r_G)\right) = f_0 \tag{2.3}$$

$$I_0\mathring{\omega} + \omega \times (I_0\omega) + mr_G \times (\mathring{v}_0 + \omega \times v_0) = m_0 \tag{2.4}$$

where

I_0 denotes the inertia tensor defined at the origin of the body-fixed frame,
$r_G = [x_G,\ y_G,\ z_G]^T$ is the vector from the frame origin to the center of gravity.
ω denotes the rotation velocity vector,
v is the translational velocity vector,

m is the rigid body mass,
f_0 constitutes the force vector acting on the vehicle,
m_0 constitutes the moments vector acting on the vehicle
$\mathring{v}_0$ denotes the time derivative of velocity vector in the body-fixed reference frame.
$\mathring{\omega}$ denotes the time derivative of rotational velocity vector in the body-fixed reference frame.

According to the notation defined in Section 2.2.2, applying substitution of $\tau_1 = f_0$, $\tau_2 = m_0$, ν_1 as the translational velocity vector and ν_2 as the rotational velocity vector to these equations yields

$$\begin{aligned}
m[\dot{u} - vr + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(pr + \dot{q})] &= X && (2.5)\\
m[\dot{v} - wp + ur - y_G(r^2 + p^2) + z_G(qr - \dot{p}) + x_G(qp + \dot{r})] &= Y && (2.6)\\
m[\dot{w} - uq + vp - z_G(p^2 + q^2) + x_G(rp - \dot{q}) + y_G(rq + \dot{p})] &= Z && (2.7)\\
I_x\dot{p} + (I_z - I_y)qr - (\dot{r} + pq)I_{xz} + (r^2 - q^2)I_{yz} + (pr - \dot{q})I_{xy} & \\
+m[y_G(\dot{w} - uq + vp) - z_G(\dot{v} - wp + ur)] &= K && (2.8)\\
I_y\dot{q} + (I_x - I_z)rp - (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + (qp - \dot{r})I_{yz} & \\
+m[z_G(\dot{u} - vr + wq) - x_G(\dot{w} - uq + vp)] &= M && (2.9)\\
I_z\dot{r} + (I_y - I_x)pq - (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + (rq - \dot{p})I_{zx} & \\
+m[x_G(\dot{v} - wp + ur) - y_G(\dot{u} - vr + wq)] &= N && (2.10)
\end{aligned}$$

Since the body-fixed frame is chosen to be coincident with the three principal axes of the VideoRay Pro III, we have $I_{xz} = I_{yz} = I_{xy} = 0$. Now, the rigid body dynamics of the vehicle can be expressed in matrix form as

$$\mathbf{M}_{RB}\dot{\nu} + \mathbf{C}_{RB}(\nu)\nu = \tau_{RB} \qquad (2.11)$$

where

$$\mathbf{M}_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_{xx} & 0 & 0 \\ mz_G & 0 & -mx_G & 0 & I_{yy} & 0 \\ -my_G & mx_G & 0 & 0 & 0 & I_{zz} \end{bmatrix} \tag{2.12}$$

$$\mathbf{C}_{RB} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -m(y_Gq + z_Gr) & m(y_Gp + w) & m(z_Gp - v) \\ m(x_Gq - w) & -m(z_Gr + x_Gp) & -m(z_Gq + u) \\ m(x_Gr + v) & m(y_Gr - u) & -m(x_Gp + y_Gq) \end{bmatrix}$$
$$\begin{bmatrix} m(y_Gq + z_Gr) & -m(x_Gp - w) & -m(x_Gr + v) \\ -m(x_Gq + w) & m(z_Gr + x_Gp) & -m(y_Gr - u) \\ -m(x_Gr - v) & -m(z_Gq + u) & m(x_Gp + y_Gq) \\ 0 & I_{zz}q & -I_{yy}p \\ -I_{zz}q & 0 & I_{xx}p \\ I_{yy}q & -I_{xx}p & 0 \end{bmatrix} \tag{2.13}$$

2.2.4 Hydrodynamics - Equations of Motion

The external forces and moments acting on an underwater vehicle can be classified as:

- *added mass* due to the inertia of the surrounding fluid,
- hydrodynamic damping due to effects like skin friction, vortex shedding, and energy carried away by generated surface waves,
- restoring forces due to the vehicle's weight and buoyancy,
- currents
- thruster/propeller forces

- control surface/rudder forces

The added mass is a concept of pressure-induced forces and moments due to a forced harmonic motion of the body, which are proportional to the acceleration of the body [9]. Therefore, the added mass forces and the acceleration will be 180 degrees out of phase to the forced harmonic motion:

$$-\mathbf{M}_A\dot{\nu} - \mathbf{C}_A(\nu)\nu$$

The hydrodynamic damping forces usually consist of linear and quadratic damping terms. These are in the opposite direction to the vehicle's velocity and can be expressed as a sum of these two terms:

$$-\mathbf{D}_{lin.}(\nu)\nu - \mathbf{D}_{quad.}(\nu)\nu$$

Restoring forces, sometimes also called hydrostatic forces, consist of gravity force and buoyancy force. These are functions of the vehicle's position and are denoted as $\mathbf{g}(\eta)$. The propulsion forces and moments are the control effects acting on the vehicle, denoted as τ.

In basic hydrodynamics, it is common to assume that the hydrodynamic forces and moments on a rigid body can be linearly superposed [8]. Hence, the external forces and moments acting on the vehicle can be expressed as:

$$\tau_{RB} = \tau_H + \tau_E + \tau = -\mathbf{M}_A\dot{\nu} - \mathbf{C}_A(\nu)\nu - \mathbf{D}_{lin.}(\nu)\nu - \mathbf{D}_{quad.}(\nu)\nu + \tau_E + \tau - \mathbf{g}(\eta) \quad (2.14)$$

In order to simplify this model, currents will be neglected, *i.e.*, $\tau_E = 0$.

Based on this assumption, the mathematical model of an underwater vehicle can be expressed, with respect to a local body-fixed reference frame, by the nonlinear equations of motion in matrix form [9]:

$$\mathbf{M}\dot{\nu} + \mathbf{C}(\nu)\nu + \mathbf{D}(\nu)\nu + \mathbf{g}(\eta) = \tau \quad (2.15)$$

$$\dot{\eta} = \mathbf{J}(\eta)\nu \quad (2.16)$$

where:

$$\begin{aligned} \mathbf{M} &= \mathbf{M}_{\mathrm{RB}} + \mathbf{M}_{\mathrm{A}}, \\ \mathbf{C}(\nu) &= \mathbf{C}_{\mathrm{RB}}(\nu) + \mathbf{C}_{\mathrm{A}}(\nu), \end{aligned}$$

$\mathbf{D}(\nu)$ $= \mathbf{D}_{\text{quad.}}(\nu) + \mathbf{D}_{\text{lin.}}(\nu)$, see Section 2.2.5,
$\mathbf{g}(\eta)$ is the hydrostatic restoring force matrix, see Section 2.2.6,
τ is the thruster input vector;
$\mathbf{J}(\eta)$ is the coordinate transform matrix which brings the inertial frame into alignment with the body-fixed frame:

$$\mathbf{J}(\eta) = \begin{bmatrix} \mathbf{J}_1(\eta) & 0 \\ 0 & \mathbf{J}_2(\eta) \end{bmatrix}$$

The mass matrix $\mathbf{M}$ consists of the rigid body component $\mathbf{M}_{RB}$ and an added mass component $\mathbf{M}_A$. The centripetal and Coriolis force matrix $\mathbf{C}$ consists of the rigid body component $\mathbf{C}_{RB}$ and an added mass component $\mathbf{C}_A$. The damping matrix $\mathbf{D}$ consists of a linear drag term $\mathbf{D}_{lin.}$ and a quadratic drag term $\mathbf{D}_{quad.}$. The quadratic drag term is significant when the vehicle is moving at higher speed, while the linear drag term predominates at low speeds.

2.2.5 Hydrodynamic Forces

The forces applied to the underwater vehicle by the surrounding fluid medium can be broken down into *added mass*, *lift*, *drag*, and *hydrostatic* or *restoring* forces. Aside from the hydrostatic forces, exact analytical expressions for these forces are very difficult to obtain. In practice, they are described in terms of corresponding hydrodynamic coefficients, which are estimated with first order Taylor series expansion. These coefficients are expressed in the form of hydrodynamic derivatives in accordance with the SNAME (1950) [1] notation.

For example, the axial quadratic drag force X on a body moving at a velocity u in a fluid medium is modeled as:

$$X = -(\frac{1}{2}\rho C_d A_f)u|u| = X_{u|u|}u|u|,$$

which implies that the drag force derivative in the *surge* direction with respect to $u|u|$ is:

$$X_{u|u|} = \frac{\partial X}{\partial(u|u|)} = -\frac{1}{2}\rho C_d A_f.$$

The task of modeling the vehicle then becomes analytically approximating or experimentally determining the coefficients to complete the vehicle's equations of motion, Equation 2.15 and Equation 2.16.

Added Mass

The concept of added mass refers to pressure-induced forces and moments proportional to the acceleration of the body. For rigid body dynamics, the added mass forces and moments τ_A can be expressed in terms of an added inertia matrix $\mathbf{M}_A$ and a matrix of hydrodynamic centripetal and Coriolis terms $\mathbf{C}_A$:

$$\mathbf{M}_A\dot{\nu} + \mathbf{C}_A(\nu)\nu = \tau_A. \tag{2.17}$$

In general, the motion of an underwater vehicle moving at high speed will be highly nonlinear and coupled. Since the VideoRay Pro III is port-starboard and top-bottom geometrically symmetric, and assuming the vehicle moves at low speed, simple expressions for $\mathbf{M}_A$ and $\mathbf{C}_A$ can be obtained as:

$$\mathbf{M}_A = \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{\dot{v}} & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{\dot{w}} & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{\dot{p}} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{\dot{q}} & 0 \\ 0 & 0 & 0 & 0 & 0 & N_{\dot{r}} \end{bmatrix} \tag{2.18}$$

$$\mathbf{C}_A = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix} \tag{2.19}$$

In practice, this diagonal approximation is reasonable for the VideoRay Pro III, due to the fact that the off-diagonal elements of the inertia matrix are much smaller than the diagonal ones.

Hydrodynamic Damping

Since the VideoRay Pro III underwater vehicle is symmetric about the x-z plane, and close to symmetric about y-z plane, we assume that the motions in *surge*, *sway*, *pitch* and *yaw* are decoupled [9]. Although it is not symmetric about the x-y plane, the *surge* and *heave* motions are considered to be decoupled because when the vehicle is operated at relatively low speed, the coupling effects can be neglected. This suggests a diagonal structure for $\mathbf{D}_{lin.}$ and $\mathbf{D}_{quad.}$:

$$\mathbf{D}_{\text{lin.}}(\nu) = \begin{bmatrix} X_u & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_w & 0 & 0 & 0 \\ 0 & 0 & 0 & K_p & 0 & 0 \\ 0 & 0 & 0 & 0 & M_q & 0 \\ 0 & 0 & 0 & 0 & 0 & N_r \end{bmatrix} \tag{2.20}$$

$$\mathbf{D}_{\text{quad.}}(\nu) = \begin{bmatrix} X_{u|u|}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{v|v|}|v| & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{w|w|}|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{p|p|}|p| & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{q|q|}|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & N_{r|r|}|r| \end{bmatrix} \tag{2.21}$$

As seen later in Section 2.4, a series of experimental tests were performed to verify these assumptions. The results indicate that the coupling effects are relatively small and can be neglected. Hence, the resulting added mass matrix and drag matrices are approximated reasonably well with diagonal matrices.

2.2.6 Hydrostatic Forces

For underwater vehicles, the gravitational and buoyant forces are called hydrostatic forces or restoring forces. The gravitational force acts through the center of gravity $\mathbf{r}_G = [x_G,\ y_G,\ z_G]^T$ of the vehicle, which is assumed to be coincident with the center of mass. Similarly, the buoyant force acts through the center of buoyancy $\mathbf{R}_B = [x_B,\ y_B,\ z_B]^T$.

For an underwater vehicle, let the submerged weight of the body be W and the buoyancy force be B, then the restoring force and moment vector in the body-fixed frame is represented with Euler angles as

$$\mathbf{g}(\eta) = \begin{bmatrix} (W-B)\sin(\theta) \\ -(W-B)\cos(\theta)\sin(\phi) \\ -(W-B)\cos(\theta)\cos(\phi) \\ y_B B\cos(\theta)\cos(\phi) - z_B B\cos\theta\sin(\phi) \\ -z_B B\sin(\theta) - x_B B\cos(\theta)\cos(\phi) \\ x_B B\cos(\theta)\sin(\phi) + y_B B\sin(\theta) \end{bmatrix} \tag{2.22}$$

2.2.7 Theoretical Parameter Estimation

Theoretically, the hydrodynamic derivatives can be determined using an approach called strip theory [25]. Fossen [9] provided some two-dimensional added mass coefficients. If the vehicle is divided into a number of strips, the added mass for each 2D strip can be computed and summed over the length of the body to get the 3D hydrodynamic derivative. Besides the added mass, the drag coefficients can also be determined with the application of strip theory. In this way, the hydrodynamic derivatives can be completely determined according to the vehicle's geometric properties, even before the vehicle is built. However, the derivatives produced using this approach are often inaccurate and sometimes unsatisfactory. Validation of these derivatives is always desired.

This approach has been implemented to model the VideoRay's added mass and damping derivatives through the strip theory. The resultant coefficients are shown in Table 2.1. More importantly, the coefficients in translational directions estimated using strip theory are in good agreement with those later obtained by experiment.

2.3 Decoupled Models

The fact that

1. the weight and buoyancy distribution of the VideoRay Pro III will always force the vehicle to return back to the zero pitch and zero roll state. Therefore, we assume for

Added mass

	Analytical*	Experimental
$X_{\dot{u}}$	1.94	NA
$Y_{\dot{v}}$	6.05	NA
$Z_{\dot{w}}$	3.95	NA
$K_{\dot{p}}$	3.26×10^{-2}	NA
$M_{\dot{q}}$	1.75×10^{-2}	NA
$N_{\dot{r}}$	3.21×10^{-2}	1.18×10^{-2}

Linear drag coefficients

	Analytical*	Experimental
X_u	2.30	0.95
Y_v	8.01	5.87
Z_w	5.81	3.70
K_p	0.0009	NA
M_q	0.0012	NA
N_r	0.0048	0.023

Quadratic drag coefficients

	Analytical*	Experimental
$X_{u\|u\|}$	8.28	6.04
$Y_{v\|v\|}$	23.69	30.73
$Z_{w\|w\|}$	20.52	26.36
$K_{p\|p\|}$	0.0048	NA
$M_{q\|q\|}$	0.0069	NA
$N_{r\|r\|}$	0.0089	0.45

* obtained by using strip theory.

Table 2.1: Hydrocoefficients for the model of VideoRay Pro III

all time $\phi = 0$, $\theta = 0$, $p = 0$ and $q = 0$

2. the thrusters of the VideoRay Pro III only have effect in surge, heave and yaw motion.

suggest that the system can be decoupled into two non-interacting subsystems:

1. x, y, ψ, u, v, r for horizontal plane motion
2. z, w for vertical plane motion

The decomposition also supports the idea that any control action for the surge direction is implemented using balanced thrusts from both side thrusters; and any control action for the yaw direction is implemented using differential thrust. The vertical thruster provides the control thrust for the heave direction.

Assuming the vehicle is always in the zero-pitch and zero-roll state, *i.e.*, , $\phi = 0$ and $\theta = 0$, we can write the decoupled models as follows.

- The model for horizontal plane motion:

$$m_{11}\dot{u} = -m_{22}vr + X_u u + X_{u|u|}u|u| + X, \tag{2.23}$$

$$m_{22}\dot{v} = m_{11}ur + Y_v v + Y_{v|v|}v|v|, \tag{2.24}$$

$$J\dot{r} = N_r r + N_{r|r|}r|r| + N, \tag{2.25}$$

where

m_{11} = the (1,1) entry of the vehicle inertia matrix M,
m_{22} = the (2,2) entry of the vehicle inertia matrix M,
J = vehicle's moment of inertia about the z axis, which is the (6,6) entry of the vehicle inertia matrix M,
X_u,$X_{u|u|}$ = linear and quadratic hydrodynamic coefficients in the surge direction,
Y_v,$Y_{v|v|}$ = linear and quadratic hydrodynamic coefficients in the sway direction,
X = external force acting on the vehicle in the surge direction,
N = external torque acting on the vehicle about the z axis.

- The model for vertical plane motion:

$$m_{33}\dot{w} = Z_w w + Z_{w|w|} w|w| + Z \tag{2.26}$$

where

$$\begin{aligned} m_{33} &= \text{the (3,3) entry of the vehicle inertia matrix } M, \\ Z_w, Z_{w|w|} &= \text{linear and quadratic hydrodynamic coefficients in the heave direction,} \\ Z &= \text{external force acting on the vehicle in the heave direction,} \end{aligned}$$

This decoupled model will facilitate the design for the trajectory tracking controllers, which will be described in Chapter 4.

2.4 Experimental Parameter Identification

The problem of modeling the VideoRay Pro III is now becoming the estimation and identification of the vehicle's mass, moments of inertia, hydrodynamic derivatives and thruster coefficients in Equation (2.15). In considering only the decoupled motions for the VideoRay Pro III, the parameters of interest are the translational drag derivatives in the *surge*, *heave*, *sway* directions, and the rotational drag derivatives in the *yaw* direction. These parameters will be determined by experiment.

The inertia matrix in Equation (2.15) consists of the vehicle's mass and moments of inertia about its three principal axes. In order to estimate the moments of inertia, a series of oscillation experiments with a small swing angle about vehicle's principal axes were performed. By measuring vehicle's oscillating frequency, the moments of inertia I_{xx}, I_{yy} and I_{zz} can be determined. (see Table 2.2 for the results).

Typically, determination of the hydrodynamic derivatives of a vehicle is performed experimentally in towing tank tests or in flumes with controlled flowing water. A series of tests were performed using a flume in the Experimental Fluid Lab at the University of Waterloo. The vehicle was mounted on a horizontal-bending mechanism and submerged in the water. The water flow rate is controlled manually by valves. The hydrodynamic forces acting on the vehicle are transferred to the horizontal-bending mechanism so that the horizontal force and the bending force can be measured by two load cells (See Figure 2.3). Using a data acquisition system, the forces are logged by a personal computer.

Parameter	Value	Units
L	3.60×10^{-1}	m
W	3.50×10^{-1}	m
H	2.30×10^{-1}	m
I_{xx}	2.28×10^{-2}	kg·m^2
I_{yy}	2.39×10^{-2}	kg·m^2
I_{zz}	2.53×10^{-2}	kg·m^2

Table 2.2: Dimensions and moments of inertia of VideoRay Pro III

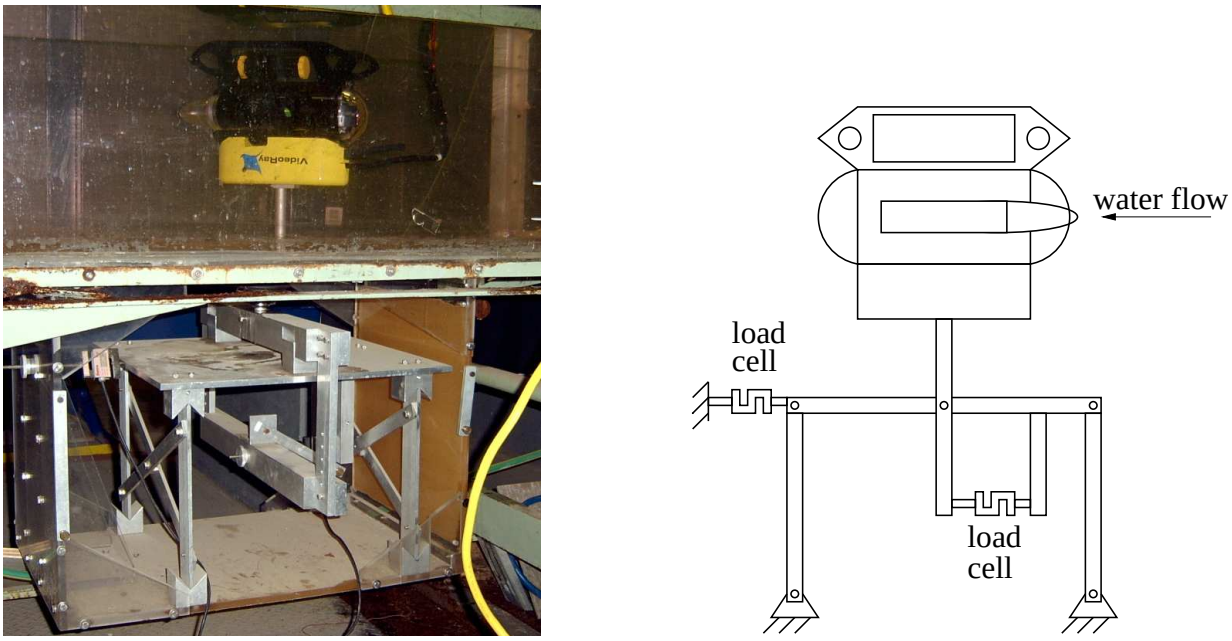


Figure 2.3: horizontal-bending mechanism in the flume test

2.4.1 Thruster Parameters

In Equation (2.15), the thruster input vector τ consists of the thruster forces and moments acting on the vehicle. This is a vector function of the thrusters' forces and their configuration.

An underwater vehicle's thrusters, both for propulsion and directional control, are highly nonlinear actuators. For a fixed pitch propeller, the force (thrust) T depends on the *advance speed* u_a, and the propeller rate n, (see Figure 2.4) as follows [4]:

$$T = \rho D^4(\alpha_1 + \alpha_2 \frac{u_a}{nD})n|n|, \tag{2.27}$$

where ρ is the water density, D is the diameter of propeller, α_1 and α_2 are constants given by the propeller's property, u_a is related through the speed of the vehicle relative to the ambient water.

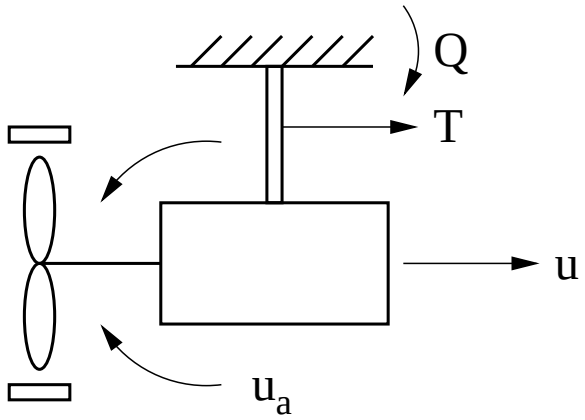


Figure 2.4: Schematic drawing of a propeller

A comprehensive study on thrusters and their influence on underwater vehicle maneuverability has been produced [37]. By considering the energy balance of a control volume about a thruster, and neglecting the effect of the advance speed u_a, simplified nonlinear equations for thrust T can be derived as:

$$\dot{n} = \beta\tau_{motor} - \alpha n|n|, \tag{2.28}$$

$$T = C_t n|n| \tag{2.29}$$

where τ_{motor} is the input torque supplied by the thruster's motor, β, α and C_t are thruster constants.

