2022-06

# AN INFORMATION THEORETIC APPROACH TO INTERACTING MULTIPLE MODEL ESTIMATION FOR AUTONOMOUS UNDERWATER VEHICLES

Howarth, Timothy J., II

Monterey, CA; Naval Postgraduate School


# NAVAL POSTGRADUATE SCHOOL

**MONTEREY, CALIFORNIA**

# THESIS

**AN INFORMATION THEORETIC APPROACH TO INTERACTING MULTIPLE MODEL ESTIMATION FOR AUTONOMOUS UNDERWATER VEHICLES**

by

Timothy J. Howarth II

June 2022

Thesis Advisor: Douglas P. Horner
Second Reader: Brian S. Bingham

THIS PAGE INTENTIONALLY LEFT BLANK

| REPORT DOCUMENTATION PAGE | | | *Form Approved OMB No. 0704-0188* |
|---|---|---|---|



| **1. AGENCY USE ONLY** *(Leave blank)* | **2. REPORT DATE** June 2022 | **3. REPORT TYPE AND DATES COVERED** Master's thesis |
|---|---|---|

**4. TITLE AND SUBTITLE**
AN INFORMATION THEORETIC APPROACH TO INTERACTING MULTIPLE MODEL ESTIMATION FOR AUTONOMOUS UNDERWATER VEHICLES

**5. FUNDING NUMBERS**

**6. AUTHOR(S)** Timothy J. Howarth II

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Naval Postgraduate School
Monterey, CA 93943-5000

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
N/A

**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES** The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

**12a. DISTRIBUTION / AVAILABILITY STATEMENT**
Approved for public release. Distribution is unlimited.

**12b. DISTRIBUTION CODE**
A

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

Accurate and robust autonomous underwater navigation (AUV) requires the fundamental task of position estimation in a variety of conditions. Additionally, the U.S. Navy would prefer to have systems that are not dependent on external beacon systems such as global positioning system (GPS), since they are subject to jamming and spoofing and can reduce operational effectiveness. Current methodologies such as Terrain-Aided Navigation (TAN) use exteroceptive imaging sensors for building a local reference position estimate and will not be useful when those sensors are out of range. What is needed are multiple navigation filters where each can be more effective depending on the mission conditions. This thesis investigates how to combine multiple navigation filters to provide a more robust AUV position estimate. The solution presented is to blend two different filtering methodologies utilizing an interacting multiple model (IMM) estimation approach based on an information theoretic framework. The first filter is a model-based Extended Kalman Filter (EKF) that is effective under dead reckoning (DR) conditions. The second is a Particle Filter approach for Active Terrain Aided Navigation (ATAN) that is appropriate when in sensor range. Using data collected at Lake Crescent, Washington, each of the navigation filters are developed with results and then we demonstrate how an IMM information theoretic approach can be used to blend approaches to improve position and orientation estimation.

**14. SUBJECT TERMS**
Particle Filter, PF, Active Terrain-Aided Navigation, ATAN, Terrain-Aided Navigation, TAN, Boltzman Entropy, BE, Bayesian Filtering, BF, dead reckoning, DR, Extended Kalman Filter, EKF, Kalman Filter, KF, Gaussian Process Model, GPM, simultaneous localization and mapping, SLAM, probability density function, PDF, dual camera system, global positioning system, GPS, external beacons, interacting multiple model, IMM, Shannon entropy, autonomous underwater vehicle, AUV

**15. NUMBER OF PAGES**
153

**16. PRICE CODE**

| **17. SECURITY CLASSIFICATION OF REPORT** | **18. SECURITY CLASSIFICATION OF THIS PAGE** | **19. SECURITY CLASSIFICATION OF ABSTRACT** | **20. LIMITATION OF ABSTRACT** |
|---|---|---|---|
| Unclassified | Unclassified | Unclassified | UU |

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release. Distribution is unlimited.**

# AN INFORMATION THEORETIC APPROACH TO INTERACTING MULTIPLE MODEL ESTIMATION FOR AUTONOMOUS UNDERWATER VEHICLES

Timothy J. Howarth, II
Lieutenant, United States Navy
BS, United States Naval Academy, 2015

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN MECHANICAL ENGINEERING**

from the

**NAVAL POSTGRADUATE SCHOOL**
**June 2022**

Approved by: Douglas P. Horner
Advisor

Brian S. Bingham
Second Reader

Garth V. Hobson
Chair, Department of Mechanical and Aerospace Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

# ABSTRACT

Accurate and robust autonomous underwater navigation (AUV) requires the fundamental task of position estimation in a variety of conditions. Additionally, the U.S. Navy would prefer to have systems that are not dependent on external beacon systems such as global positioning system (GPS), since they are subject to jamming and spoofing and can reduce operational effectiveness. Current methodologies such as Terrain-Aided Navigation (TAN) use exteroceptive imaging sensors for building a local reference position estimate and will not be useful when those sensors are out of range. What is needed are multiple navigation filters where each can be more effective depending on the mission conditions. This thesis investigates how to combine multiple navigation filters to provide a more robust AUV position estimate. The solution presented is to blend two different filtering methodologies utilizing an interacting multiple model (IMM) estimation approach based on an information theoretic framework. The first filter is a model-based Extended Kalman Filter (EKF) that is effective under dead reckoning (DR) conditions. The second is a Particle Filter approach for Active Terrain Aided Navigation (ATAN) that is appropriate when in sensor range. Using data collected at Lake Crescent, Washington, each of the navigation filters are developed with results and then we demonstrate how an IMM information theoretic approach can be used to blend approaches to improve position and orientation estimation.