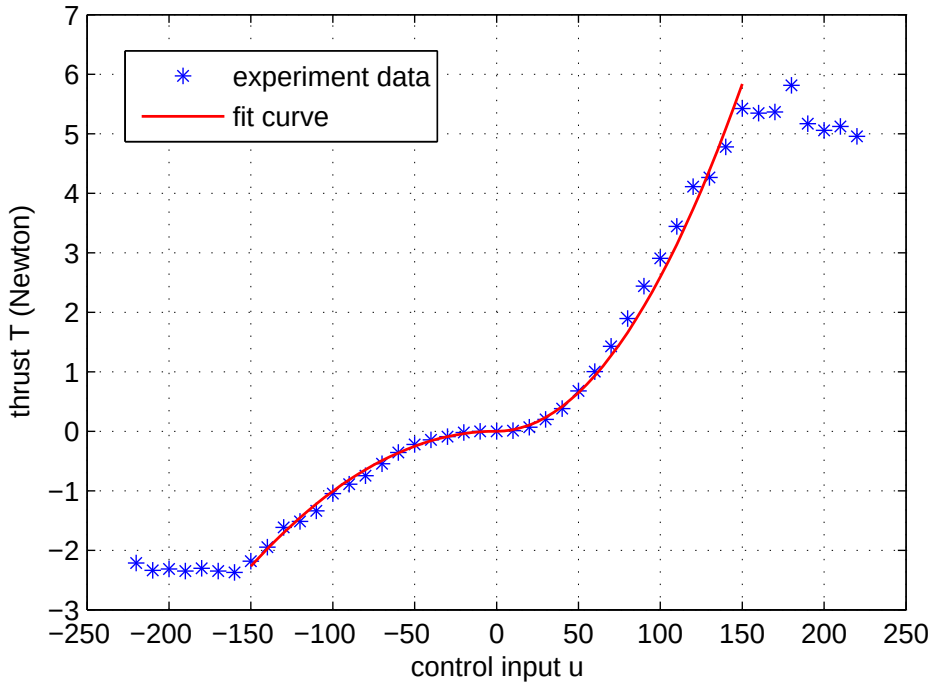


Figure 2.5: Output thrust vs. input signal for port/starboard thrusters

The VideoRay Pro III has 3 thrusters: port, starboard and vertical. Each one has its own drive which controls the rotational speed. Since the diameter and the mass of the propellers and their driving motors are small, the dynamics of the thruster control system in Equation 2.28 is much faster than the dynamics of the vehicle. For this reason, the propeller dynamics are neglected.

The C_T parameter from Equation 2.29 needs to be identified experimentally. The vehicle was mounted on the horizontal-bending mechanism where the thrust of the horizontal thrusters and vertical thruster were measured and recorded at various thruster control signals. The least squares method was applied to compute the coefficients for the port/starboard thrusters and the vertical thruster.

Mapping of the output thrust versus the thruster input for the two horizontal thrusters

is shown in Figure 2.5. The estimated thruster coefficients are shown in Table 2.3. Since the geometry of the propellers is non-symmetrical, the forward and backward thrusts are also non-symmetrical. It can be seen that the propellers are more efficient driving forward than backward. Another property for the propellers is that the thrust saturatutes when the control input is beyond the range of -150 and +150. Therefore, the effective will be restricted in this range.

	C_t (N)	
thruster	forward	backward
port/starborad	2.59×10^{-4}	1.01×10^{-4}
vertical	1.19×10^{-4}	7.53×10^{-5}

Table 2.3: Thruster coefficients

2.4.2 Experimental Set-up for Derivatives in Translational Motions

Translational hydrodynamic drag forces in the x, y and z directions are modeled as the sum of linear and quadratic terms [9]. For example, the hydrodynamic drag in the x direction due to surge motion is expressed as:

$$\text{Drag Force} = X_u u + X_{u|u|} u|u| \tag{2.30}$$

where u is the surge velocity, X_u is the surge drag force derivative with respect to u, $X_{u|u|}$ is the surge drag force derivative with respect to $u|u|$. When the vehicle moves at low speed, the linear drag term is dominant, while the quadratic drag term is dominant when the vehicle is moving at higher speed. These coefficients are part of the entries in the drag matrix in Equation (2.15).

In determining the drag coefficients, a number of flume experiments were performed using the horizontal-bending mechanism to test the drag force under various water flow speeds up to 0.55 m/s. Figure 2.6, 2.7 and 2.8 show the experimental data and resulting fit curves for the drag forces in the surge, sway and heave directions.

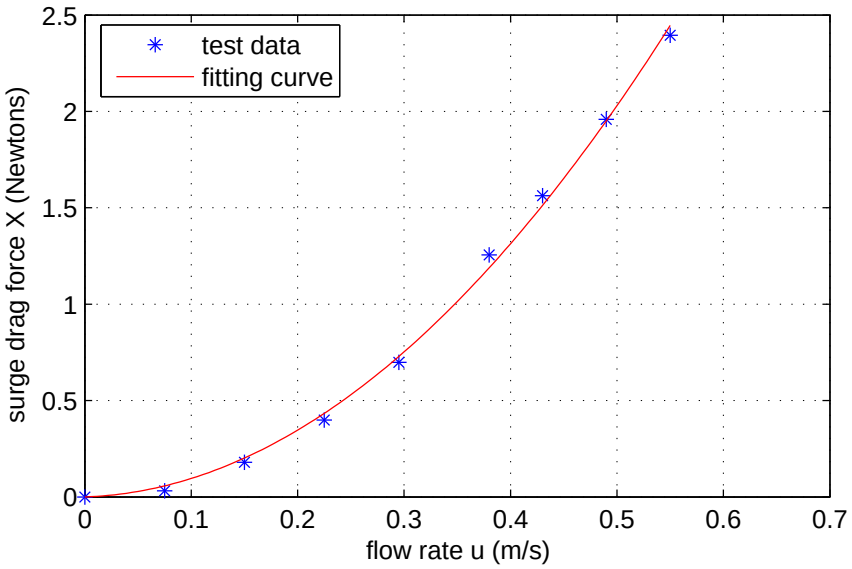


Figure 2.6: Drag force in surge direction: experiment data and fit curve

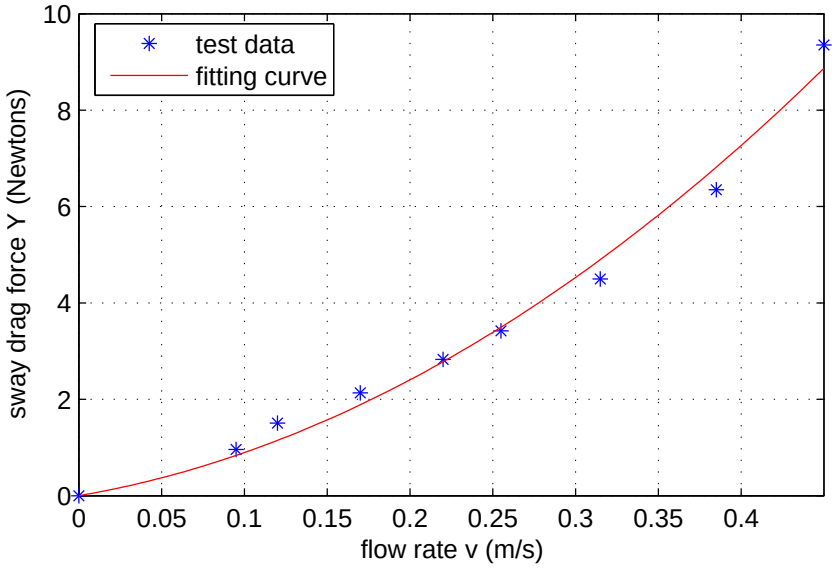


Figure 2.7: Drag force in sway direction: experiment data and fit curve

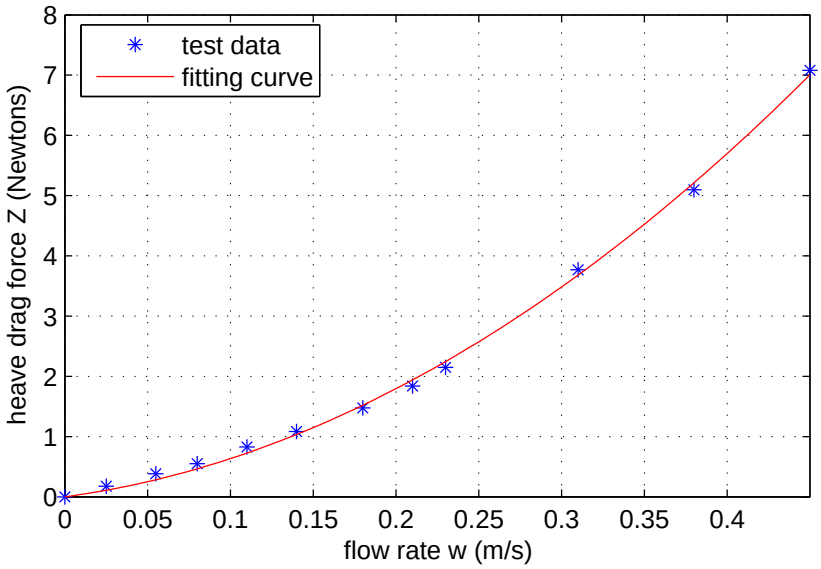


Figure 2.8: Drag force in heave direction: experiment data and fit curve

The hydrodynamic forces in the heave and sway directions were also tested and recorded while the vehicle was moving in the surge direction. Figure 2.9 shows the relationship between the change of hydrodynamic force in heave as a function of the surge speed. The results demonstrate that change in the heave direction drag force resulting from surge motion is less than one tenth of the drag force in the surge direction. However, the heave direction drag force resulting from surge motion could be a result of inaccurate positioning of the vehicle during experiments, which causes a slight angle of attack to the water flow. Because its magnitude is relatively small, it can be neglected.

Figure 2.10 shows that the sway drag force is constant when tested in different surge speeds. Therefore, it is considered independent of the surge speed. This is expected since the vehicle is symmetrical about the x-z plane.

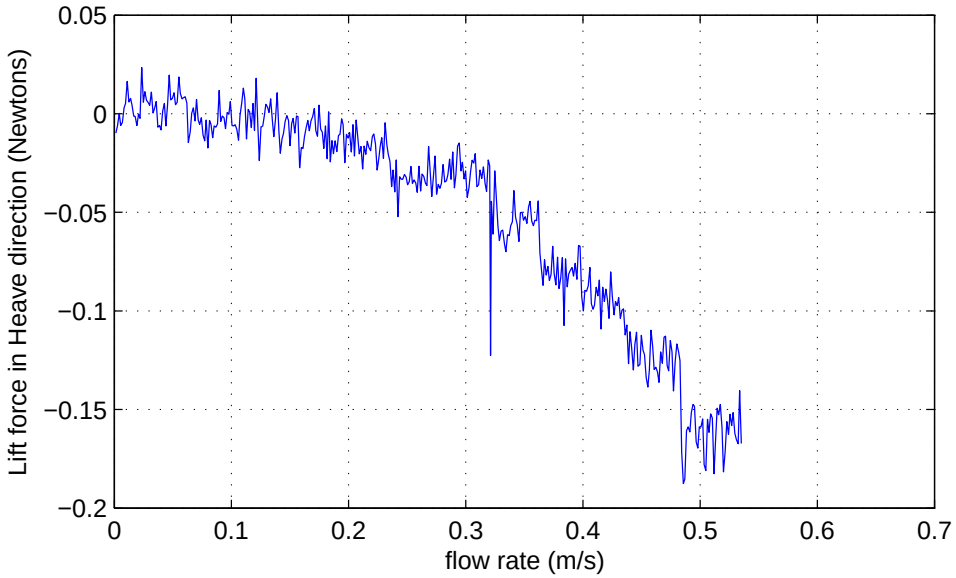


Figure 2.9: Heave lift force vs. surge speed

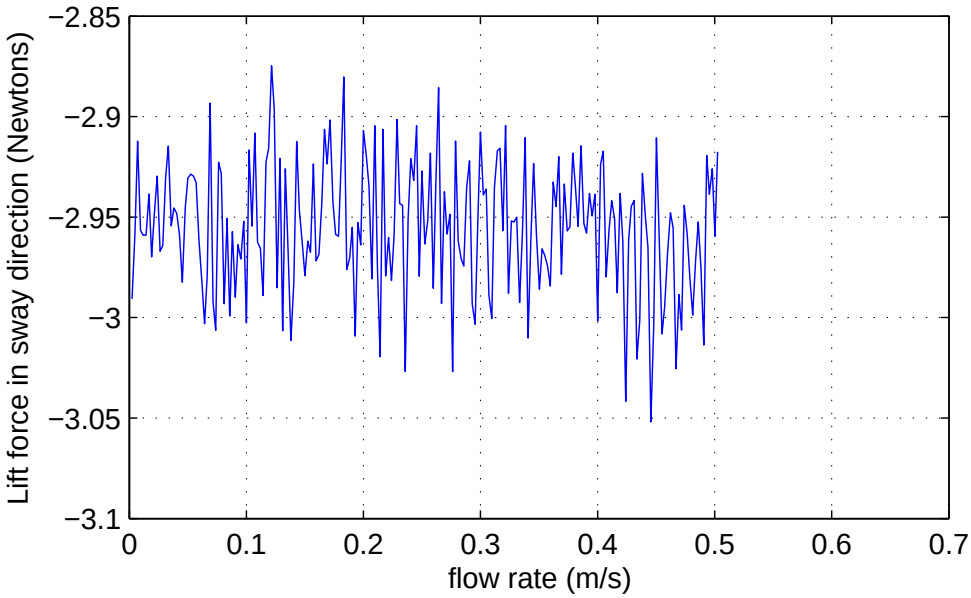


Figure 2.10: Sway drag force vs. surge speed

2.4.3 Experimental Set-up and Identification for the Yaw Movement

Accurate hydrodynamic derivatives for the *yaw* motion is essential for modeling the VideoRay Pro III. Because of the symmetry of the x-z and the y-z planes, the yaw motion is decoupled from other motions [9] and described by Equation 2.25. If we substitute parameters $a = N_r/J$, $b = N_{r|r|}/J$, $n = N$ and $c = 1/J$, and introduce a bias parameter d, Equation 2.25 can be rewritten as:

$$\dot{r} = ar + br|r| + cn + d \tag{2.31}$$

where r is the state variable describing the yaw rate, n is the input variable describing the torque the thrusters exert on the vehicle, the unknown parameters a and b are the linear and quadratic drag coefficients, the unknown parameter c is the inverse of the vehicle's moment of inertia about the y-axis, including the rigid body and added mass.

To estimate the unknown parameters in Equation 2.31, we start by describing the system dynamics. The state variable r in Equation 2.31 is completely controllable by the control variable n and completely observable at discrete time instants $\{t_k\}_{k\geq 0}$ through the output variable $y(t_k)$, corrupted by the additive zero-mean noise $e(t_k)$. The system dynamics can be expressed as [22]:

$$\dot{r} = \phi_r(r(t), n(t))\theta_r \tag{2.32}$$

$$y(t_k) = r(t_k) + e(t_k) \tag{2.33}$$

where $\phi_r(r(t), n(t)) = [r \;\; r|r| \;\; n \;\; 1]$ is a row vector of the state and control input, $\theta_r = [a \; b \; c \; d]^T$ is a constant (unknown) parameter vector that characterizes the system dynamics.

The identification problem consists of estimating the unknown parameter vector θ_r on the basis of a finite number of discrete time measurements of input variable $\{n(t_k)\}$ and output variable $\{y(t_k)\}$. The parameter vector θ_r can be identified by minimizing the following cost function with the Least Squares method:

$$J_r(\theta_r) = \sum \epsilon(t_k)^2 \tag{2.34}$$

The cost function is a sum of squares of prediction errors $\epsilon(t_k)$, which are the difference between the observed output variable and the one-step-ahead prediction of the output

$\hat{y}(t_k)$:

$$\epsilon(t_k) = y(t_k) - \hat{y}(t_k) \tag{2.35}$$

If the measurement noise $e(t_k)$ is zero-mean, then the output variable is simplified as:

$$\hat{y}(t_k) = \hat{r}(t_k) \tag{2.36}$$

where $\hat{r}(t_k)$ is the expected state variable at time t_k.

The one-step-ahead prediction of the output variable $\hat{y}(t_k)$ can be obtained by integrating the state space equation in Equation (2.33) between two subsequent time instants t_{k-1} and t_k:

$$r(t_k) - r(t_{k-1}) = \left[\int_{t_{k-1}}^{t_k} \phi_r(\hat{r}(\tau), n(\tau))d\tau\right]\theta_r \tag{2.37}$$

From Equation (2.36), it is implied that $r(t_{k-1}) = \hat{y}(t_{k-1})$. The following estimate for the state variable r at time t_k is obtained as:

$$\hat{r}(t_k) = \hat{y}(t_{k-1}) + \Phi_k\theta_r \tag{2.38}$$

where

$$\Phi_k = \int_{t_{k-1}}^{t_k} \phi_r(\hat{r}(\tau), n(\tau))d\tau \tag{2.39}$$

Hence, the one-step-ahead prediction error of Equation 2.35 can be evaluated as:

$$\epsilon(t_k) = \hat{y}(t_k) - \hat{y}(t_{k-1}) - \Phi_k\theta_r \tag{2.40}$$

Inserting this prediction error into the cost function $J_r(\theta_r)$, we can find the parameter vector θ_r that minimizes the cost function on the basis of M observations through the Least Squares algorithm:

$$\theta_r = (\Phi(M)^T\Phi(M))^{-1}\Phi(M)^T Y(M) \tag{2.41}$$

where

$$\Phi(M) = \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_M \end{bmatrix},\ Y(M) = \begin{bmatrix} \hat{y}(t_1) - \hat{y}(t_0) \\ \hat{y}(t_2) - \hat{y}(t_1) \\ \vdots \\ \hat{y}(t_M) - \hat{y}(t_{M-1}) \end{bmatrix} \tag{2.42}$$

Thus, if we have M observations of the yaw rate r and control variable n, we can estimate the system parameters a, b, c and d. Furthermore, we can estimate the yaw dynamic parameters N_r and $N_{r|r|}$.

The experimental setup for the yaw motion is depicted in Figure 2.11. The vehicle is mounted on a pivot which allows the vehicle to rotate about its z-axis freely. An overhead video camera is placed on top of the vehicle to record its angular movement during the test. The vehicle is driven by the horizontal thrusters with a series of oscillating input signals, which have the same oscillating period and various amplitude from $n = 50$ to $n = 150$. The vehicle oscillates about its z-axis following the input signals. The measured rotational angles of the vehicle are shown in Figure 2.12. The estimated parameters are: $a = 0.62$, $b = 1.12$, $c = 26.95$ and $d = 0.0316$.

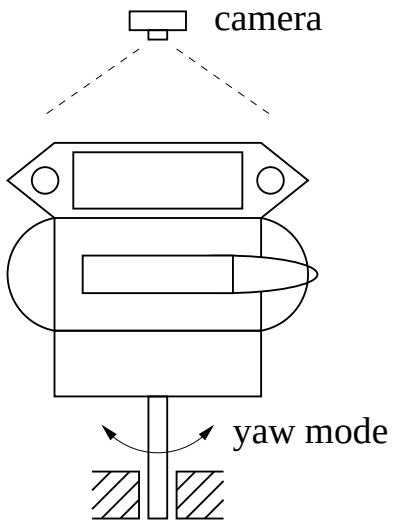


Figure 2.11: Experimental set-up for yaw motion

From the obtained values of a, b, and c, the corresponding hydrodynamic derivatives related to yaw motion N_r, $N_{r|r|}$ and $N_{\dot{r}}$ can be derived and the results are shown in Table 2.1.

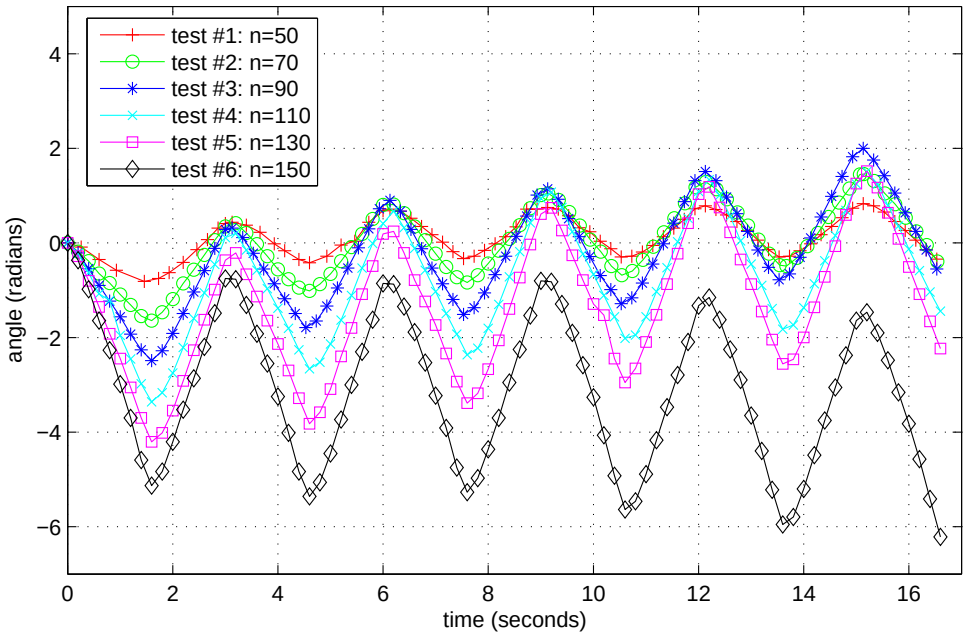


Figure 2.12: Test data for yaw motion

2.5 Model Verification

2.5.1 Surge mode

To verify the surge mode of the dynamic model of the VideoRay Pro III, a series of surge tests were performed in a pool. The movements of the vehicle were recorded with a video camera and the distance traveled analyzed and processed with Matlab. Figure 2.13 shows the observed and simulated surge speed with an applied thruster input of $n = 60$. The predicted surge speed with the dynamic model is $u = 0.51$m/s, which is a bit higher than the actual testing speed of 0.47m/s. This could be attributed to the effect of the tether on the vehicle, something not included in our dynamic model.

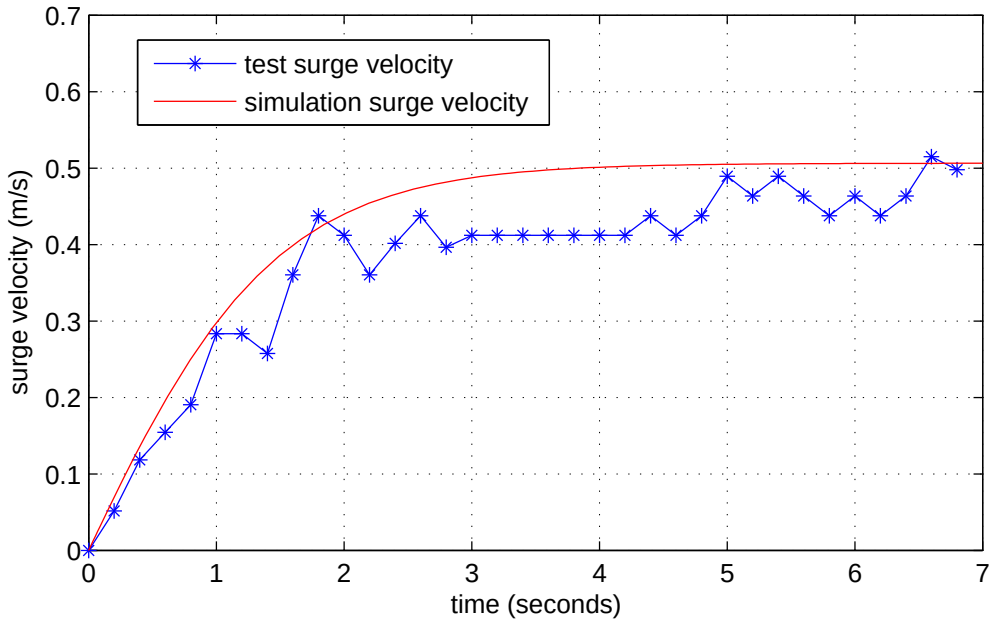


Figure 2.13: Surge test experiment data and simulation result

2.5.2 Yaw Mode

Figure 2.14 shows the experimental yaw angle and the simulated yaw angle with the model parameters determined in section 2.4.3. In this test, the thrusters input is $n = 150$ and oscillating period is $t = 1.5$ seconds. We see that the predicted yaw angles match the test results very well.

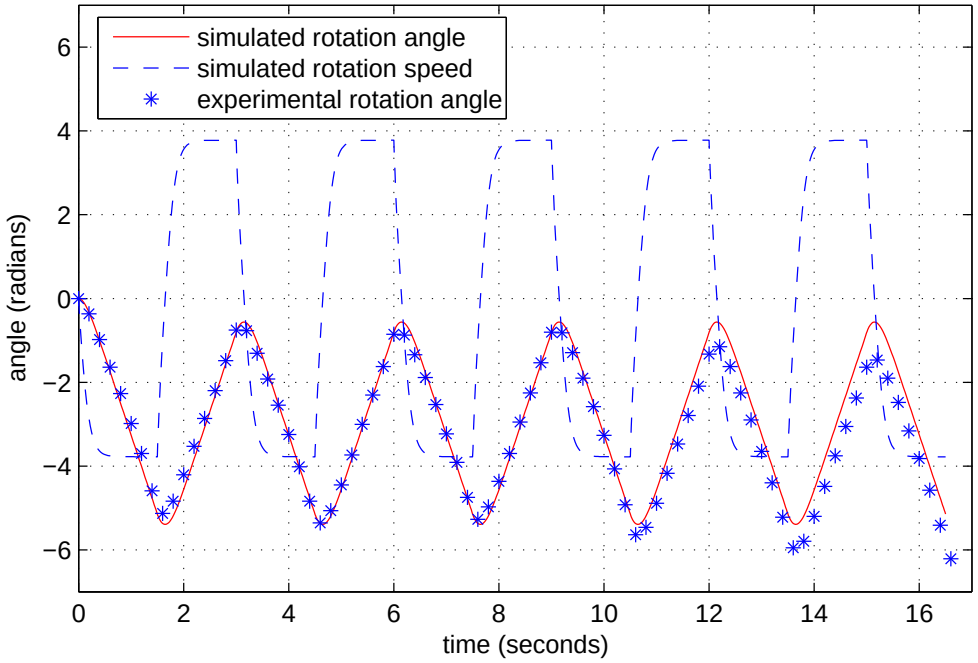


Figure 2.14: Identification for yaw motion (thruster inputs $n = \pm 150$, period $t = 1.5$ seconds)

Chapter 3

State Observer/Estimator Design

3.1 Introduction

Having accurate VideoRay Pro III motion information, namely its position information x, y, z, ψ and velocity information u, v, w, r, is crucial for the trajectory tracking controller to work properly. Unfortunately, among these parameters only the 3-dimension position information (x, y, z) and heading information (ψ) are available from the vehicle's sensor system and underwater acoustic positioning system; the velocity could not be measured directly. Also, the position information obtained through the measurement is uncertain due to noise and other imperfections. To handle this problem, filtering is applied to the measurements.

Filtering is used when estimating the system states as observations become available on-line. The Kalman filter (KF) is a popular choice for estimating the system state for linear systems with Gaussian process and measurement noise. To solve the problem of non-Gaussian, nonlinear filtering, the most often used algorithm is the extended Kalman filter (EKF). This filter is based on the principle of linearizing the measurements and evolution models using Taylor series expansion.

Unfortunately, the EKF has two significant drawbacks. First, it requires the derivation of the Jacobian matrices, *i.e.*, the linear approximation to the nonlinear functions, which can be complex and causes implementation difficulties. Second, these linearizations can lead to filter instability if the timestep intervals are not sufficiently small [18]. Besides

these drawbacks, the EKF is not suitable for discontinuous process models [17], where the representation of the nonlinear functions and probability distribution of interest is not adequate.

Another popular solution strategy for the general filtering problem is to use sequential Monte Carlo methods [11, 29, 7, 32, 6, 21], also known as particle filters. These methods allow for a complete representation of the posterior distribution of the states, so that any statistical estimates, such as mean, modes, kurtosis and variance, can be easily computed. Therefore, they can deal with any nonlinearities or distributions. This approach is very general, but can be very inefficient.

Recently, a new filter called the unscented Kalman filter (UKF) has been developed by Julier and Uhlmann [16, 34]. The UKF operates on the presumption that it is easier to approximate a Gaussian distribution than it is to approximate arbitrary nonlinear functions. Instead of linearizing using Jacobian matrices, the UKF uses a deterministic sampling approach to capture the mean and covariance estimates with a minimum set of sample points. It is shown that the UKF generates much better estimates of the mean and covariance of the system states than the EKF in various applications including road vehicle navigation [14], parameter estimation for time series modeling [35] *etc.*

In this chapter, we discuss the use of UKF as the state observer/estimator for the VideoRay Pro III autonomous control system. The notation and the general state-space model formulation are first introduced. Since the UKF is a direct extension to the EKF, the EKF is first discussed, followed by UKF in more detail.

3.2 Dynamic State-Space Model

By using $[\eta^T \ \nu^T]^T$ as the system state, τ as the control input, the equation of motion in Equation 2.15 can be generalized as a state-space model:

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}, \mathbf{m}_{t-1}) \tag{3.1}$$

$$\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{n}_t) \tag{3.2}$$

where

$\mathbf{x}$ is the state of the system with dimension n,

u denotes the control input observations,
y denotes the sensor output observations,
m the process noise with dimension n_m, and
n the measurement noise with dimension n_n

Our goal is to estimate the posterior distribution of the state variable $\mathbf{x}_t$ at time step t, given the knowledge of the output observations $\mathbf{y}_t$ at time t and the process control input $\mathbf{u}_{t-1}$ at time $t-1$.

3.3 The EKF and Unscented Kalman Filters

3.3.1 The Extended Kalman Filter

The EKF is a minimum mean-square-error (MMSE) estimator that has the recursive "predict-correct" structure. It is based on Taylor series expansion of the nonlinear functions $\mathbf{f}$ and $\mathbf{h}$ around the estimates $\bar{\mathbf{x}}_{t|t-1}$ of the state $\mathbf{x}_t$ [12]. For example

$$\mathbf{f}(\mathbf{x}_t) = \mathbf{f}(\bar{\mathbf{x}}_{t|t-1}) + \left.\frac{\partial \mathbf{f}(\mathbf{x}_t)}{\partial \mathbf{x}_t}\right|_{(\mathbf{x}_t = \bar{\mathbf{x}}_{t|t-1})} (\mathbf{x}_t - \bar{\mathbf{x}}_{t|t-1}) + \cdots \tag{3.3}$$

Using only the linear expansion terms (ignoring the higher order expansion terms), it is easy to derive the following update equations for the mean $\bar{\mathbf{x}}$ and covariance $\mathbf{P}$ of the Gaussian approximation to the posterior distribution of the states [12]:

- *predict* step:

$$\bar{\mathbf{x}}_{t|t-1} = \mathbf{f}(\bar{\mathbf{x}}_{t-1}, \bar{\mathbf{u}}_t, 0) \tag{3.4}$$
$$\mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1} \mathbf{F}_t^T + \mathbf{M}_t \mathbf{Q}_t \mathbf{M}_t^T \tag{3.5}$$

- *correct* step:

$$\mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^T [\mathbf{N}_t \mathbf{R}_t \mathbf{N}_t^T + \mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T]^{-1} \tag{3.6}$$
$$\bar{\mathbf{x}}_t = \bar{\mathbf{x}}_{t|t-1} + \mathbf{K}_t [\mathbf{y}_t - \mathbf{h}(\bar{\mathbf{x}}_{t|t-1}, \bar{\mathbf{u}}, 0)] \tag{3.7}$$
$$\mathbf{P}_t = \mathbf{P}_{t|t-1} - \mathbf{K}_t \mathbf{H}_t \mathbf{P}_{t|t-1} \tag{3.8}$$

where

$\bar{\mathbf{x}}_{t|t-1}$ is the predicted state mean,
$\mathbf{P}_{t|t-1}$ is the predicted state covariance,
$\mathbf{K}_t$ is known as the Kalman gain;
$\mathbf{Q}$ is the variance of the process noise;
$\mathbf{R}$ is the variance of the measurement noise;
$\mathbf{F} = \frac{\partial \mathbf{f}(\mathbf{x}_t)}{\partial \mathbf{x}_t}$ is the Jacobian of the process model, with respect to the states;
$\mathbf{M} = \frac{\partial \mathbf{f}(\mathbf{m}_t)}{\partial \mathbf{m}_t}$ is the Jacobian of the process model, with respect to the process noise;
$\mathbf{H} = \frac{\partial \mathbf{h}(\mathbf{x}_t)}{\partial \mathbf{x}_t}$ is the Jacobian of the measurement model, with respect to the states;
$\mathbf{N} = \frac{\partial \mathbf{h}(\mathbf{n}_t)}{\partial \mathbf{n}_t}$ is the Jacobian of the measurement model, with respect to the measurement noise.

3.4 The Unscented Kalman Filter

The unscented Kalman filter (UKF) is also a recursive MMSE estimator that has the "predict-correct" structure. It addresses some of the approximation issues of the EKF [16]. Because in the EKF, only the first order terms of the Taylor series expansion of nonlinear functions is used for approximating the original system, large errors are usually introduced in the posterior distribution estimates of the states. This is especially evident when the models are highly nonlinear and the local linearity assumption breaks down, in which situation the effects of higher order terms of the Taylor series expansion becomes significant.

Unlike the EKF, the UKF uses another way to solve this problem. Rather than approximating the nonlinear system with Taylor series expansion, it approximates the distribution of the state random variable. Like the EKF, the UKF will work for nonlinear systems with Gaussian random variables.

In the UKF, the state distribution is still represented by a Gaussian random variable (GRV). Instead of using the mean and variance to describe the distribution of the GRV, the UKF uses a minimal set of deterministrically chosen sample points. These sample points completely capture the true mean and the covariance of the GRV, and when propagated through the true non-linear system, captures the posterior mean and covariance accurately

to the second order for any nonlinearity. The errors of the UKF only exhibit in the third and higher orders.

In the next sections, we start by first explaining the *unscented transformation*, followed by the *scaled unscented transformation*, which is a generalizing extension of the unscented transformation. The scaled unscented transformation forms the algorithmic core of the unscented Kalman filter.

3.4.1 The unscented transformation

The unscented transformation (UT) is a method for calculating the statistics of a random variable which undergoes a nonlinear transformation. It was first proposed by Simon Julier and Jeffrey K. Uhlmann as a general method for approximating nonlinear transformations of probability distributions [17]. This method is based on the principle that it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function/transformation [17].

Consider propagating an n dimensional Gaussian random variable $\mathbf{x}$ through an arbitrary nonlinear function $\mathbf{g} : \mathbf{R}^n \rightarrow \mathbf{R}^m$ to generate $\mathbf{y}$,

$$\mathbf{y} = \mathbf{g}(\mathbf{x}) \tag{3.9}$$

Suppose the Gaussian random variable $\mathbf{x}$ has mean $\bar{\mathbf{x}}$ and covariance $\mathbf{P}$. To calculate the mean and covariance of $\mathbf{y}$ using the UT, we proceed as follows:

1. a set of $2n+1$ weighted samples or *sigma points* $S_i = \{W_i, \mathcal{X}_i\}$ are deterministically chosen so that they completely capture the true mean and covariance of the prior Gaussian random variable $\mathbf{x}$. A selection scheme that satisfies this requirement is

$$\begin{array}{lll} \mathcal{X}_0 = \bar{\mathbf{x}} & W_0 = \kappa/(n+\kappa) & i = 0 \\ \mathcal{X}_i = \bar{\mathbf{x}} + [\sqrt{(n+\kappa)\mathbf{P}}]_i & W_i = 1/[2(n+\kappa)] & i = 1, \ldots, n \\ \mathcal{X}_i = \bar{\mathbf{x}} - [\sqrt{(n+\kappa)\mathbf{P}}]_i & W_i = 1/[2(n+\kappa)] & i = 1+n, \ldots, 2n \end{array} \tag{3.10}$$

   where:

   κ is a scaling aprameter;
   $\sqrt{(n+\kappa)\mathbf{P}}$ is the ith row or column of the matrix square root of $(n+\kappa)\mathbf{P}$.

W_i is the weight associated with the ith point such that $\sum_{i=0}^{2n} W_i = 1$.

2. The transformed set of sigma points are evaluated for each of the $S_i = \{W_i, \mathcal{X}_i\}$ by

$$\mathcal{Y}_i = \mathbf{g}(\mathcal{X}_i) \quad i = 0, \ldots, 2n \tag{3.11}$$

3. the predicted mean of $\mathbf{y}$ is computed as:

$$\bar{\mathbf{y}} = \sum_{i=0}^{2n} W_i \mathcal{Y}_i \tag{3.12}$$

4. and the predicted covariance is computed as

$$\mathbf{P} = \sum_{i=0}^{2n} W_i (\mathcal{Y}_i - \bar{\mathbf{y}})(\mathcal{Y}_i - \bar{\mathbf{y}})^T \tag{3.13}$$

It is shown that these estimates of the mean and covariance are accurate to the second order of the Taylor series expansion of any nonlinear function $\mathbf{g}(\mathbf{x})$ [17]. In comparison, the EKF only calculates the *a posterior* mean and covariance accurately to the first order with all higher order moments truncated.

As an example, a comparison of the performance of the UT versus that of the linearization approach used in the EKF is shown in Figure 3.1 [34]. In this example, 5000 samples are drawn from a Gaussian distribution by a Monte Carlo approach and propagated through an arbitrarily highly nonlinear function. The posterior sample mean and covariance are then calculated as the truth data.This is shown in the left plot of Figure 3.1. Next, the posterior random variable's mean and covariance are calculated by a linearization method same as in the EKF, shown in the middle plot. The estimates calculated by the unscented transformation are shown in the right plot. It is evident that there is almost no bias error in the mean estimate, and the covariance estimate is much closer to the true covariance than that of the linearization method. The performance of UT is superior over the linearization approach as in the EKF.

Another advantage of the UT over linearization approach is that the UT doesn't need the Jacobian to be calculated or make any other approximation of $\mathbf{g}(\cdot)$. Also, the computation complexity is the same as the linearization method [17]. This property will make the implementation of the UT very easy.

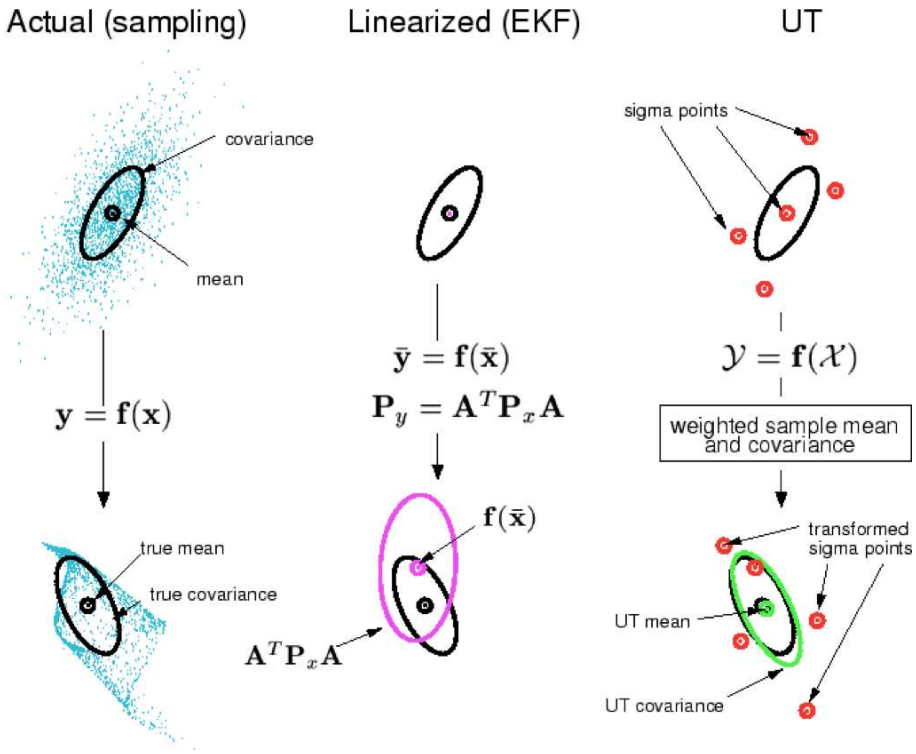


Figure 3.1: Example of UT for mean and covariance propagation [32]

The sigma point selection scheme used in the UT has two difficulties. The first difficulty is that the radius of the sphere that bounds all the sigma points is a function of $\sqrt{(n+\kappa)}$. Therefore, as n increases, the radius also increases. This will lead to a problem of sampling non-local effects. If the nonlinearities in question are very severe, this effect can lead to significant difficulties. Although the sigma points can be scaled towards or away from the mean point by a proper choice of κ, there is still a potential that the calculated covariance could be non-positive definite.

The second difficulty is that the sigma points are asymmetrically distributed about the mean point. Therefore, higher order effects such as the skew become more significant as the state dimension n increases.

To address these difficulties, the *scaled unscented transformation* was developed [13].

3.4.2 The scaled unscented transformation

To overcome the dimensional scaling effects, the scaled unscented transformation (SUT) replaces the original set of sigma points with a transformed set given by [13]:

$$\mathcal{X}_i' = \mathcal{X}_0 + \alpha(\mathcal{X}_i - \mathcal{X}_0) \quad i = 0, \ldots, 2n, \tag{3.14}$$

where α is a positive scaling parameter which can be made arbitrarily small to minimize higher order effects. With proper choice of α, the predicted covariance will be guaranteed positive semidefinite, and the second order accuracy in both the mean and covariance will also be preserved.

In [13], a method was proposed to select α in Equation 3.14. First, a set of sigma points $\mathcal{S} = \{W, \mathcal{X}\}$ is calculated using equation 3.10. Then they are transformed into the scaled set $\mathcal{S}' = \{W', \mathcal{X}'\}$ by

$$\begin{aligned} \mathcal{X}_i' &= \mathcal{X}_0 + \alpha(\mathcal{X}_i - \mathcal{X}_0) \\ W_i' &= \begin{cases} W_0/\alpha^2 + (1 - 1/\alpha^2) & i = 0 \\ W_i/\alpha^2 & i \neq 0 \end{cases} \end{aligned} \tag{3.15}$$

where α is the new sigma point *scaling* parameter. These two steps of selecting and scaling the sigma points can also be combined into a single step by setting

$$\lambda = \alpha^2(n + \kappa) - n \tag{3.16}$$

and selecting the sigma point set by

$$\begin{aligned}
\mathcal{X}_0 &= \bar{\mathbf{x}} \\
\mathcal{X}_i &= \bar{\mathbf{x}} + \left[\sqrt{(n+\lambda)\mathbf{P}}\right]_i \quad i = 1, \ldots, n \\
\mathcal{X}_i &= \bar{\mathbf{x}} - \left[\sqrt{(n+\lambda)\mathbf{P}}\right]_i \quad i = n+1, \ldots, 2n \\
W_0^{(m)} &= \lambda/(n+\lambda) \\
W_0^{(c)} &= \lambda/(n+\lambda) + (1 - \alpha^2 + \beta) \\
W_i^{(m)} &= W_i = 1/[2(n+\lambda)] \quad i = 1, \ldots, 2n
\end{aligned} \tag{3.17}$$

where the superscript (m) and (c) in the weighting parameters correspond to mean and covariance respectively. Because the weighting on the zeroth sigma point $\mathcal{X}_0$ directly affects the magnitude of the errors in the fourth and higher order terms for symmetric *a priori* distributions [13]. A third parameter, β, is thus introduced to incorporate prior knowledge of the distribution of $\mathbf{x}$ (for Gaussian distribution, $\beta = 2$ is optimal).

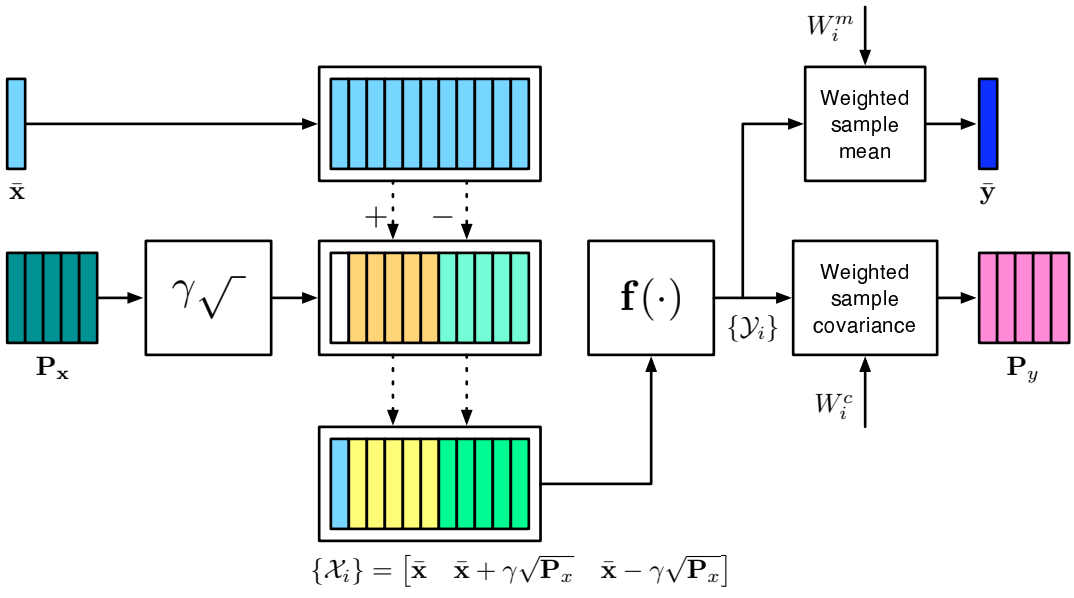


Figure 3.2: UT block diagram

A block diagram illustrating the procedures in performing the UT is shown in Figure 3.4.2. The algorithm is given as follows:

1. Choose the parameters κ, α and β. The parameter α determines the spread of the sigma points around $\bar{\mathbf{x}}$ and is usually set to a small positive value. The parameter κ is a secondary scaling parameter. β is used to incorporate the knowledge of the higher order moments of the distribution.

2. Calculate the set of $2n+1$ scaled sigma points and weights $\mathcal{S} = \{\mathcal{W},\ \mathcal{X}\}$ by using Equation 3.16 and Equation 3.17. Here n is the dimension of random variable $\mathbf{x}$.

3. Propagate each sigma point through the nonlinear transformation

$$\mathcal{Y}_i = \mathbf{f}(\mathcal{X}_i) \quad i = 0, \ldots, 2n \tag{3.18}$$

4. Calculate the mean $\bar{\mathbf{y}}$ and covariance $\mathbf{P}_y$ as follows

$$\bar{\mathbf{y}} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_i \tag{3.19}$$

$$\mathbf{P}_y = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{Y}_i - \bar{\mathbf{y}}][\mathcal{Y}_i - \bar{\mathbf{y}}]^T \tag{3.20}$$

3.4.3 An Example of Unscented Transformation for VideoRay Pro III

To compare the performance of the unscented transformation to the linearization method as in EKF, we take as an example the VideoRay Pro III running straight forward from position $(0,\ 0)$ in the x-y plane, with a heading angle of $\psi = 90$ degrees and a constant forward speed of $u = 1.2$ m/s and zero yaw rate $r = 0$. After one sampling period of 0.033 seconds (assuming the sampling frequency is 30 Hz), the vehicle will move from $(0,\ 0)$ to somewhere around $(0,\ 40cm)$ in the horizontal plane. The kinematic equations of motion in the x-y plane for the vehicle can be expressed as:

$$\begin{bmatrix} x_t \\ y_t \\ \psi_t \end{bmatrix} = \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \psi_{t-1} \end{bmatrix} + \begin{bmatrix} u_{t-1}\Delta t \cos(\psi_{t-1}) \\ u_{t-1}\Delta t \sin(\psi_{t-1}) \\ 0 \end{bmatrix} \tag{3.21}$$

where

x_t is the x position at time step t,
y_t is the y position at time step t,
ψ_t is the yaw angle at time step t,
u is the surge speed,
ψ is the yaw rate,
Δt is the sampling period

Suppose the system states are described with a state variable sv = [x y ψ]$^{\mathrm{T}}$, and the covariance for the initial state is

$$\mathbf{P}_{\mathrm{sv}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix} \tag{3.22}$$

The Monte Carlo method is used as the truth data. 500 samples starting from the initial position are drawn with covariance of P_{sv} and propagated through Equation 3.21. The effect of the large error variance on the nonlinearly transformed estimates is shown in Figure 3.3. Problems arise when the heading error is significant as in this example. The speed control can be made very accurate while the bearing measurement can be very poor (standard deviation of 5 degrees). As we can see, the points lie on a "banana-shaped" arc. The range error causes the points to lie in a band, and the heading error causes this region to stretch around the circumference of a circle. As a result, the mean does not lie at (0, 40 cm) but is actually located closer to the origin.

In Figure 3.4, the transformed means and covariances using Monte Carlo sampling, linearization and UT are illustrated respectively. The figure plots the $1-\sigma$ contours calculated by each method. Compared to the "true" result, the linearized estimate is biased and inconsistent. The linearized mean is at (0, 40cm) but the true mean is at (0, 38.2cm). The covariance of the linearized method is underestimated to the true covariance. On the other hand, we can see that the UT transformation captures the mean and covariance more accurately. This property of UT makes it a very promising method for estimating states of Gaussian nonlinear systems.

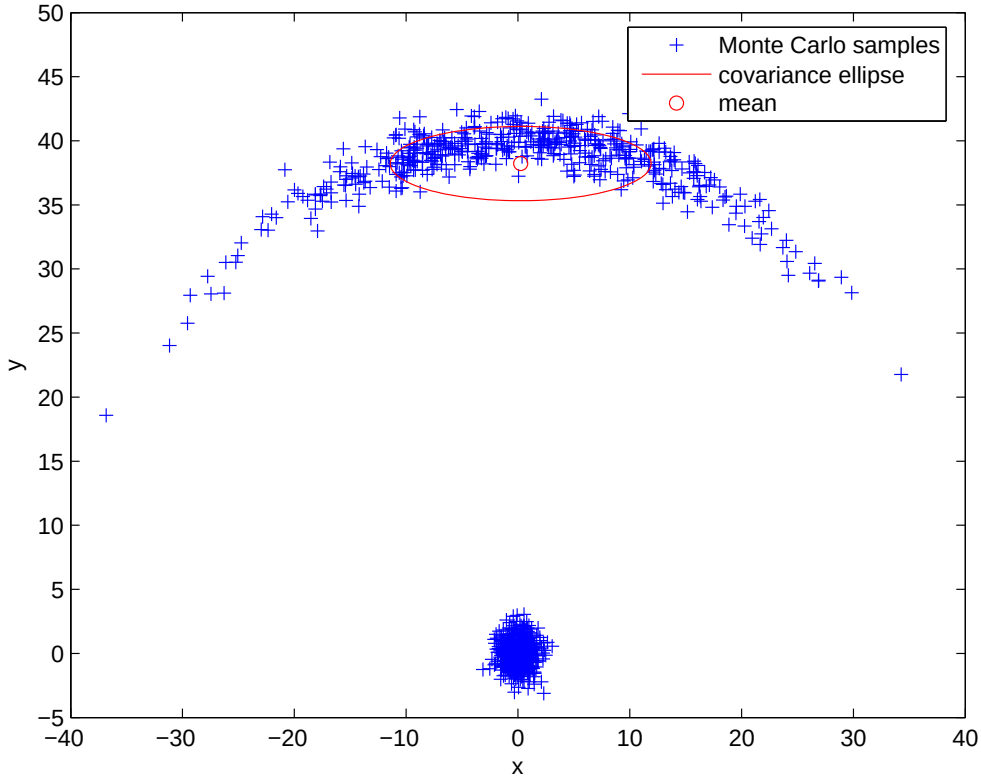


Figure 3.3: The true nonlinear transformation

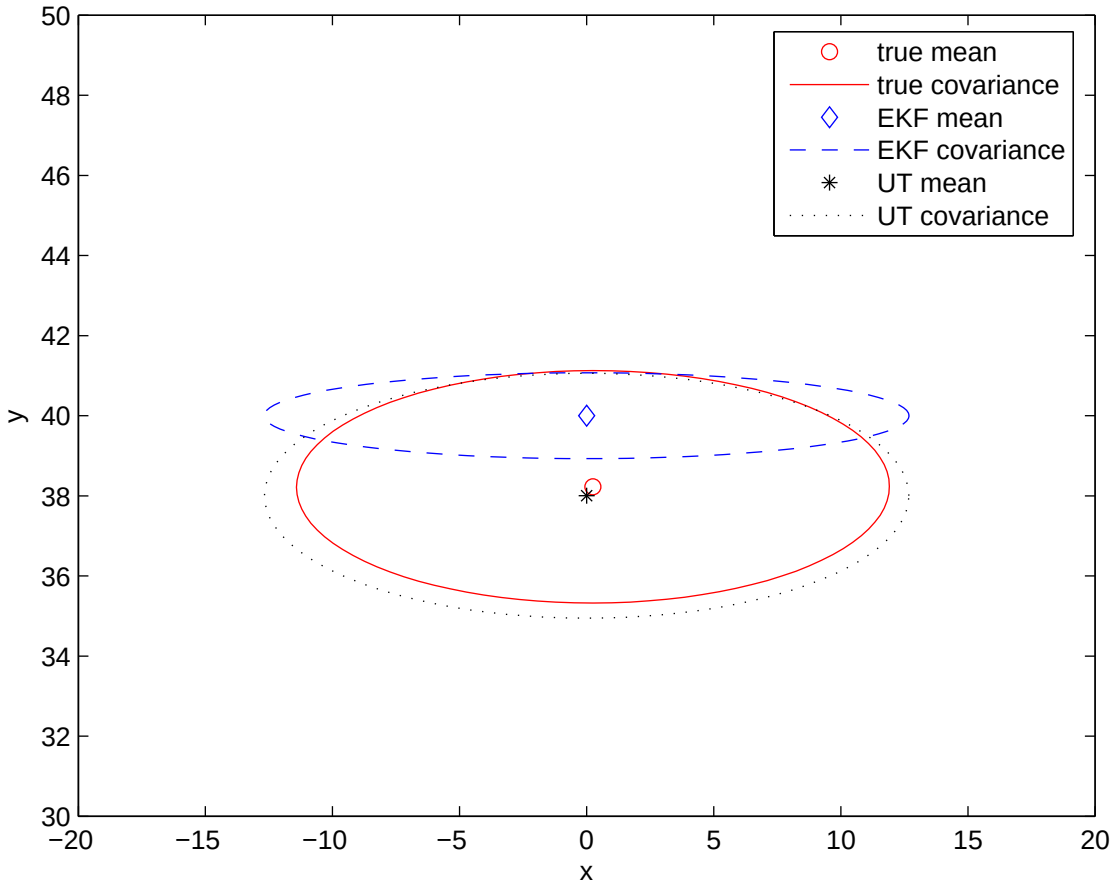


Figure 3.4: UT vs. EKF

3.5 The Unscented Kalman Filter

The unscented Kalman filter (UKF) is a straightforward application of the scaled unscented transformation to recursive minimum mean- squared-error estimation [16]. The state random variable is redefined as the concatenation of the original state variables $\mathbf{x}_t$ and noise variables $\mathbf{m}_t$ and $\mathbf{n}_t$, which is called the augmented state variable:

$$\mathbf{x}_t^a = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{m}_t \\ \mathbf{n}_t \end{bmatrix} \tag{3.23}$$

The covariances associated with $\mathbf{x}_t$, $\mathbf{m}_t$ and $\mathbf{n}_t$ are $\mathbf{P}_t$, $\mathbf{Q}$, and $\mathbf{R}$, respectively. Here, we assume that the $\mathbf{Q}$ and $\mathbf{R}$ are constant. The UKF algorithm will use the augmented state variable to predict the system state with contron input and correct the predicted states using the measurements.

The complete UKF algorithm is given as following:

1. Initialize with:

$$\bar{\mathbf{x}}_0 = \mathrm{E}[\mathbf{x}_0] \tag{3.24}$$

$$\mathbf{P}_0 = \mathrm{E}[(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^{\mathrm{T}}] \tag{3.25}$$

$$\bar{\mathbf{x}}_0^a = \mathrm{E}[\mathbf{x}^{\mathrm{a}}] = \begin{bmatrix} \bar{\mathbf{x}}_0 \\ 0 \\ 0 \end{bmatrix} \tag{3.26}$$

$$\mathbf{P}_0^a = \mathrm{E}[(\mathbf{x}_0^{\mathrm{a}} - \bar{\mathbf{x}}_0^{\mathrm{a}})(\mathbf{x}_0^{\mathrm{a}} - \bar{\mathbf{x}}_0^{\mathrm{a}})^{\mathrm{T}}] = \begin{bmatrix} \mathbf{P}_0 & 0 & 0 \\ 0 & \mathbf{Q} & 0 \\ 0 & 0 & \mathbf{R} \end{bmatrix} \tag{3.27}$$

2. For $t \in \{1, \ldots, \infty\}$,

   (a) Calculate sigma points:

$$\mathcal{X}_{t-1}^a = \begin{bmatrix} \bar{\mathbf{x}}_{t-1}^a & \bar{\mathbf{x}}_{t-1}^a + \sqrt{(n+\lambda)\mathbf{P}_{t-1}^a} & \bar{\mathbf{x}}_{t-1}^a - \sqrt{(n+\lambda)\mathbf{P}_{t-1}^a} \end{bmatrix} \tag{3.28}$$

(b) Prediction step:

$$\mathcal{X}^x_{t|t-1} = \mathbf{f}(\mathcal{X}^x_{t-1}, \mathbf{u}_{-t-1}, \mathcal{X}^v_{t-1}) \tag{3.29}$$

$$\bar{\mathbf{x}}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}^x_{i,t|t-1} \tag{3.30}$$

$$\mathbf{P}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}^x_{i,t|t-1} - \bar{\mathbf{x}}_{t|t-1}][\mathcal{X}^x_{i,t|t-1} - \bar{\mathbf{x}}_{t|t-1}]^T \tag{3.31}$$

$$\mathcal{Y}_{t|t-1} = \mathbf{h}(\mathcal{X}^x_{t|t-1}, \mathcal{X}^n_{t|t-1}) \tag{3.32}$$

$$\bar{\mathbf{y}}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,t|t-1} \tag{3.33}$$

(c) Correction step:

$$\mathbf{P}_{\tilde{\mathbf{y}}_t \tilde{\mathbf{y}}_t} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{Y}_{i,t|t-1} - \bar{\mathbf{y}}_{t|t-1}][\mathcal{Y}_{i,t|t-1} - \bar{\mathbf{y}}_{t|t-1}]^T \tag{3.34}$$

$$\mathbf{P}_{\mathbf{x}_t \mathbf{y}_t} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,t|t-1} - \bar{\mathbf{x}}_{t|t-1}][\mathcal{Y}_{i,t|t-1} - \bar{\mathbf{y}}_{t|t-1}]^T \tag{3.35}$$

$$\mathbf{K}_t = \mathbf{P}_{\mathbf{x}_t \mathbf{y}_t} \mathbf{P}^{-1}_{\tilde{\mathbf{y}}_t \tilde{\mathbf{y}}_t} \tag{3.36}$$

$$\bar{\mathbf{x}}_t = \bar{\mathbf{x}}_{t|t-1} + \mathbf{K}_t(\mathbf{y}_t - \bar{\mathbf{y}}_{t|t-1}) \tag{3.37}$$

$$\mathbf{P}_t = \mathbf{P}_{t|t-1} - \mathbf{K}_t \mathbf{P}_{\tilde{\mathbf{y}}_t \tilde{\mathbf{y}}_t} \mathbf{K}_t^T \tag{3.38}$$

where:

$\mathbf{x}^a = \begin{bmatrix} \mathbf{x} \\ \mathbf{v} \\ \mathbf{n} \end{bmatrix}$ is the augmented state variable with dimension n_a;

$\mathcal{X}^a = \begin{bmatrix} \mathcal{X}^x \\ \mathcal{X}^v \\ \mathcal{X}^n \end{bmatrix}$ is the augmented sigma points,

λ = composite scaling parameter,

$n_a = n_x + n_m + n_n$ is the dimension of the augmented state variable,

$\mathbf{Q}$ = process noise covariance,

$$\begin{aligned}\mathbf{R} &= \text{measurement noise covariance,}\\ \mathbf{K} &= \text{Kalman gain,}\\ W_i &= \text{weights for sigma points}\end{aligned}$$

Note that the unscented Kalman filter has the same recursive "predict-correct" structure as the Kalman filter. However, there is no requirement to calculate the Jacobians in this algorithm. Moreover, the computation complexity in the UKF is as the same as in the EKF. So, not only does the UKF outperform the EKF in accuracy and robustness, it does so at no extra computational cost. The superior performance of the UKF over that of the EKF have been reported in numerous publications [15, 16, 35, 36].

The algorithm discussed above is a general form of the unscented Kalman filter. There are many practical cases where the process and measurement noise are purely additive or can be approximately considered additive as expressed as

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}) + \mathbf{m}_{t-1} \tag{3.39}$$

$$\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{n}_t. \tag{3.40}$$

For this system, the computational complexity of the UKF can be reduced.

When the process and measurement noise are additive, the system state need not be augmented with the noise random variables. The dimension of the sigma points as well as the total number of sigma points can be reduced. The covariances of the noise sources are then incorporated in the state covariance using a simple additive procedure [33], which is given as follows:

1. Initialize with:

$$\bar{\mathbf{x}}_0 = \mathrm{E}[\mathbf{x}_0] \tag{3.41}$$

$$\mathbf{P}_0 = \mathrm{E}[(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^{\mathrm{T}}] \tag{3.42}$$

2. For $t \in \{1, \ldots, \infty\}$,

   (a) Calculate sigma points:

$$\mathcal{X}_{t-1} = \left[\bar{\mathbf{x}}_{t-1} \quad \bar{\mathbf{x}}_{t-1} + \sqrt{(n+\lambda)\mathbf{P}_{t-1}} \quad \bar{\mathbf{x}}_{t-1} - \sqrt{(n+\lambda)\mathbf{P}_{t-1}}\right] \tag{3.43}$$

(b) Predict step:

$$\mathcal{X}_{t|t-1} = \mathbf{f}(\mathcal{X}_{t-1}, \mathbf{u}_{t-1}) \tag{3.44}$$

$$\bar{\mathbf{x}}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}_{i,t|t-1} \tag{3.45}$$

$$\mathbf{P}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,t|t-1} - \bar{\mathbf{x}}_{t|t-1}][\mathcal{X}_{i,t|t-1} - \bar{\mathbf{x}}_{t|t-1}]^T + \mathbf{Q} \tag{3.46}$$

$$\mathcal{Y}_{t|t-1} = \mathbf{h}(\mathcal{X}_{t|t-1}, \mathbf{u}_{t|t-1}) \tag{3.47}$$

$$\bar{\mathbf{y}}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,t|t-1} \tag{3.48}$$

(c) Correct step:

$$\mathbf{P}_{\tilde{\mathbf{y}}_t \tilde{\mathbf{y}}_t} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{Y}_{i,t|t-1} - \bar{\mathbf{y}}_{t|t-1}][\mathcal{Y}_{i,t|t-1} - \bar{\mathbf{y}}_{t|t-1}]^T + \mathbf{R} \tag{3.49}$$

$$\mathbf{P}_{\mathbf{x}_t \mathbf{y}_t} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,t|t-1} - \bar{\mathbf{x}}_{t|t-1}][\mathcal{Y}_{i,t|t-1} - \bar{\mathbf{y}}_{t|t-1}]^T \tag{3.50}$$

$$\mathbf{K}_t = \mathbf{P}_{\mathbf{x}_t \mathbf{y}_t} \mathbf{P}^{-1}_{\tilde{\mathbf{y}}_t \tilde{\mathbf{y}}_t} \tag{3.51}$$

$$\bar{\mathbf{x}}_t = \bar{\mathbf{x}}_{t|t-1} + \mathbf{K}_t(\mathbf{y}_t - \bar{\mathbf{y}}_{t|t-1}) \tag{3.52}$$

$$\mathbf{P}_t = \mathbf{P}_{t|t-1} - \mathbf{K}_t \mathbf{P}_{\tilde{\mathbf{y}}_t \tilde{\mathbf{y}}_t} \mathbf{K}_t^T \tag{3.53}$$

where:

$\mathbf{x}$ is the state variable;
$\mathcal{X}$ is the sigma points,
λ = composite scaling parameter,
$\mathbf{Q}$ = process noise covariance,
$\mathbf{R}$ = measurement noise covariance,
$\mathbf{K}$ = Kalman gain,
W_i = weights for sigma points

In this work, we implement the UKF using the additive noise approach, because the process model and measurement model for the VideoRay Pro III are considered to be

noise additive. When calculating the square root for the state covariance matrix, we use Cholesky factorization, which is in general of order $n^3/6$.

Chapter 4

Controller Design

4.1 Introduction

Controllers can be classified according to whether they are applied to linear systems or nonlinear systems.

Linear controllers have been applied in the autopilot system for marine vehicles for several decades. Many researchers have successfully applied linear controllers, such as PID controllers, in forward speed control, course-keeping control, turning rate control for marine vehicles [9].

However, linear controllers have two major drawbacks when applied in the autonomous control for underwater vehicles. First, the linear controllers rely on the key assumption of small range operation for the linearization model to be valid. This assumption is not always true for underwater vehicles because their operation range is usually large, in which case a linear controller is likely to perform very poorly or be unstable. Second, in designing linear controllers, it is usually necessary to assume that the parameters of the system model are reasonably well known. On the contrary, in underwater vehicle application, the uncertainties in the model parameters and unpredictable disturbances have a large impact on the vehicle modeling accuracy. A linear controller based on inaccurate model parameters may exhibit significant performance degradation or even instability.

These two issues can be solved with the use of nonlinear control tools. Nonlinear controllers can handle the nonlinearities across a larger range of operation and overcome

the impact of model inaccuracy. Since the VideoRay Pro III ROV is usually operated in various working conditions, we need a robust nonlinear controller to work over the whole operating range so as to achieve the control objective for the autopilot system.

4.1.1 Control Objective

The objective of autopilot design for the VideoRay Pro III ROV is to control the vehicle such that it can fulfill its mission as listed in the following:

- following a feasible trajectory which is defined by a series of waypoints, consisting of x, y and z coordinates, along the trajectory;
- maintaining a required depth and heading while the vehicle is on its mission.

Because the VideoRay Pro III ROV is actuated by two parallel thrusters for its surge and yaw control, and a vertical thruster for the depth control, the above objectives sometimes conflict with each other. For example, when the ROV is following a series of waypoints, it is not always possible to keep the heading in a certain orientation. And, from the mission point of view, it is also not necessary to fulfill the two objectives simultaneously.

Therefore, we want to develop an autopilot system so as to control the VideoRay Pro III to work in two modes:

1. 3D trajectory tracking of x, y and z coordinates. In this mode, we will develop a planar trajectory tracking controller for the VideoRay Pro III to track a path in the horizontal x-y plane; In the vertical plane, the z coordinate tracking will be achieved with a depth controller.
2. Bearing control mode. We will develop a bearing controller to control VideoRay Pro III's bearing.

By combining these two control modes in a mission, the VideoRay Pro III can carry out its objective using the autopilot system. For example, the vehicle may run in the 3D trajectory tracking mode to follow a predefined trajectory. Once it gets to the desired position, the controller may change to the bearing control mode to adjust its attitude for tracking target.

4.1.2 Trajectory Tracking Systems

Trajectory tracking problems are concerned with the design of control laws that force a vehicle to reach and follow a time parameterized reference.

Controlling the VideoRay Pro III to track a trajectory is challenging because of its actuator configuration. The control actuators available for the VideoRay Pro III are the two thrusters on the port and starboard sides for surge and yaw control, and a vertical thruster for the depth control. See Figure 2.1. Note that there is no thruster for the sway control. When moving in the horizontal plane, the VideoRay Pro III has no control in the sway direction. Therefore, it falls into the category of underactuated system, which is defined to be one where the dimension of the space spanned by the control vector is less than the dimension of the configuration space. Simply stated, this refers to a system that has fewer control inputs than degrees of freedom.

In the past few decades, many efforts have been made in the area of trajectory tracking control for underactuated vehicles. Among such methods, the guidance, navigation and control (GNC) has been widely used in the aircraft industry, missile guidance and marine ship industry. More recently, a new approach that is based on model-based-direct-Lyapunov method has obtained attention from reserachers.

The Guidance, Navigation and Control (GNC) Approach

The classical approach for implementing autopilot control for underactuated systems like airplanes or ships uses the so called guidance navigation control (GNC) method.

The classical GNC autopilot control of airplanes or ships involves controlling the vehicle's yaw angle ψ. The system is usually designed such that the vehicle can move forward with constant speed u at the same time as the yaw angle ψ is controlled. Hence, the vehicle can be made to track a predifined reference path, which is easily specified by way points.

A typical guidance, navigation and control system is depicted in Figure 4.1. In addition to the navigation system, autopilot control of this kind usually consists of a control system and a guidance system. The guidance system usually utilizes techniques such as [9]:

- way point guidance based on the straight line between two points;
- way point guidance by line of sight

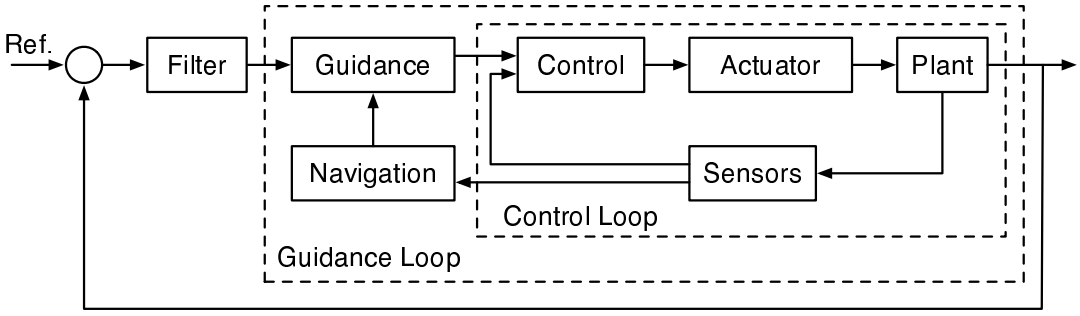


Figure 4.1: A typical guidance, navigation, and control system

Many GNC systems have been implemented in controling marine vehicles to track a predefined path. In [26], Fotis A. Papoulias presented a theoretical analysis of the nonlinear dynamic phenomena involved in pure pursuit guidance of marine vehicles. The navigation loop processes positional information, determines the actual geographical location of the vehicle and compares it to the commanded path. This loop has its own dynamics but it operates at a rather slow rate. The guidance loop accepts navigational information and generates appropriate commands which in turn become the input to the vehicle control systems, and these determine the necessary vehicle actuator signals. For accurate path keeping, the guidance and autopilot functions have similar dynamic response characteristics and this may create stability problems once the two systems are coupled together.

In [24], W. Naeem, *et al.* investigated a novel approach to an underwater vehicle cable tracking mission by employing an integrated guidance and control system using a proportional navigation guidance (PNG) law and model predictive control (MPC), which was originally used for missile systems.

However, GNC systems exhibit some drawbacks. Strictly speaking, GNC systems are just path following systems rather than trajectory tracking systems. This is because GNC systems follow a path that is not defined as a function of time. The way points that define the path do not have an associated time requirement for the vehicle to reach the points. On the other hand, the goal that we want to accomplish for the autonomous control of the VideoRay Pro III is the ability to drive the vehicle to track a pre-defined trajectory, which is defined by a series of way points associated with time.

Another drawback for the GNC systems is loss of stability, which is possible if the control law is not sufficiently responsive compared to the dynamics of the guidance law. Explicit stability conditions could be derived with extensive use of bifurcation theory, center manifold reduction and integral averaging. A great deal of effort is required to achieve stability for GNC systems.

Model-Based Direct-Lyapunov Stability Approach

The classical approach for trajectory tracking of underactuated marine vehicles utilizes local linearization and coordinate decoupling to steer the same number of degrees of freedom as the number of available control inputs [9]. Alternative approaches include the linearization of the vehicle error dynamics about trajectories combined with gain scheduling [19]. The basic limitation of these approaches is that the stability is only guaranteed in a neighborhood of the selected operating points. Moreover, performance can suffer significantly when the vehicle executes maneuvers that emphasize its nonlinearity and cross-couplings.

Another method is to consider the kinematics of the vehicle and design a time-varying control law for the surge and yaw inputs [27] to follow a feasible trajectory defined with line segments and circle arcs. Under this controller, the errors in position and orientation with respect to the reference trajectory is practically globally exponentially stabilized to zero.

These approaches suffer from the drawback that the vehicle's dynamics usually exhibit complex nonlinear terms with significant uncertainty, making the task of computing a feasible trajectory hard. Fortunately, only the need to track the position rather than the state-space trajectory is desired in practical applications. Motivated by this considerations, in [3] Aguiar and Hespanha proposed a model-based direct-Lyapunov stability method to the position tracking problem for a fairly general class of underactuated autonomous vehicles that is applicable to motion in either two or three dimensional spaces. The control algorithm proposed builds on iterative Lyapunov-based techniques and it was shown to yield global stability and exponential convergence of the position tracking error to a neighborhood of the origin that can be made arbitrarily small. Furthermore, the desired trajectory does not need to be specially chosen. In fact, it can be any sufficiently smooth bounded curve parameterized by time. In this method, an integrator backstepping

technique is used to develop the control law to perform position tracking.

Since our goal is for the VideoRay Pro III to track a position trajectory without tracking a desired bearing, this model-based direct-Lyapunov stability approach is applicable. We will develop a trajectory tracking algorithm based on this approach, and the resulting closed loop system can be proven Lyapunov stable.

4.1.3 Bearing Control

Sliding mode control has been applied successfully in the control of underwater vehicles with a series of single-input single-output controllers [39, 40, 38]. All these experiments show that sliding mode controllers have significant advantages to traditional linear controllers. The main advantange for sliding mode control is that it is robust when there exists much uncertainty for the model parameters and the disturbance from environment is relatively high.

We will apply sliding mode control for the bearing and depth control of the VideoRay Pro III.

4.2 Planar Trajectory Tracking Control Design

In this section we will develop a control law to allow the VideoRay Pro III to track a horizontal planar position trajectory $\Phi(t)$ defined by

$$\Phi(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}.$$

The key step in the approach is to apply the backstepping technique to track only two position variables instead of the entire three dimensional configuration [x, y, ψ] in the horizontal plane, based on the work by Aguiar *et al.* in [2]. However, Aguiar *et al.* did not consider in their vehicle's model the quadratic damping terms, which are significant in our application. We developed the trajectory tracking controller with the quadratic drag terms considered in the model.

The next step is to select the surge force X and yaw torque N so that the control inputs appear one at a time as we iterate the backstepping procedure. The resulting control law

can be proven globally stable with the condition that the actuator never saturates.

We will start by describing the kinematic and dynamic equations for the VideoRay Pro III, followed by the formulation of the corresponding problem of planar trajectory tracking control. Finally, we will derive the solution to this problem by utilizing an integrator backstepping technique.

Vehicle Modeling

From Chapter 2, the general kinematic equations and dynamic equations of motion of the vehicle can be developed using an earth-fixed coordinate frame $\{U\}$ and a body-fixed coordinate frame $\{B\}$ that are depicted in Figure 4.2.

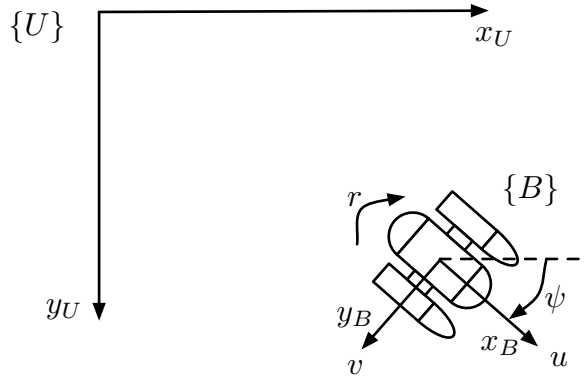


Figure 4.2: Body-fixed $\{B\}$ and earth-fixed $\{U\}$ coordinate frames

In the horizontal plane, the kinematic equations of motion for the vehicle can be reduced as:

$$\dot{x} = u\cos\psi - v\sin\psi, \tag{4.1}$$

$$\dot{y} = u\sin\psi + v\cos\psi, \tag{4.2}$$

$$\dot{\psi} = r \tag{4.3}$$

where:

x = Cartesian coordinate of vehicle's center of mass, expressed in the $\{U\}$ frame;

y = Cartesian coordiante of vehicle's center of mass, expressed in the $\{U\}$ frame;
u = surge speeed in the $\{B\}$ frame;
v = sway speed in the $\{B\}$ frame;
ψ = vehicle's orientation, expressed in the $\{U\}$ frame;
r = $\dot{\psi}$ vehicle's angular speed.

Therefore, the kinematic equations and the dynamic equations 2.23, 2.24 and 2.25 can also be written in matrix format as:

$$\dot{p} = R(\psi)\nu, \tag{4.4}$$
$$\dot{\psi} = r, \tag{4.5}$$
$$M\dot{\nu} = -S(r)M\nu + D_\nu(\nu)\nu + gX, \tag{4.6}$$
$$J\dot{r} = d_r(r)r + N. \tag{4.7}$$

where

$p = \begin{bmatrix} x \\ y \end{bmatrix}$ is the position vector, expressed in the $\{U\}$ frame;

$\nu = \begin{bmatrix} u \\ v \end{bmatrix}$ is the velocity vector, expressed in the $\{B\}$ frame;

$R(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix}$ is the rotation matrix;

$S(r) = \begin{bmatrix} 0 & -r \\ r & 0 \end{bmatrix}$ is a skew-symmetric matrix;

$M = \begin{bmatrix} m_{11} & 0 \\ 0 & m_{22} \end{bmatrix}$ is the mass matrix;

$D_\nu(\nu) = \begin{bmatrix} X_u + X_{u|u|}|u| & 0 \\ 0 & Y_v + Y_{v|v|}|v| \end{bmatrix}$ is the damping coefficient matrix, which is negative definite and time varying;

$d_r(r) = N_r + N_{r|r|}|r|$, which is negative definite and time varying;

$g = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

From the equations of motion above, we see that the VideoRay Pro III is underactuated because there is no available control in the sway direction.

Problem Formulation

Given the kinematic and dynamic equations of motion, the problem of trajectory tracking for the VideoRay Pro III can be formulated as following:

Suppose that $p_d : [0, \infty) \rightarrow \mathcal{R}^2$ be a given sufficiently smooth time-varying desired trajectory with its first three derivatives bounded. We are going to design a controller such that all the closed-loop signals are globally bounded and the tracking error $||p - p_d||$ converges exponentially to a neighborhood of the origin that can be made arbitrarily small.

4.2.1 Backstepping Technique

In this section, we are going to design a control law for the VideoRay Pro III to track trajectories. The integrator backstepping technique will be used.

Step 1: Coordinate Transformation

The tracking error e can be expressed in the body-fixed frame $\{B\}$, using the coordinate transformation from earth-fixed coordinate frame $\{U\}$, as:

$$e = R(\psi)^T(p - p_d), \tag{4.8}$$

and since $\dot{R}(\psi) = S(r)R(\psi)$, $\dot{R}^T(\psi) = -S(r)R^T(\psi)$ and $\nu = R^T(\psi)\dot{p}$, the dynamics $\dot{e}$ can be expressed as

$$\dot{e} = -S(r)e - R(\psi)^T\dot{p}_d + \nu. \tag{4.9}$$

where

$p = \begin{bmatrix} x \\ y \end{bmatrix}$ position of the vehicle;

$p_d = \begin{bmatrix} x_d \\ y_d \end{bmatrix}$ desired position of the vehicle;

$S(r)$ is the skew-symmetric matrix.

The block diagram of the error dynamics is depicted in Figure 4.3.

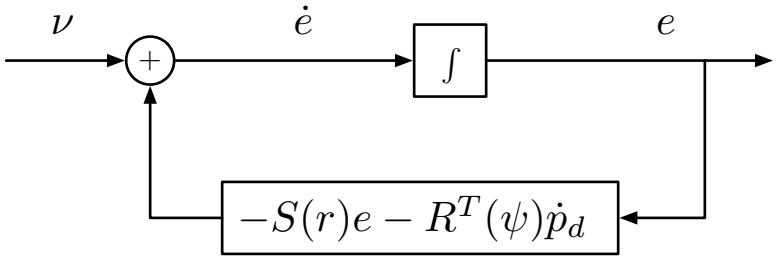


Figure 4.3: Block diagram for error dynamics $\dot{e}$

Step 2: Convergence of e

We define the Lyapunov function V_1 as:

$$V_1 = \frac{1}{2}e^T e \tag{4.10}$$

and since $e^T S(r)e = 0$, its time derivative is computed as:

$$\dot{V}_1 = e^T \dot{e} = e^T[\nu - R(\psi)^T \dot{p}_d] \tag{4.11}$$

We would like to use ν as a virtual control to make $\dot{V}_1$ negative, so that the tracking error e will converge to zero exponentially fast. This could be achieved if the virtual control input ν could be set equal to θ, which is defined as:

$$\theta := R(\psi)^T \dot{p}_d - K_e M^{-1} e \tag{4.12}$$

where K_e is a positive constant.

We define a new error variable z_1 as:

$$z_1 := \nu - \theta = \nu - R(\psi)^T \dot{p}_d + K_e M^{-1} e. \tag{4.13}$$

Hence,

$$\nu = R(\psi)^T \dot{p}_d - K_e M^{-1} e + z_1. \tag{4.14}$$

Rewrite the tracking error dynamics $\dot{e}$ as, using Equation 4.9 and 4.14:

$$\dot{e} = -S(r)e - K_e M^{-1} e + z_1. \tag{4.15}$$

We can now rewrite $\dot{V}_1$ as:

$$\dot{V}_1 = -K_e e^T M^{-1} e + e^T z_1 \tag{4.16}$$

We see that if we can drive z_1 to zero, we can guarantee that the tracking error e converge to zero exponentially fast.

By adding and subtracting θ, the error dynamics is now depicted in Figure 4.4.

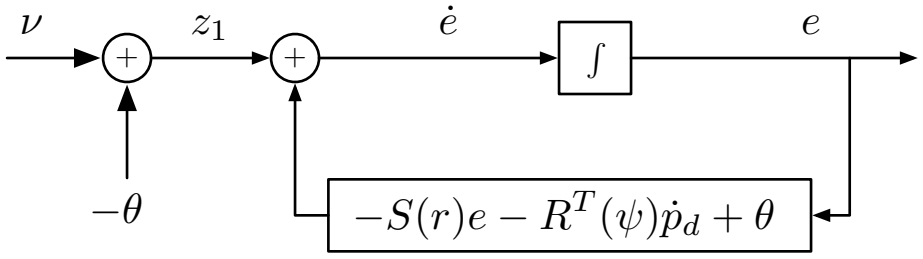


Figure 4.4: Introducing θ

Step 3: Backstepping for z_1

Before backstepping for z_1 term, the block diagram of z_1 and the vehicle's dynamics are depicted in Figure 4.5.

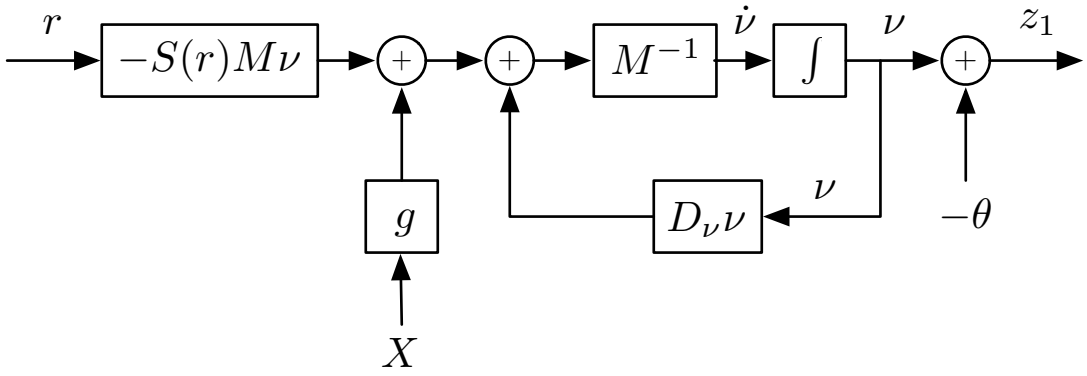


Figure 4.5: Before backstepping for z_1

By using equations 4.6, 4.12, 4.14, and 4.15, the z_1 dynamics can be written as:

$$\begin{aligned}
M\dot{z}_1 &= M\dot{\nu} - M\dot{\theta} \\
&= M\dot{\nu} - M[\dot{R}(\psi)^T\dot{p}_d + R(\psi)^T\ddot{p}_d - K_eM^{-1}\dot{e}] \\
&= M\dot{\nu} + MS(r)R(\psi)^T\dot{p}_d - MR(\psi)^T\ddot{p}_d + K_e\dot{e} \\
&= M\dot{\nu} + MS(r)R(\psi)^T\dot{p}_d - MR(\psi)^T\ddot{p}_d + K_e[-S(r)e - K_eM^{-1}e + z_1] \\
&= -S(r)M\nu + D_\nu\nu + gX \\
&\quad + MS(r)R(\psi)^T\dot{p}_d - MR(\psi)^T\ddot{p}_d + K_e[-S(r)e - K_eM^{-1}e + z_1] \\
&= [-S(r)M + D_\nu][R(\psi)^T\dot{p}_d - K_eM^{-1}e + z_1] + gX \\
&\quad + MS(r)R(\psi)^T\dot{p}_d - MR(\psi)^T\ddot{p}_d + K_e[-S(r)e - K_eM^{-1}e + z_1] \\
&= -S(r)Mz_1 + D_\nu z_1 + gX + h(e, \psi, z_1, \dot{p}_d, \ddot{p}_d) \qquad (4.17)
\end{aligned}$$

where

$$h = D_\nu R(\psi)^T\dot{p}_d - K_eD_\nu M^{-1}e - MR(\psi)^T\ddot{p}_d + K_ez_1 - K_e^2M^{-1}e$$

From equation 4.17, it turns out that it will not always be possible to drive z_1 to zero. Instead, we will drive z_1 to a small constant δ, which could be made arbitrarily small. To achieve this we define a new error variable φ,

$$\varphi := z_1 - \delta \qquad (4.18)$$

where

$$\delta = \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix}$$

Consider the augmented Lyapunov function V_2:

$$V_2 := V_1 + \frac{1}{2}\varphi^TM^TM\varphi. \qquad (4.19)$$

The time derivative of V_2 is written as:

$$\dot{V}_2 = -K_ee^TM^{-1}e + e^T\delta + \varphi^TM^TD_\nu\varphi + \varphi^T[M^TB(\delta)\mu + M^Th + M^TD_\nu\delta + e] \qquad (4.20)$$

where

$$B(\delta) = \begin{bmatrix} 1 & m_{22}\delta_2 \\ 0 & -m_{11}\delta_1 \end{bmatrix}$$
$$\mu = \begin{bmatrix} X \\ r \end{bmatrix}$$
D_ν is negative definite.

Here, μ can be regarded as a virtual control that we would like to use to make $\dot{V}_2$ negative. This goal could be achieved if we could set μ equal to α, which is defined as:

$$\alpha := -B(\delta)^{-1}[h(e, \psi, z_1, \dot{p}_d, \ddot{p}_d) + D_\nu \delta + M^{-1}e + K_\varphi M^{-1}\varphi] \tag{4.21}$$

where

$$K_\varphi = \begin{bmatrix} K_{\varphi 1} & 0 \\ 0 & K_{\varphi 2} \end{bmatrix}, K_{\varphi 1} > 0, K_{\varphi 2} > 0.$$

So far, in virtual control μ, we actually get the real control X. If we set X as:

$$X = \begin{bmatrix} 1 & 0 \end{bmatrix} \alpha \tag{4.22}$$

and r as:

$$r = \begin{bmatrix} 0 & 1 \end{bmatrix} \alpha, \tag{4.23}$$

we will be able to drive z_1 to the small constant δ.

Still, we have a virtual control r. Let's introduce the error variable z_2 as:

$$z_2 := r - \begin{bmatrix} 0 & 1 \end{bmatrix} \alpha. \tag{4.24}$$

We can rewrite $\dot{V}_2$ as:

$$\dot{V}_2 = -K_e e^T M^{-1} e + e^T \delta + \varphi^T M D_\nu \varphi - \varphi^T K_\varphi \varphi + \varphi^T M B_b(\delta) z_2 \tag{4.25}$$

where $B_b(\delta)$ is the second column of $B(\delta)$.

After backstepping for z_1, the system is depicted as in Figure 4.6, where we use real control input of

$$X = \begin{bmatrix} 1 & 0 \end{bmatrix} \alpha \tag{4.26}$$

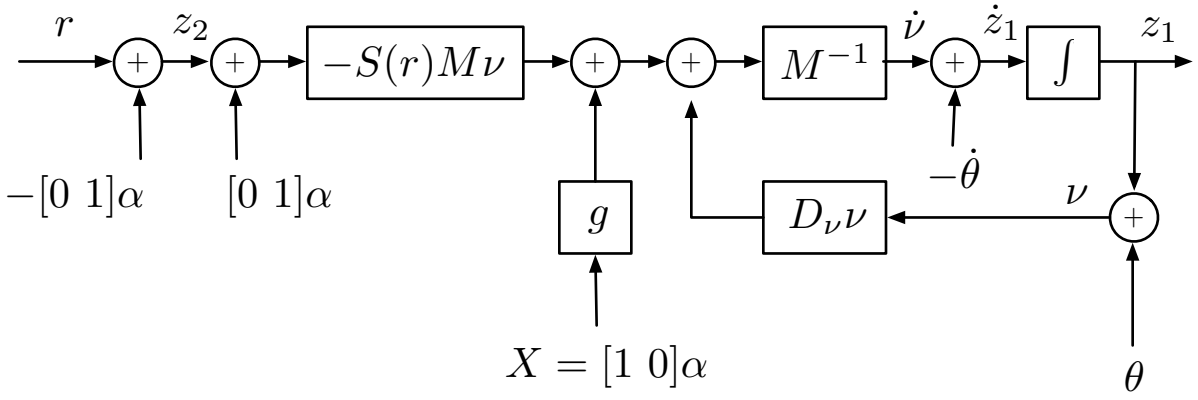


Figure 4.6: After backstepping for z_1

and virtual control input of

$$r = \begin{bmatrix} 0 & 1 \end{bmatrix} \alpha. \tag{4.27}$$

Examining $\dot{V}_2$, we see that although $\dot{V}_2$ is not necessarily always negative, it will be sufficient to achieve practical stability.

Step 4: Backstepping for z_2

Before backstepping for z_2, the dynamical system is depicted as in Figure 4.7.

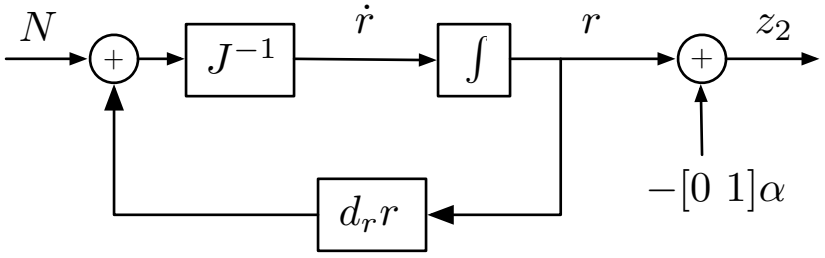


Figure 4.7: Before backstepping for z_2

Consider a third Lyapunov function V_3 as:

$$V_3 := V_2 + \frac{1}{2} J z_2^2 \tag{4.28}$$

The time derivative of V_3 is computed as:

$$\begin{aligned} \dot{V}_3 &= -K_e e^T M^{-1} e + e^T \delta + \varphi^T M D_\nu \varphi - \varphi^T K_\varphi \varphi + d_r z_2^2 \\ &\quad + z_2 (\varphi^T M B_b(\delta) + d_r \begin{bmatrix} 0 & 1 \end{bmatrix} \alpha - \begin{bmatrix} 0 & J \end{bmatrix} \dot{\alpha} + N) \end{aligned} \tag{4.29}$$

where D_ν is negative definite, $d_r < 0$.

If we then choose

$$N = -\varphi^T M B_b(\delta) - d_r \begin{bmatrix} 0 & 1 \end{bmatrix} \alpha + \begin{bmatrix} 0 & J \end{bmatrix} \dot{\alpha} - k_{z_2} z_2, \tag{4.30}$$

the time derivative of V_3 becomes

$$\dot{V}_3 = -K_e e^T M^{-1} e + e^T \delta - \varphi^T M D_\nu \varphi - \varphi^T K_\varphi \varphi + d_r z_2^2 - k_{z_2} z_2^2 \tag{4.31}$$

The z_2 dynamics after backstepping is depicted in Figure 4.8.

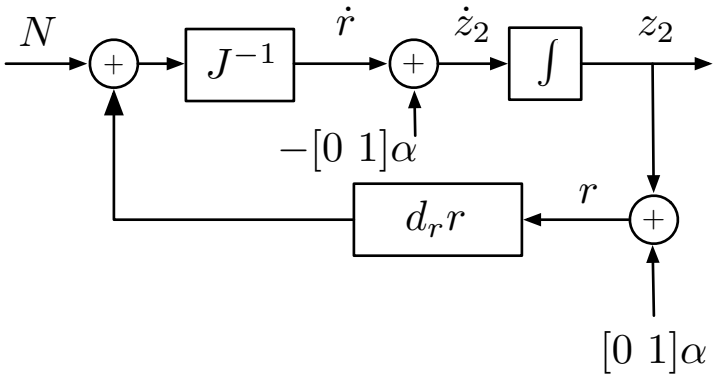


Figure 4.8: Before backstepping for z_2

Again, note that although $\dot{V}_3$ is not necessarily always negative, this will be sufficient to achieve practical stability.

Up to now, we have developed the control law for surge force X and yaw torque N as in Equation 4.26 and Equation 4.30. By selecting values for the control gains of K_e, K_φ and k_{z_2} we can adjust the performance of this trajectory tracking controller. Parameters δ_1 and δ_2 can be used to adjust the controller accuracy. Although theoretically they can be made arbitrarily small, it is usually set to a value that will not cause the controller to be overly sensitive to system noise.

4.2.2 Stability of the PTT Controller

It can be proved, in [3] that given a three-times continuously differentiable time-varying desired trajectory $p_d : [0, \infty) \rightarrow \mathbf{R}^2$ with its first three derivatives bounded, consider the closed-loop system $\sum$ consisting of the underactuated vehicle model and feedback controller,

1. for any initial condition the solution to $\sum$ exists globally, all closed-looop signals are bounded, and the tracking error $||p(t) - p_d(t)||$ satisfies

$$||p(t) - p_d(t)|| \leq e^{-\lambda t} c_0 + \varepsilon, \tag{4.32}$$

   where λ, c_0, ε are positive constants. From these, only c_0 depends on initial conditions.

2. By appropriate choice of the controller parameters k_e, K_φ, k_{z2}, any desired values for ε and λ in Equation 4.32 can be obtained.

4.3 Bearing and Depth Control

In this section, we will derive the sliding mode controller for bearing and depth control for VideoRay Pro III.

By neglecting all kinematic and dynamic cross-coupling terms and using substitution of $\dot{z} = w$ and $\dot{\psi} = r$ for Equation 2.25 and Equation 2.26, we obtain a general 2nd-order system model for the heave motion and yaw motion:

$$m_{33}\ddot{z} + Z_w\dot{z} + Z_{w|w|}\dot{z}|\dot{z}| = Z \tag{4.33}$$

$$m_{66}\ddot{\psi} + N_r\dot{\psi} + N_{r|r|}\dot{\psi}|\dot{\psi}| = N \tag{4.34}$$

Note that the heave and yaw motion have the equation of motion of the same structure.

We will use $m\ddot{\eta} + d_1\dot{\eta} + d_2\dot{\eta}\,|\dot{\eta}| = \tau$ as a general form for the heave and yaw motion to derive the sliding mode control law.

4.3.1 SMC for Bearing and Depth Control

Consider the second-order system:

$$m\ddot{\eta} + d_1\dot{\eta} + d_2\dot{\eta}\,|\dot{\eta}| = \tau \tag{4.35}$$

where m, d_1 and d_2 are system parameters that are not well known.

Tracking Error and Sliding Surface

The tracking error is defined as:

$$e = \eta - \eta_d \tag{4.36}$$

where η_d is the desired position.

We define a scalar measure of tracking error

$$s = \dot{e} + \lambda e \tag{4.37}$$

where e is the tracking error, and $\lambda > 0$ is the control bandwidth.

Suppose we can design a control law that constrains the motion of the system to the manifold (or surface)

$$s = \dot{e} + \lambda e = 0, \tag{4.38}$$

This surface is called sliding surface.

On this surface, the motion is governed by:

$$\dot{e} = -\lambda e. \tag{4.39}$$

The solution to this equation is

$$e(t) = \exp(-\lambda(t - t_0))e(t_0), \tag{4.40}$$

Choosing $\lambda > 0$ guarantees that η tends to zero as t tends to infinity and the rate of convergence can be controlled by choice of λ. Actually, the error trajectory will reach the time-varying sliding surface in finite time for any initial condition $e(t_0)$ and then slide along the sliding surface towards $e(t) = 0$ exponentially. The motion of the system on the sliding surface $s = 0$ is independent of the original system. The sliding surface is depicted in Figure 4.9. Once the system gets on the sliding surface, it cannot leave it.

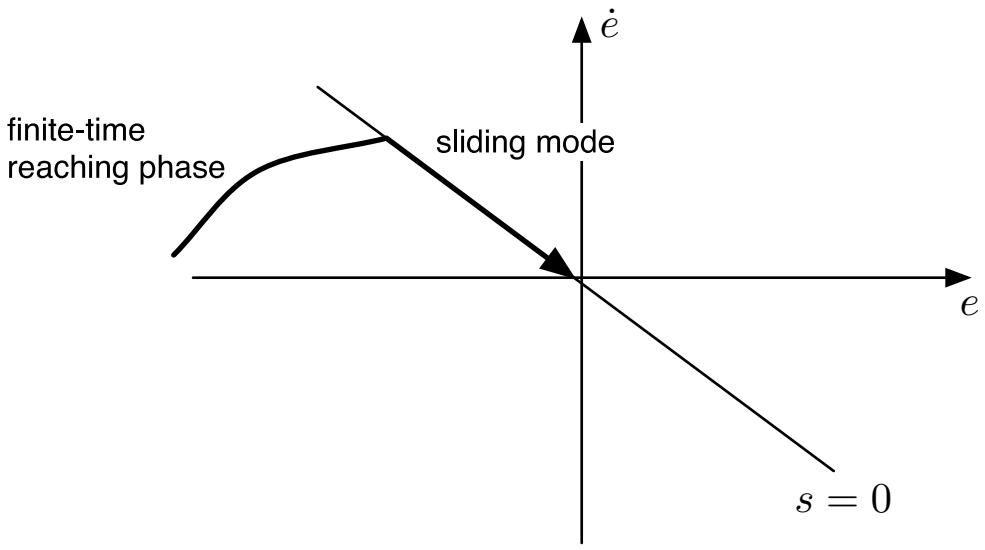


Figure 4.9: Graphical interpretation of sliding surface

Hence, the control objective is reduced to finding a nonlinear control law to ensure that:

$$\lim_{t \to \infty} s(t) = 0. \tag{4.41}$$

Control Law

In finding out the control law, the direct Lyapunov method is used.

Consider the dynamics of s:

$$\begin{aligned} m\dot{s} &= m\ddot{e} + m\lambda\dot{e} \\ &= m\ddot{\eta} - m\ddot{\eta}_d + m\lambda\dot{e} \\ &= \tau - d_1\dot{\eta} - d_2\dot{\eta}|\dot{\eta}| - m\ddot{\eta}_d + m\lambda\dot{e} \\ &= -d_2|\dot{\eta}|(\dot{\eta} - \dot{\eta}_d + \lambda e) + [\tau - m(\ddot{\eta}_d - \lambda\dot{e}) - d_1\dot{\eta} - d_2|\dot{\eta}|(\dot{\eta}_d - \lambda e)] \\ &= -d_2|\dot{\eta}|(\dot{e} + \lambda e) + [\tau - m(\ddot{\eta}_d - \lambda\dot{e}) - d_1\dot{\eta} - d_2|\dot{\eta}|(\dot{\eta}_d - \lambda e)] \\ &= -d_2|\dot{\eta}|s + [\tau - m(\ddot{\eta}_d - \lambda\dot{e}) - d_1\dot{\eta} - d_2|\dot{\eta}|(\dot{\eta}_d - \lambda e)] \end{aligned} \tag{4.42}$$

Define a virtual reference η_r satisfying:

$$\dot{\eta}_r = \dot{\eta}_d - \lambda e \tag{4.43}$$

and plug it into Equation 4.42, we get:

$$m\dot{s} = -d_2|\dot{\eta}|s + [\tau - m\ddot{\eta}_r - d_1\dot{\eta} - d_2|\dot{\eta}|\eta_r] \tag{4.44}$$

Consider the Lyapunov function candidate:

$$V = \frac{1}{2}ms^2 \tag{4.45}$$

Differentiating V with respect to time yields

$$\begin{aligned} \dot{V} &= m\dot{s}s \\ &= -d_2|\dot{\eta}|s^2 + s\left[\tau - m\ddot{\eta}_r - d_1\dot{\eta} - d_2|\dot{\eta}|\dot{\eta}_r\right] \end{aligned} \tag{4.46}$$

In order to make $\dot{V}$ negative, we take the control law to be

$$\tau = \hat{m}\ddot{\eta}_r + \hat{d}_1\dot{\eta} + \hat{d}_2|\dot{\eta}|\dot{\eta}_r - K_d s - K\mathrm{sgn(s)} \tag{4.47}$$

where

$\hat{m}$ = the estimates of m,
$\hat{d}_1$ = the estimates of d_1,
$\hat{d}_2$ = the estimates of d_2,

$$
\begin{aligned}
K_d &= \text{damping coefficient, } K_d > 0, \\
K &= \text{switching gain,} \\
\text{sgn}(s) &= \begin{cases} 1 & \text{if } s > 0 \\ 0 & \text{if } s = 0 \\ -1 & \text{otherwise} \end{cases}
\end{aligned}
$$

This control law yields

$$\dot{V} = -(K_d + d_2|\dot{\eta}|)s^2 + (\tilde{m}\ddot{\eta}_r + \tilde{d}_1\dot{\eta} + \tilde{d}_2|\dot{\eta}|\dot{\eta}_r)s - K|s| \tag{4.48}$$

where

$$
\begin{aligned}
\tilde{m} &= \hat{m} - m, \\
\tilde{d}_1 &= \hat{d}_1 - d_1 \\
\tilde{d}_2 &= \hat{d}_2 - d_2
\end{aligned}
$$

The switching gain K can be found by applying the requirement of $\dot{V} < 0$. We select the particular choice of

$$K \geq |\tilde{m}\ddot{\eta}_r + \tilde{d}_1\dot{\eta} + \tilde{d}_2|\dot{\eta}|\dot{\eta}_r| + \mu \tag{4.49}$$

where $\mu > 0$. This implies that

$$\dot{V} \leq -(K_d + d_2|\dot{\eta}|)s^2 - \mu|s| \leq 0 \tag{4.50}$$

which further implies that s is bounded. Finally, applying Barbalat's lemma [20] yields the result of $s \to 0$ and thus $e \to 0$ as $t \to \infty$.

In summary, the sliding mode control signal is:

$$\tau = \hat{m}\ddot{\eta}_r + \hat{d}_1\dot{\eta} + \hat{d}_2|\dot{\eta}|\dot{\eta}_r - K_d s - K\text{sgn(s)}. \tag{4.51}$$

The motion of the system consists of a *reaching phase* during which the system trajectory starting off the manifold $s = 0$ moves towards it and reaches it in finite time, and a following *sliding phase* during which the motion is confined to the manifold $s = 0$ and the dynamics of the system are represented by the model $\dot{e} = -\lambda e$.

The striking feature of sliding mode control is its robustness with respect to system parameters. We only need to know the uppper bound $|\tilde{m}\ddot{\eta}_r + \tilde{d}_1\dot{\eta} + \tilde{d}_2|\dot{\eta}|\dot{\eta}_r|$ and during the sliding phase, the motion is completely independent of system parameters.

Chattering

Because of imperfections in switching devices and delays, sliding mode control suffers from *chattering*. In Figure 4.9, the system trajectory starts off the sliding surface $s = 0$. It first hits the manifold at some point a. In ideal sliding mode control, the trajectory should start sliding on the manifold from point a. In reality, there will be a delay between the time the sign of s changes and the time the control switches. During this delay period, the trajectory crosses the manifold. When the control switches, the trajectory reverses its direction and heads again toward the manifold. Once again it crosses the manifold, and repetition of this process creates the "zig-zag" motion known as chattering. See Figure 4.10.

Chattering results in low control accuracy, high heat losses in electrical power circuits, and high wear of moving mechanical parts. It may also excite unmodeled high-frequency dynamics, which degrades the performance of the system and may even lead to instability.

Practically, for the controller to perform properly, elimination of chattering is desirable. One method to eliminate chattering is to replace the $\text{sgn}(\cdot)$ function in the control law with a saturation function $\text{sat}(\cdot)$ to smooth out the discontinuity inside a boundary layer [30]:

$$\text{sat}(s/\phi) = \begin{cases} \text{sgn}(s) & \text{if } |s/\phi| > 1 \\ s/\phi & \text{otherwise} \end{cases} \tag{4.52}$$

where ϕ is the boundary layer thickness.

Furthermore, we will examine the derivative of the Lyapunov function after replacing $\text{sgn}(\cdot)$ with $\text{sat}(\cdot)$ for both inside and outside the boundary layer:

- Inside the boundary layer: $|s/\phi| \leq 1$

$$\begin{aligned} \dot{V} &= -(K_d + d|\dot{\eta}|)s^2 + (\tilde{m}\ddot{\eta}_r + \tilde{d}_1\dot{\eta} + \tilde{d}_2|\dot{\eta}|\dot{\eta}_r)s + Ks\frac{s}{\phi} \\ &\leq -(K_d + d|\dot{\eta}| + \frac{\mu}{\phi})s^2 \end{aligned} \tag{4.53}$$

- Outside the boundary layer: $|s/\phi| > 1$

$$\begin{aligned} \dot{V} &= -(K_d + d|\dot{\eta}|)s^2 + (\tilde{m}\ddot{\eta}_r + \tilde{d}_1\dot{\eta} + \tilde{d}_2|\dot{\eta}|\dot{\eta}_r)s + Ks\,\text{sgn}(s) \\ &= -(K_d + d|\dot{\eta}|)s^2 - \mu|s| \end{aligned} \tag{4.54}$$

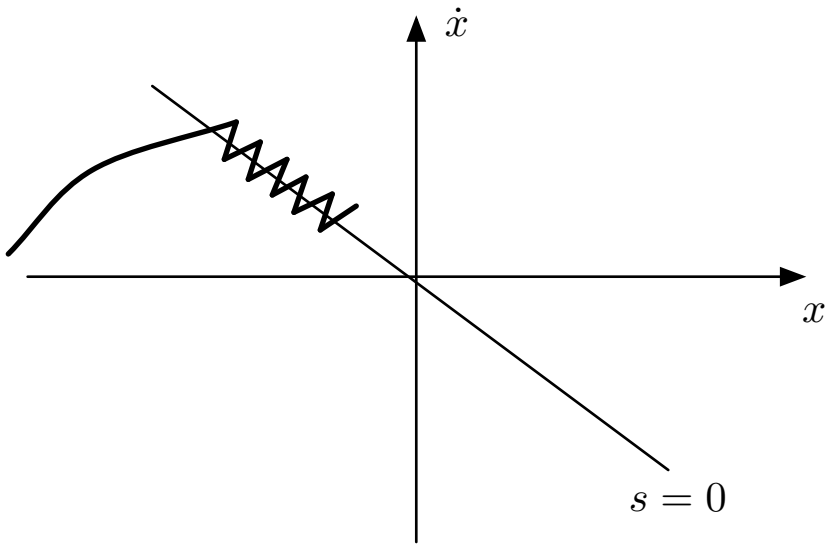


Figure 4.10: Chattering as a result of imperfect control switchings

For better accuracy, we need to choose ϕ as small as possible, but too small a value of ϕ will induce chattering in the presence of time delays or unmodeled fast dynamics.

4.4 Summary

In this chapter, a planar trajectory tracking in the x-y plane motion was developed using the integrator backstepping technique. This controller is Lyapunov stable with the condition that the actuator will never saturate. The depth and bearing controller were implemented using sliding mode control, which is robust when the nonlinear system model has uncertainty in its parameters. The planar trajectory tracking controller and depth controller work together as a 3-D trajectory tracking controller because the planar motion and the vertical motion of the vehicle are decoupled. The controllers will be validated by simulation and actual tests on the VideoRay Pro III in the next Chapter.

Chapter 5

System Evaluation

In this chapter an analysis of controller performance is presented from simulations and experiments. We start by describing the system architecture that we use for the trajectory tracking control, followed by a description of the equipment used in the experimental tests. Finally, the test results will be presented.

5.1 System Architecture

The block diagram of the trajectory tracking system used for the VideoRay Pro III is shown in Figure 5.1. It primarily consists of a trajectory reference input, a control system, and a physical system. The control system is composed of a planar trajectory tracking controller, a depth controller, a bearing controller, a command conversion and an unscented Kalman filter. These controllers were described in Chapter 4. The unscented Kalman filter was described in Chapter 3.

The physical system is composed of a VideoRay Pro III ROV system and a Pilot underwater acoustic positioning system.

The working principle of this entire trajectory tracking system is described as follows:

1. The control system takes the reference trajectory and decides in which mode it will run. The reference trajectory contains the horizontal position information x, y and their first three time derivatives:$\dot{x}$, $\ddot{x}$, $\dddot{x}$, $\dot{y}$, $\ddot{y}$ and $\dddot{y}$.

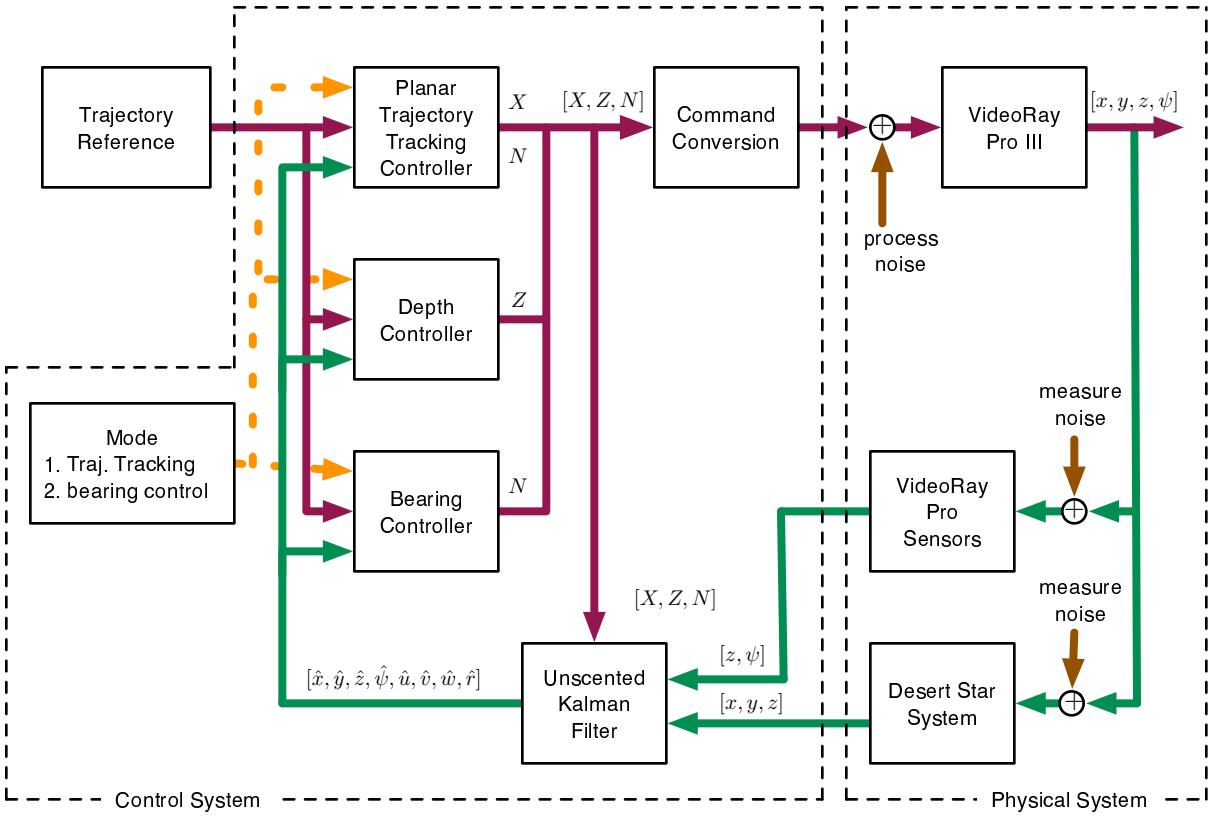


Figure 5.1: Architecture of Entire Tracking System for VideoRay Pro III

2. The control system then computes the control input based on the reference trajectory and the state feedback from the unscented Kalman filter;

3. The control system converts the control input of surge force X, yaw torque N and heave force Z into the thruster control command and sends them to the VideoRay Pro III.

4. Meanwhile, the unscented Kalman filter predicts the system states $[x\ y\ z\ \psi\ u\ v\ w\ r]$ based upon the control input. Whenever there are measurements from the physical system coming in, the UKF corrects the system states with the knowledge of the measurement models.

The controller works at a sampling frequency of 25 Hz. The sensor system on board VideoRay Pro III provides the information of heading angle ψ and depth z, also at the bandwidth of 25 Hz. The Pilote acoustic positioning system works at a lower bandwidth of about one measurement every 1.5 to 3 seconds, depending on the distance of the sonar transducers.

The performance of the trajectory tracking system will be investigated by using simulations and experiments in the following sections.

5.2 Simulation Results

The controller performance was studied by computer simulation. In this simulation, the vehicle is initially located at the origin $(0,\ 0)$ with heading angle $\psi = 0$. The reference trajectory is a circle with the following parameters: radius=2.5 m, motion speed is 0.3 m/s, depth=1 m, starting point (2.5, 0) at time $t_0 = 0$ and stopping time at $t_f = 45$.

The process noise for the vehicle has the covariance of $\mathbf{Q} = \text{diag}(0.05,\ 0.05,\ 0.05)$ for its three thrusters. The measurement variances for heading angle and depth are obtained by tests, which are 0.05 rad^2 and 0.03 m^2 respectively. The variance for position measurement from the Pilot acoustic system is given by the manufacturer as 0.4 m^2, which is also confirmed by test. The heading and depth measurement will be obtained at the frequency of 25 Hz, while the position information from the Pilot system will come out once every 2 seconds.

The simulation result is shown in Figure 5.2. The desired trajectory in x-y plane is shown with dashed blue line. The position measurements are shown with asterisks. The red dash-dot line shows the simulated vehicle trajectory that is calculated with the 6-DOF nonlinear dynamic model of the VideoRay Pro III developed in Chapter 2. The green solid line shows the estimated vehicle position calculated by the UKF based on the decoupled dynamic model in Equation 2.23, 2.24 and 2.25.

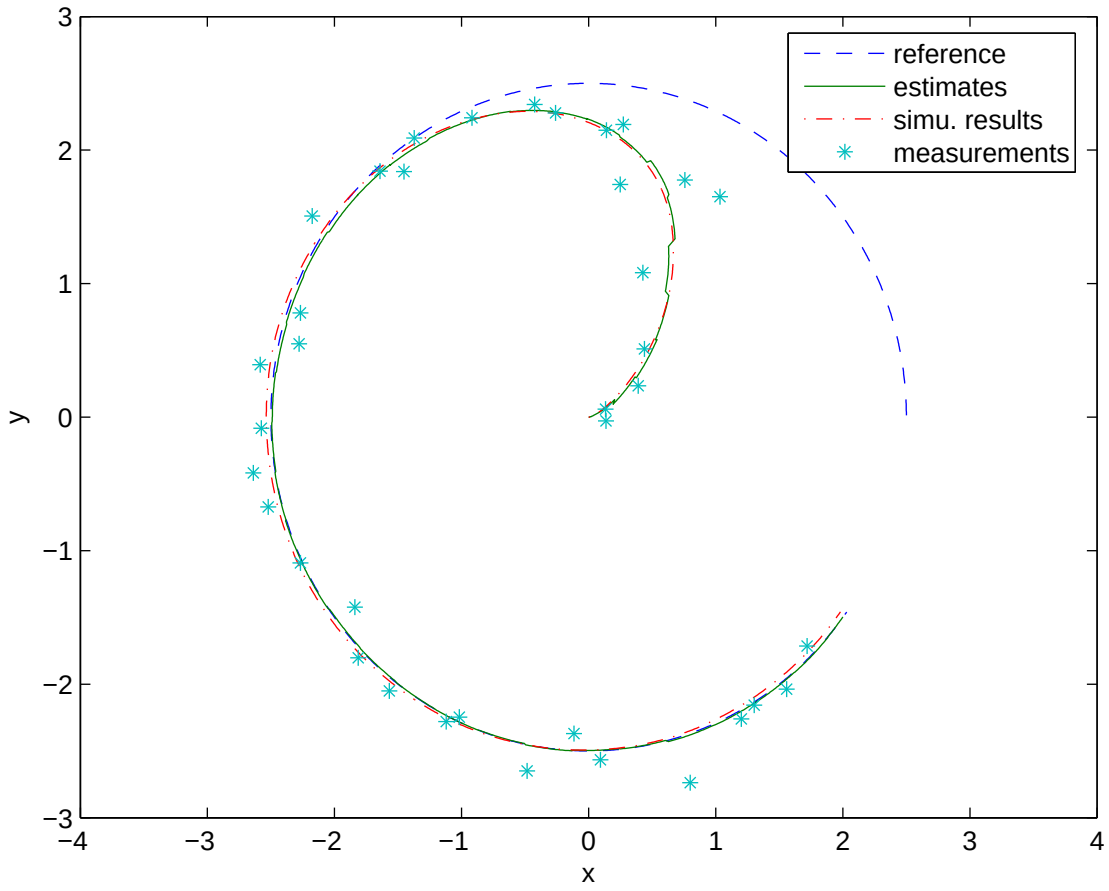
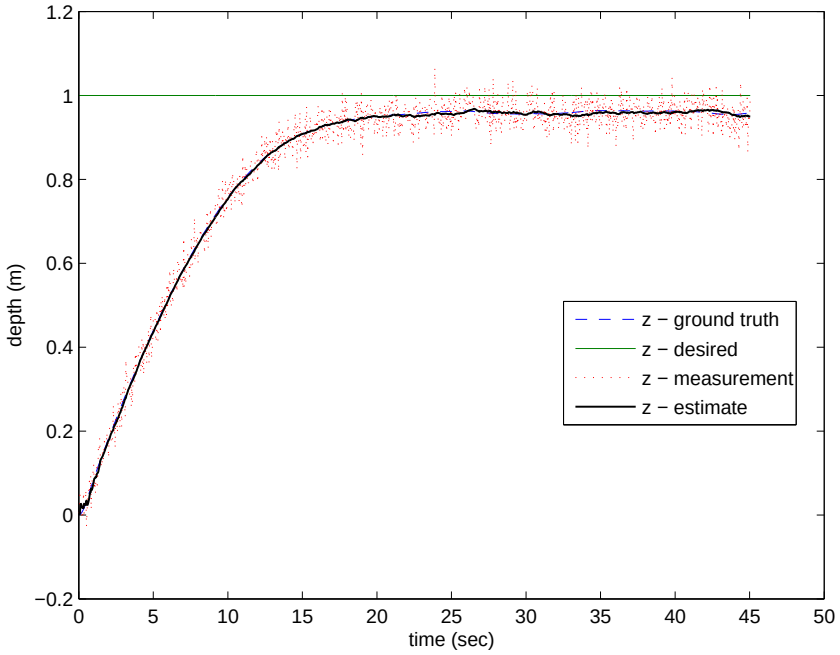
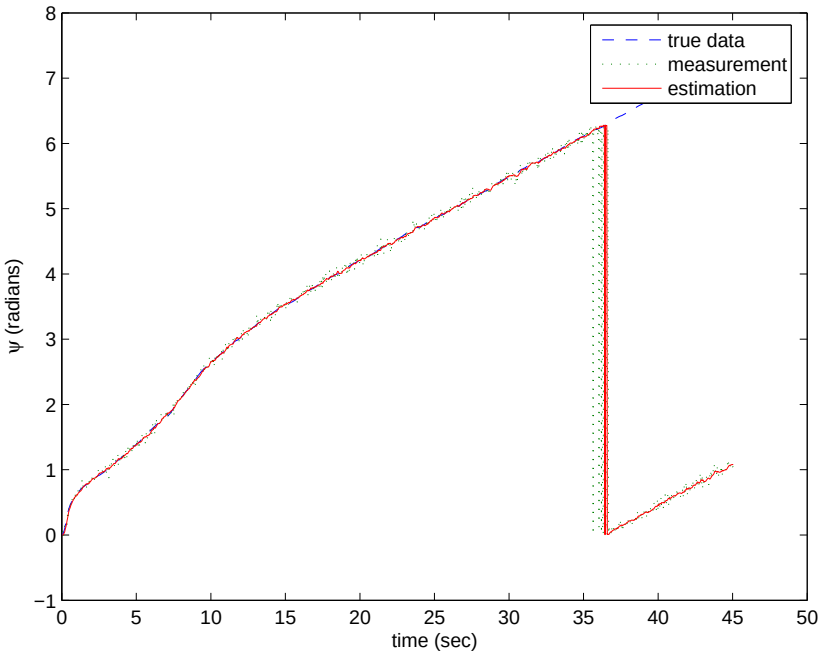


Figure 5.2: Planar trajectory tracking simulation

Figure 5.3 shows the depth and yaw position in this simulation. Figure 5.4 shows the

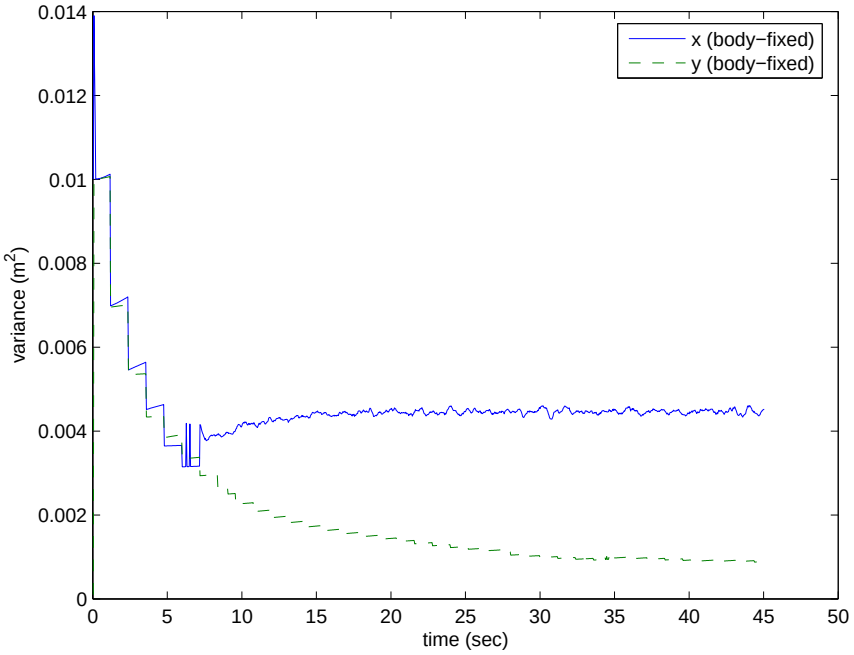


(a) depth for ptt simulation

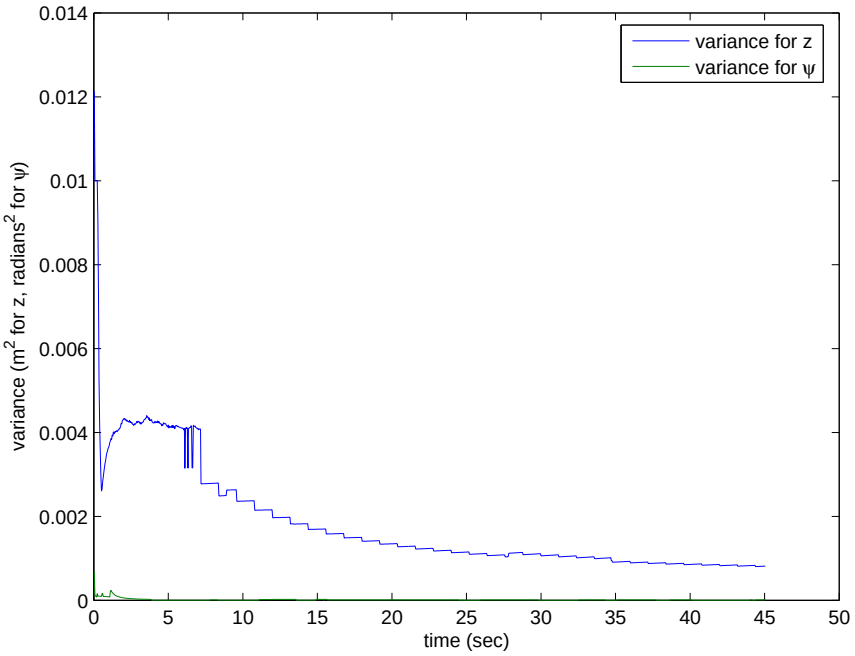


(b) yaw angle for ptt simulation

Figure 5.3: Depth and bearing in simulation

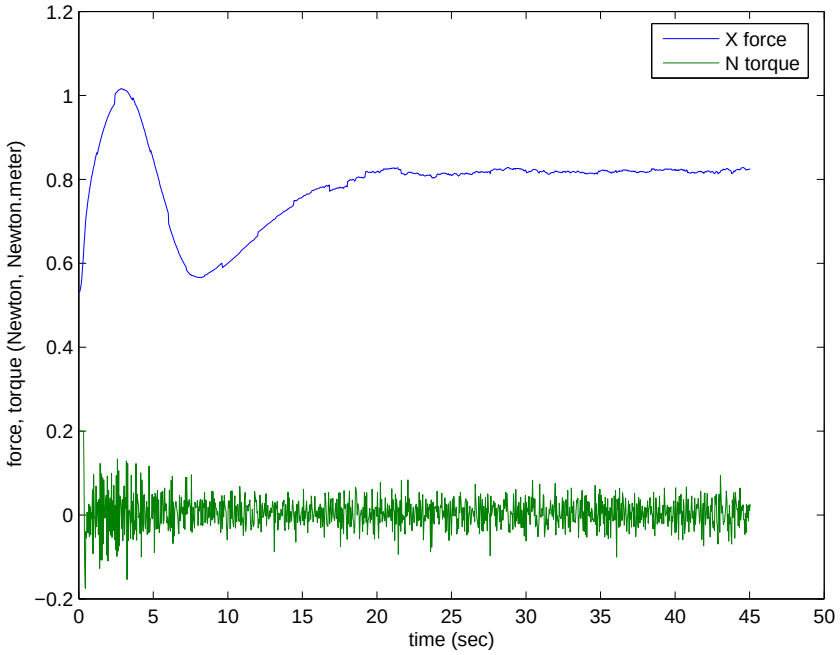


(a) x and y variance

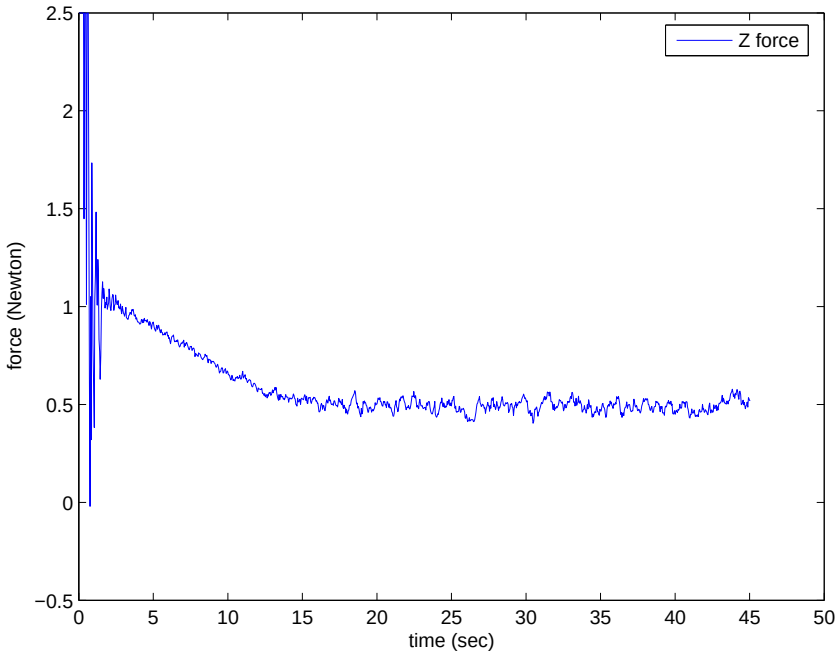


(b) depth and yaw variance

Figure 5.4: Position variance in simulation



(a) force/torque in surge/yaw direction



(b) force in heave direction

Figure 5.5: Controller input forces and torque in simulation

estimated covariance computed by the UKF. The controller output forces and torque are shown in Figure 5.5.

From the simulation, we see that the proposed controller makes the tracking error converge to a very small value. Therefore, the performance of the tracking error system is satisfactory in simulation.

5.3 Experimental Results

5.3.1 Experimental Setup

We used the VideoRay Pro III micro ROV as the test vehicle. It is propelled by two horizontal thrusters for surge and yaw motion control, and a vertical thruster for heave motion control. The dynamics of the vehicle are underactuated and input constrained, and exhibit highly nonlinear behavior. This makes it challenging to control the vehicle to track a predefined trajectory in real-time.

We implemented the planar trajectory tracking controller and the sliding mode controller for depth and bearing tracking on the VideoRay Pro III micro ROV. The control program is implemented in C++ running on a laptop which communicates with the VideoRay's control console through a serial communication. The control bandwidth is about 25Hz, *i.e.*, the control sampling time is 0.04 seconds. The depth and heading sensors on board VideoRay Pro III also provide us the measurement with bandwidth of 25 Hz.

The Pilot acoustic positioning system was used for the tests. The system consists of two principle parts: a surface station and an ROV transponder. The surface station has three sonar transducers attached and put into the water over the side of the boat. The transducers form a triangle which is used to locate the ROV. The surface station will track the underwater vehicle that is equipped with a ROV transponder.

Figure 5.6 shows how the Pilot system works. The surface station is on the boat. The three cabled sonar transducers are lowered over the side. The ROV transponder is mounted on the underwater vehicle. To find the ROV position, the surface station transmits an interrogate signal through transducer #1. This signal travels through the water and reaches the ROV mounted transponder, which replies with a message that includes its

current depth. The reply travels back to all three surface station transducers. The surface station measures the time elapsed between the transmission of the interrogate and the reception of the ROV reply at transducers #1, #2 and #3. Because the speed of sound in water is well known, these signal run times are easily converted to distance (d_1, d_2 and d_3. Finally, calculation with trigonometry yields the position of the ROV.

The tests were conducted in the swimming pool in the University of Waterloo. We used the Pilot underwater acoustic system as the position system. Three cabled sonar transducers are lowered over the three corners of the pool constructing a triangle. A sonar transponder was mounted on the VideoRay Pro III ROV to receive and respond the acoustic signal from the three cabled sonar transducers.

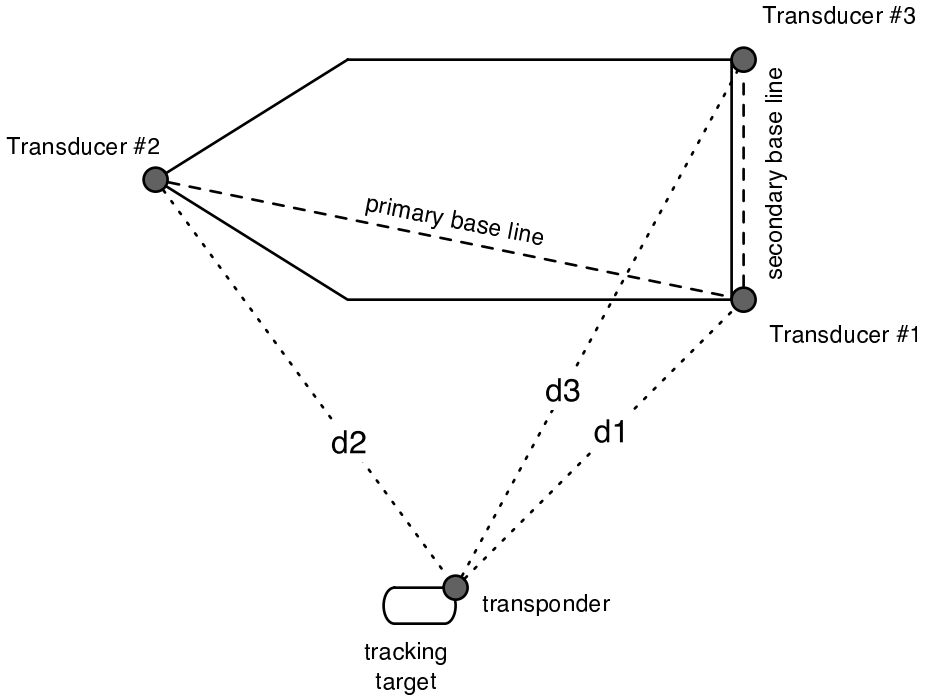


Figure 5.6: Target position determined by "Short Baseline" method

5.3.2 Test Results

Seven tests were conducted with the desired trajectory being straightline starting from a position of (3 m, 0) in the x-y plane and moving to the target position of (3 m, 7.8 m); Once the vehicle arrives the target position, it returns back and moves to the original starting points. During the entire trajectory, the desired depth is set to 0.5 m below water surface. One of these tests is shown in Figure 5.7, where the arrow shows the moving direction of the VideoRay Pro III.

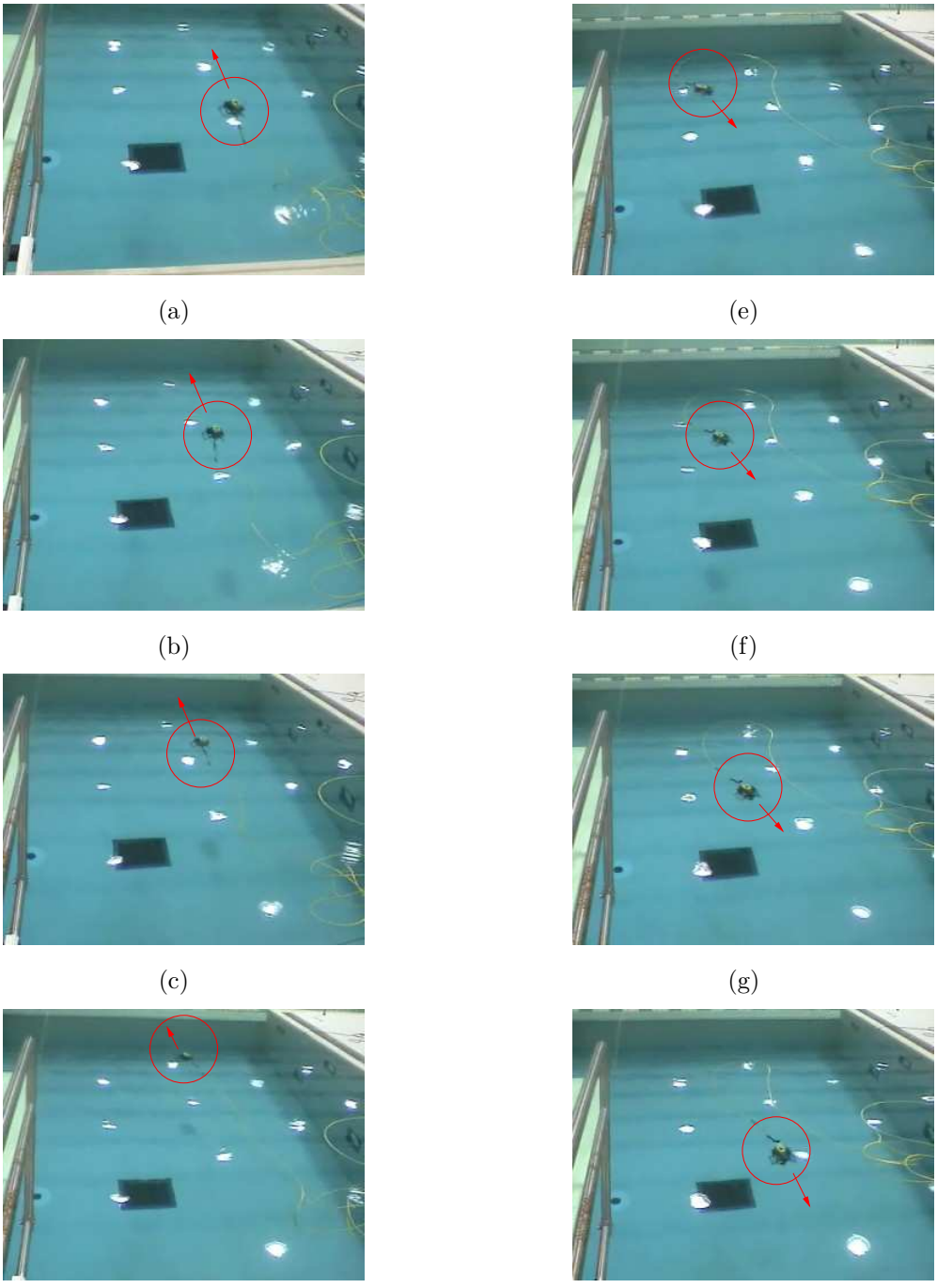
The statistics of the test results are listed in Table 5.1. The results show that the controller works well in tracking the trajectory. The mean error of the actual trajectory is within 0.2 m. The maximum variance is 0.245 m.

experiment #	mean value of x	variance	standard deviation
1	3.19	0.185	0.43
2	2.91	0.245	0.50
3	3.10	0.017	0.13
4	3.04	0.019	0.14
5	3.19	0.032	0.18
6	2.89	0.070	0.27
7	2.97	0.024	0.15

Table 5.1: Straightline test results

The reference and actual trajectory tracked for one of the tests are shown in Figure 5.8. The VideoRay Pro III was launched at (2 m, 0), which is 1 meters off the desired starting point. Note that the vehicle moves towards the reference trajectory and eventually converges to a neighborhood of the trajectory. The controller parameters were first obtained by simulation and then fine tuned by trial and error in the experiments. The parameters that we used in the expermients are:

- The planar trajectory tracking controller parameters:
  $k_e = 4.0$, $\delta_1 = -0.01$, $\delta_2 = -0.001$, $k_{\phi_1} = 20$, $k_{\phi_2} = 20$ and $k_{z_2} = 0.1$.
- The depth controller parameters:
  $\lambda = 4.0$, $\eta = 0.1$, $k_d = 8.0$, $\phi = 0.01$.



(a) (e)

(b) (f)

(c) (g)

(d) (h)

Figure 5.7: Pool test

- The heading controller parameters:
  $\lambda = 2.0$, $\eta = 0.01$, $k_d = 0.05$, $\phi = 0.02$.

Note that the Pilot acoustic positioning system returns the position information with relatively large noise (with standard deviation of about 0.15 m). Also, not only the position measurements are noisy, they are coming up at vary time intervals as well. These noisy measurements are filtered out throught the unscented Kalman filter. In the plots of position variance shown in Figure 5.9 and Figure 5.11, we can see that the variances grow when there are no measurements coming in. At the moment the measurement is obtained, the position covariance is reduced as a result of the sensor fusion achieved through the UKF.

Figure 5.10 shows the depth trajectory and the yaw angle. The tests show that the performance of the trajectory tracking controller relies heavily on the accuracy and performance of the compass. However, the compass is affected in a large extent by the environment, especially for the indoor test. As we can see, when the VideoRay Pro III is heading forward, the compass response is relatively smooth. When the vehicle is moving backwards returning to the starting point, the response of the compass is very sensitive to the environment and exhibits a large error.

In Figure 5.10, we see that the depth sensor has a resolution of about 0.175 meters. This characteristic of the depth sensor is compensated by use of the UKF. Hence, the depth estimates is smoothed out, and the resulting performance of the depth control is satisfied.

Two other field tests were conducted in Paradise Lake in Waterloo. In these tests, no positioning system was used. The reference trajectory is given as a half figure 8. The vehicle will only use its compass for navigation. The results are shown in Figure 5.12 and Figure 5.13. In Figure 5.12(a), the vehicle's position estimates match the reference trajectory considerably well. In Figure 5.13(a), the vehicle's estimates match the reference very well except when in the last part the vehicle got stuck in weeds.

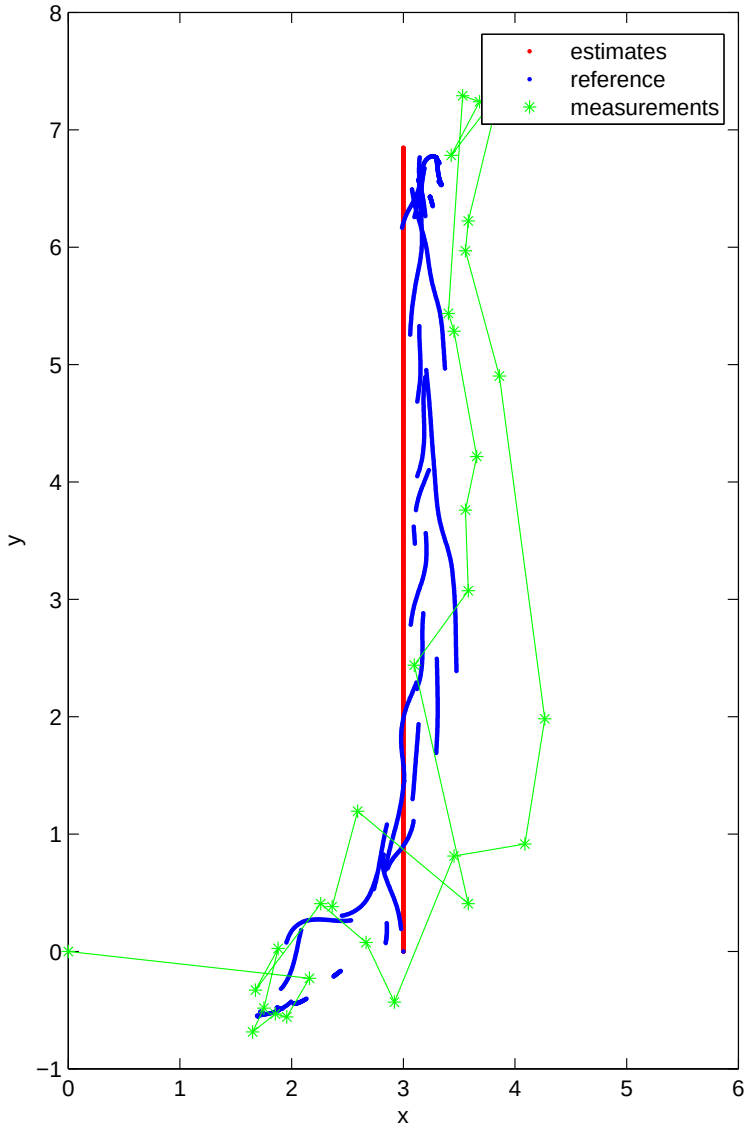


Figure 5.8: Straight line test trajectory: The reference trajectory is a straightline starting from (3, 0) to (3, 7.8) and then returning back along the same line to the starting point; The green asterisks show the Pilot measurements as the vehicle is moving along the trajectory. The blue line segments show the estimates of the vehicle's position achieved through the UKF.

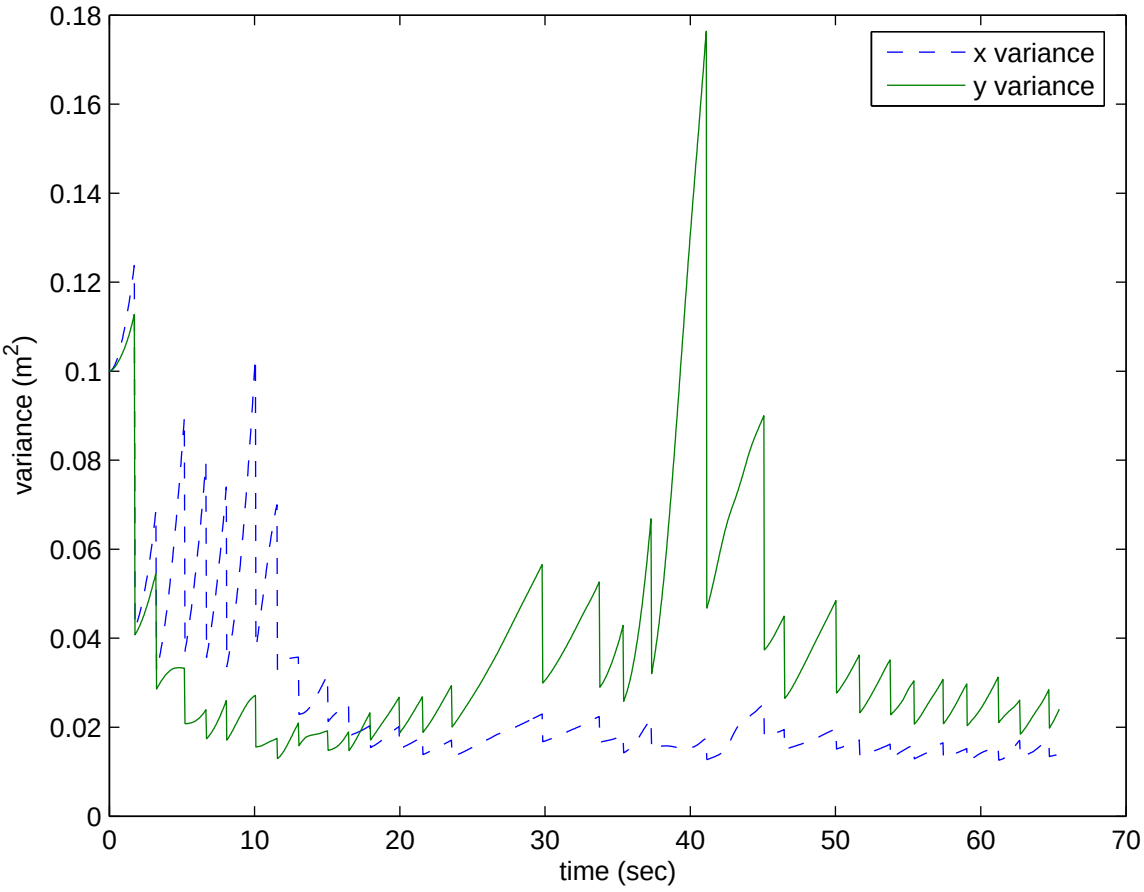
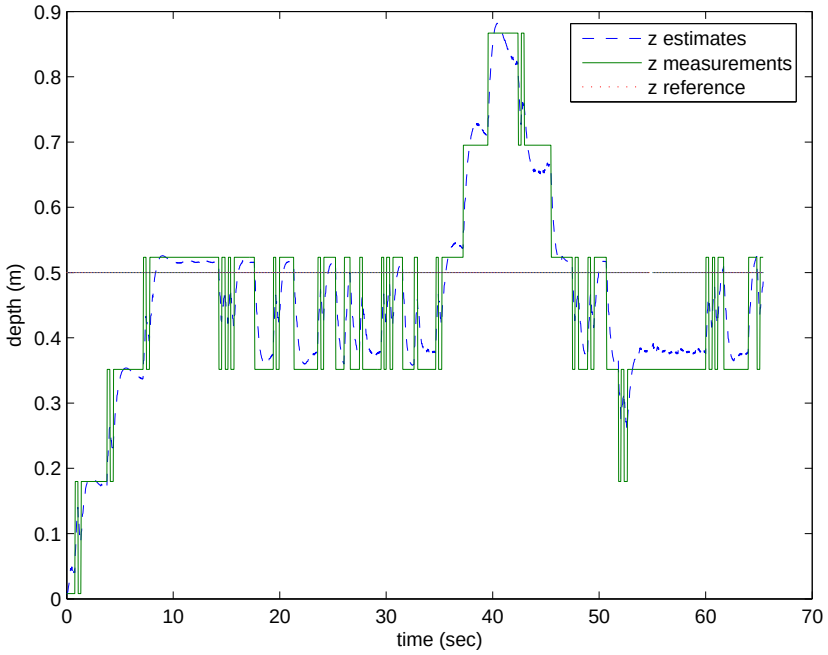
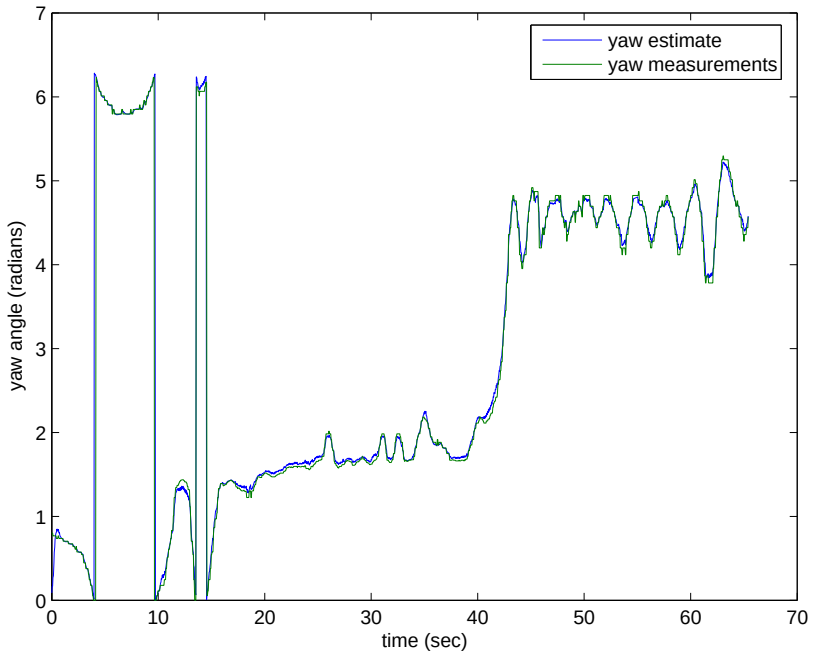


Figure 5.9: Straight line test: x and y variance

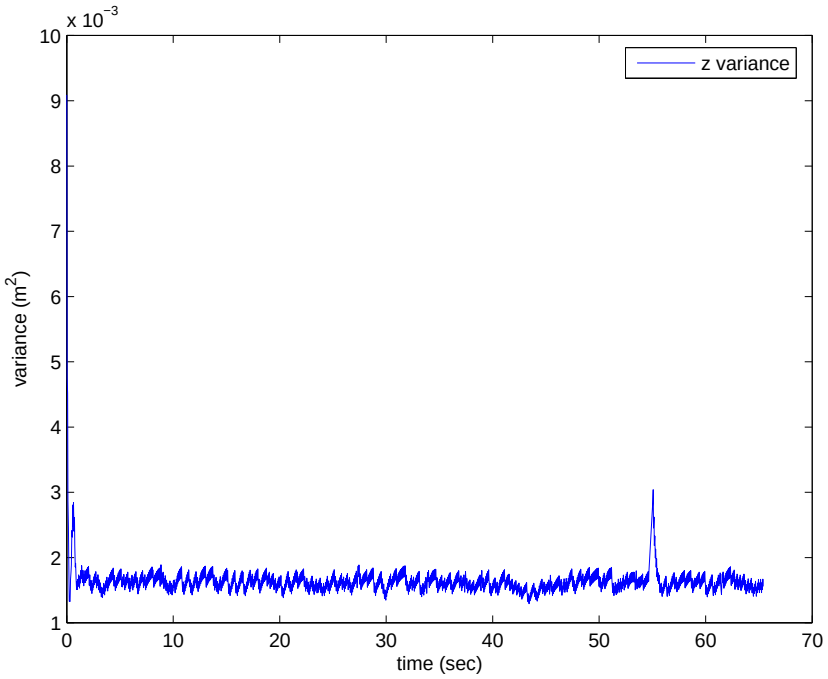


(a) depth

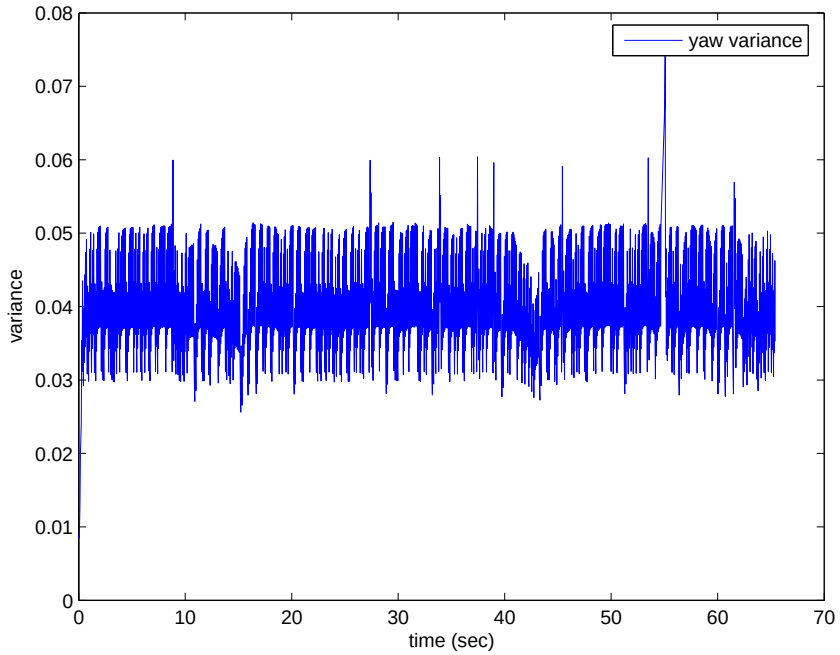


(b) yaw

Figure 5.10: Test results for depth and yaw

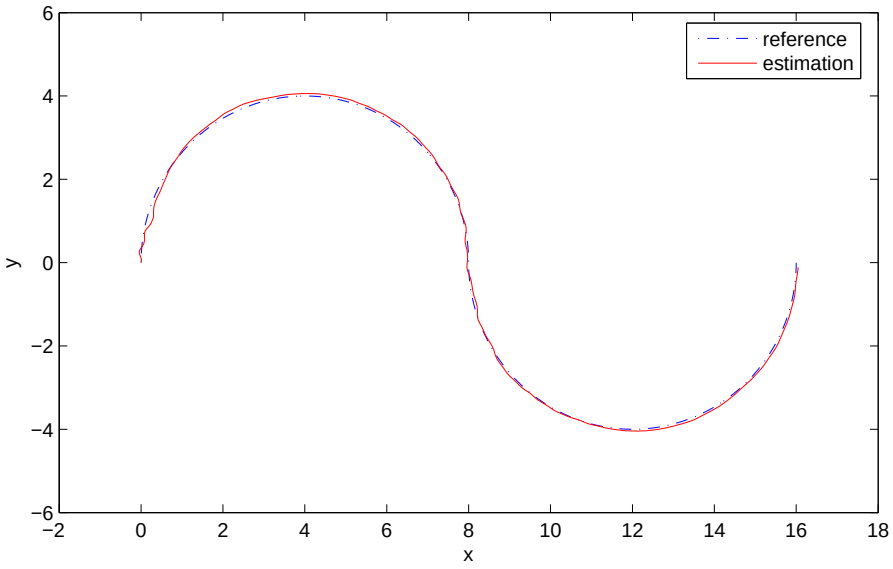


(a) depth variance

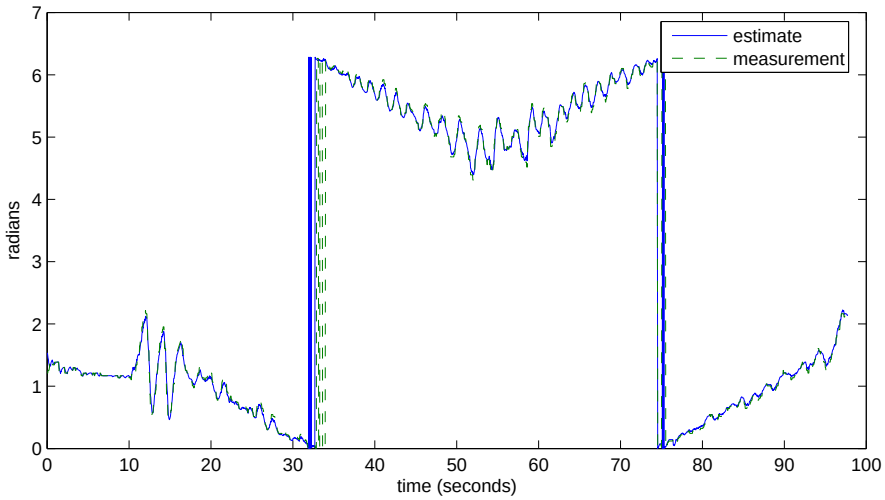


(b) yaw variance

Figure 5.11: Straight line test: z variance

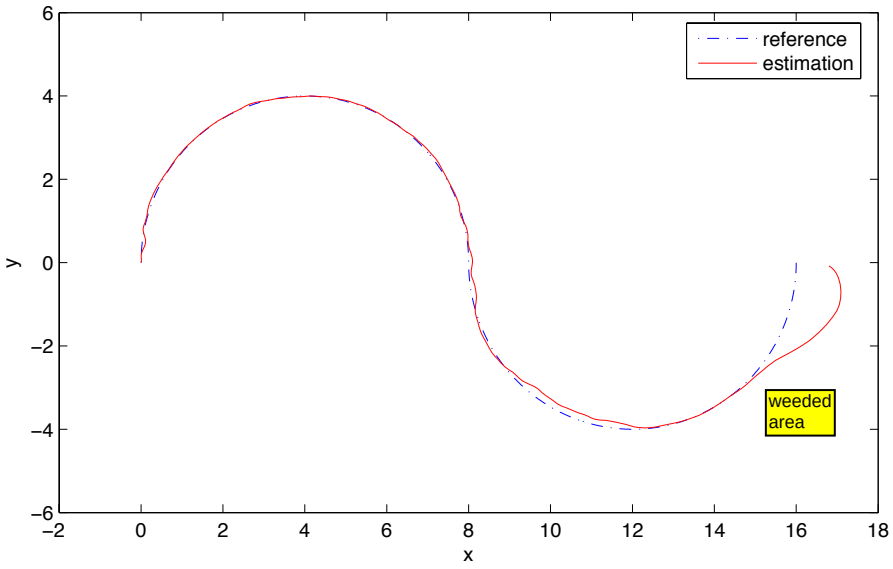


(a) planar trajectory

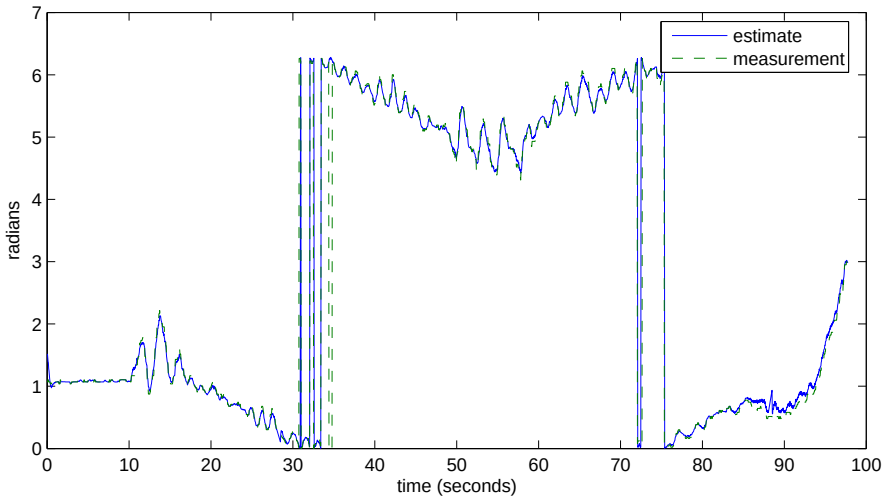


(b) yaw angle

Figure 5.12: Figure 8 test #1



(a) planar trajectory



(b) yaw angle

Figure 5.13: Figure 8 test #2

Chapter 6

Conclusions

In this work, we presented the development and analysis of a dynamical vehicle model as well as a trajectory tracking controller for the VideoRay Pro III ROV.

With respect to vehicle modeling, our basic approach was to model the vehicle as having constant inertial and added mass characteristics, decoupled vehicle motion due to its symmetric geometric profile, and low operating speed. The hydrodynamic coefficients of the model were determined by both theoretical and experimental approaches. The model was verified by experiments and exhibited adequate accuracy for the design of trajectory tracking controller.

A state estimator was designed using the unscented Kalman filter. Since the vehicle's model exhibits high non-linearity and the vehicle has a large operating range, the unscented Kalman filter was used to overcome several drawbacks that come with the traditional extended Kalman filter, which has been widely used for state estimation.

With the decoupled vehicle model developed, we designed a trajectory tracking controller using the integrator backstepping technique, based on the work done by Aguiar *et al.* [2]. As a result, we obtained a Lyapunov stable trajectory tracking controller, considering the quadratic damping terms in the dynamical model which were neglected by Aguiar *et al.*. For the depth and heading control, we developed a sliding mode controller which provided robust tracking control despite the the fact that the vehicle model may have inaccurate parameters and may be subject to unmodeled disturbances during its mission.

The controllers were validated by simulation and experiments. For validation, we used

the Pilot acoustic underwater positioning system as our position measuring device. In the simulation, the closed-loop system was shown to be stable to track a feasible predefined trajectory. In the experiments, we have shown that the controller was working well in tracking a straight line trajectory. The tracking error was within a certain range that is acceptable to the AUV of its category.

Currently, the controller developed heavily relies on the performance the onboard compass to obtain bearing information. Unfortunately, the onboard compass exhibits high nonlinearity measuring heading and is extremely affected by its working environment. This made it difficult to control the vehicle in tracking the trajectory.

6.1 Future Work

As a result of the work performed in this project, serveral issues require further investigation. These include:

- Parameter error is inevitable in the developed dynamic model for the VideoRay Pro III ROV. In particular, when the vehicle is outfitted with various sensors, such as the sonar transponder that we have mounted, the vehicle model needs to be rebuilt and reverified.

- Since the trajectory tracking controller relies heavily on the performance of heading measuring compass, which in our experiment was exhibiting large noise in certain directions, we need to find solutions to filter out the noise.

- Another approach to get relatively accurate heading measurement would be to use a inertial measurement unit (IMU) to compensate for the compass noise, as well as to use a scanning sonar to detect environmental features as reference point to compensate for the compass errors.

Bibliography

[1] SNAME (1950). Nomenclature for treating the motion of a submerged body through a fluid. *The Society of Naval Architects and Marine Engineers, Technical and Research Bulletin*, No. 1-5:1–15, April 1950.

[2] A. P. Aguiar, L. Cremean, and J. P. Hespanha. Position tracking for a nonlinear underactuated hovercraft: Controller design and experimental results. In *Proceedings of the 42nd Conference Decision and Control*, Dec. 2003.

[3] A. P. Aguiar and J. P. Hespanha. Position tracking of underactuated vehicles. In *Proceedings of the 2003 American Control Conference*, Denver, CO, USA, June 2003.

[4] M. Blanke. *Ship Propulsion Losses Related to Automated Steering and Prime Mover Control*. PhD thesis, The Technical University of Denmark, Lyngby, 1981.

[5] M. Caccia, G. Indiveri, and G. Veruggio. Modeling and identification of open-frame variable configuration underwater vehicles. *IEEE Journal of Ocean Engineering*, 25(2):227–240, 2000.

[6] Y. Chen and Y. Rui. Real-time speaker tracking using particle filter sensor fusion. In *Proc. of the IEEE*, volume 92(3), pages 485–494, 2004.

[7] A. Doucet, S. Godsill, and C. Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10(2):197–208, 2000.

[8] O. M. Faltinsen. *Sea Loads on Ships and Offshore Structures*. Cambridge University Press, 1990.

[9] T. I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley Sons, New York, 1994.

[10] K. R. Goheen. Techniques for urv modeling. *Underwater Robotic Vehicles*, 1995.

[11] N. Gordon, D. Salmon, and A. Smith. A novel approach to nonlinear/nongaussian bayesian state estimation. In *IEEE Proc. Radar Signal Precessing*, volume 140, pages 107–113, 1993.

[12] S. Haykin. *Kalman Filters*, chapter 1, pages 1–21. John Wiley & Sons, Inc., 2001.

[13] S. J. Julier. The scaled unscented transformation. In *American Control Conference*, 2000.

[14] S. J. Julier and H. F. Durrant-Whyte. Navigation and parameter estimation of high speed road vehicles. In *Proc. of Robotics and Automation Conference*, pages 101–105, 1995.

[15] S. J. Julier and J. K. Uhlman. Consistent debiased method for converting between polar and cartesian coordinate systems. In *Proc. of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, simulation and Controls*, volume Multi Sensor Fusion, Tracking and Resource Management II, pages 110–121, jun 1997.

[16] S. J. Julier and J. K. Uhlman. A new extension of the kalman filter to nonlinear systems. In *Proc. of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, simulation and Controls*, volume Multi Sensor Fusion, Tracking and Resource Management II, pages 110–121, jun 1997.

[17] S. J. Julier and J. K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. Technical report, RRG, Dept. of Engineering Science, University of Oxford, 1996.

[18] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. A new approach for filtering nonlinear system. In *Proc. of 1995 American Control Conference*, pages 1628–1632, 1995.

[19] I. Kaminer, A. Pascoal, E. Hallberg, and C. Silvestre. Trajectory tracking controllers for autonomous vehicles: An integrated approach to guidance and control. *Journal of Guidance, Control and Dynamics*, 21(1):29–38, 1998.

[20] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, Upper Saddle River, New Jersey, 3rd edition, 2002.

[21] C. Kwok, D. Fox, and M. Meila. Real-time particle filters. In *Proc. of the IEEE*, volume 92(3), pages 469–481, 2004.

[22] L. Ljung. *System Identification: Theory for the User*. PTR Prentice Hall, Upper Saddle River, N.J., second edition, 1999.

[23] T. W. McLain and S. M. Rock. Experiments in the hydrodynamic modeling of an underwater manipulator. In *Proceedings AUV '96*, Monterey, CA, 1996.

[24] W. Naeem, R. Sutton, and S. M. Ahmad. Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking. *IMarEST Journal of Marine Science and Environment*, C(1):15–25, March 2004.

[25] J. Newman. *Marine Hydrodynamics*. MIT Press, 8 edition, 1989.

[26] F. A. Papoulias. On the nonlinear dynamics of pursuit guidance for marine vehicles. *Journal of Ship Research*, 37(4):342–353, December 1993.

[27] K. Y. Pettensen and H. Nijmeijer. Global practical stabilization and tracking for an underactuated ship - a combined averaging and backstepping approach. In *Proc. IFAC Conf, on Systems structure and Contr.*, pages 59–64, Nantes, France, July 1998.

[28] T. Prestero. Verification of a six-degree of freedom simulation model for the remus autonomous underwater vehicle. Master's thesis, MIT/WHOI Joint Program in Applied Ocean Science and Engineering, 2001.

[29] C. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, 1999.

[30] J. J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice-Hall Int., Eaglewood Cliffs, New Jersey, 1991.

[31] D. A. Smallwood and L. L. Whitcomb. Toward model based trajectory tracking of underwater robotic vehicles: Theory and simulation. In *the 12th International Symposium on Unmanned Untethered Submersible Technology*, Durham, New Hampshire, USA, August 2001.

[32] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. Technical report, Cambridge University Engineering Department, August 2000.

[33] R. van der Merwe and E. Wan. The square-root unscented kalman filter for state and parameter-estimation. In *Proc. of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Salt Lake City, Utah, America, 2001.

[34] E. Wan and R. van de Merwe. *The Unscented Kalman Filter*, chapter 7, pages 221–280. John Wiley & Sons, Inc., 2001.

[35] E. Wan and R. van der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of IEEE Symposium 2000 Adaptive Systems for Signal Processing, Communication and Control (AS-SPCC)*, Lake Louise, Alberta, Canada, 2000.

[36] E. Wan, R. van der Merwe, and A. T. Nelson. Dual estimation and the unscented transformation. *Advances in Neural Information Processing Systems*, 12:666–672, 2000.

[37] D. R. Yoerger, J. G. Cooke, and J. J. E. Slotine. The influence of thruster dynamic on underwater vehicle behavior and their incorporation into control system design. *IEEE Journal of Oceanic Engineering*, 15:167–168, July 1990.

[38] D. R. Yoerger, J. B. Newman, and J. J. E. Slotine. Supervisory control system for the jason rov. *IEEE Oceqnic Engineering*, 15(3):392–400, 1986.

[39] D. R. Yoerger and J. J. E. Slotine. Nonlinear trajectory control of autonomous underwater vehicles using the sliding methodology. In *Proceedings of the ROV'84 Conference*, pages 245–51, 1984.

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