THIS PAGE INTENTIONALLY LEFT BLANK

# Table of Contents

# List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

# List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

# List of Acronyms and Abbreviations

| | |
|---|---|
| **3DX** | Numurus 3DX-CI-1K-MKS Camera System |
| **ADCP** | acoustic doppler current profiler |
| **AI** | artificial intelligence |
| **AI/RL** | artificial intelligence/reinforcement learning |
| **ARC** | Ames Research Center |
| **ATAN** | Active Terrain-Aided Navigation |
| **AUV** | Autonomous underwater vehicle |
| **BE** | Boltzman Entropy |
| **BF** | Bayesian Filtering |
| **C-IMM** | Classical Interacting Multiple Model |
| **C4I** | Command, Control, Communications, Computers, and Intelligence |
| **CAVR** | Center for Autonomous Vehicle Research |
| **CBWH** | Corrected Background Weighted Histogram |
| **CLT** | Central Limit Theorem |
| **CLTP** | Completed Local Ternary Patterns |
| **CPU** | Central Processing Unit |
| **CRLB** | Cramér-Rao Lower Bound |
| **CTD** | conductivity temperature and depth |
| **CTT** | cross tunnel thrusters |

**DCT** discrete cosine transform

**DFT** discrete Fourier transform

**DOD** Department of Defense

**DOF** degrees of freedom

**DR** dead reckoning

**DVL** doppler velocity log

**EKF** Extended Kalman Filter

**EKPF** Extended Kalman Particle Filter

**EMI** Electromagnetic Interference

**EOM** equations of motion

**ESS** effective sampling size

**FLS** forward looking sonar

**FOV** field of view

**GBF** Generalized Bayesian Filtering

**GPM** Gaussian Process Model

**GPS** global positioning system

**GPU** graphics processing unit

**HOG** Histogram of Oriented Gradients

**I/O** Input/Output

**ICRA** International Conference on Robotics and Automation

**IMM** Interacting Multiple Model

**IMU** inertial measurement unit

**INS** inertial navigation system

**IoT** Internet of Things

**IT-IMM** Information Theoretic Interacting Multiple Model

**IQR** interquartile range

**KF** Kalman Filter

**KLA** Kullback-Leibler average

**KLD** Kullback-Leibler Divergence

**LAAS-CNRS** Laboratoire d'analyse et d'architecture des systèmes

**LBL** long baseline

**LCN** Lake Crescent North

**LCS** Lake Crescent South

**LED** light emitting diode

**lidar** light detection and ranging

**LTI** linear time invariant

**MAD** mean absolute difference

**MAGNA** Multi-platform Anti-jamming GPS Navigation Antenna

**MAP** Maximum *a posteriori*

**MATLAB** Matrix Laboratory

**MC** Markov chain

**MCL** Monte Carlo localization

**MCMC** Markov chain Monte Carlo

**MF** Meta-Filter

**ML** machine learning

**MSD** mean square difference

**MSE** mean square error

**MWS** Micro Web services

**NASA** National Aeronautics and Space Administration

**NAVSSI** Navigation Sensor System Interface

**NEPI** Numurus Edge Platform Interface

**NOAA** National Oceanic and Atmospheric Administration

**NPS** Naval Postgraduate School

**NXCOR** normalized cross-correlation

**ONR** Office of Naval Research

**OPAREA** operational areas

**OSE** Optimal Spatial Estimation

**PDF** probability density function

**PEO C4I** Program Executive Office Command, Control, Communications, Computers and Intelligence

**PF** Particle Filter

**PFL** Particle Filter Localization

**PNT** position, navigation, and timing

**POMCP** Partially Observable Monte Carlo Planning Process

**POSYDON** Positioning System for Deep Ocean Navigation

**PRC** People's Republic of China

**R-IMM** Re-weighted Interacting Multiple Model

**radar** radio detection and ranging

**RBPF** Rao-Blackwellized Particle Filter

**RBT** Rao-Blackwell Theorem

**REMUS** Remote Environmental Monitoring UnitS 100

**RF** radio frequency

**RL** reinforcement learning

**ROS** robot operating system

**RPM** revolutions per minute

**RUI** resident user interface

**SIR** Sequential Importance Sample with Re-sampling

**SIS** Sequential Importance Sampling

**SLAM** simultaneous localization and mapping

**SLF** Selective Logic Filter

**SNAME** Society of Naval Architects and Marine Engineers

**sonar** sound navigation and ranging

**STCAN** Service Technique des Constructions et Armes Navales

**STP** salinity, temperature, and pressure

**TAN** Terrain-Aided Navigation

**TAP** Terrain-Aided Positioning

**TERCOM** Terrain Contour Matching

**TERNAV** Terrain Navigation

**TERPROM** Terrain Profile Matching

**UAV** unmanned aerial vehicle

**UKF** Unscented Kalman Filter

**ULV** unmanned land vehicle

**USBL** ultra short baseline

**USGS** United States Geological Survey

**UT** unscented transform

**UTM** Universal Transverse Mercator

**UUV** unmanned underwater vehicle

**UUVRON 1** Unmanned Underwater Vehicle Squadron One

**UV** Unmanned Vehicles

**VDC** Voltage DC

**XCOR** cross-correlation

# Executive Summary

Within recent years the Department of Defense has directed an accelerated adoption of artificial intelligence (AI) and the creation of a force that is technologically advanced and capable of ensuring the security of the United States. A crucial component of future autonomous maritime operations is the ability for unmanned autonomous vehicles to operate without the use of global positioning system (GPS) or other external beacon systems.

In a rapidly evolving technological world, operating in a GPS denied environment or without the use of systems such as acoustic transponders or even the Positioning System for Deep Ocean Navigation (POSYDON) system has never been more crucial. The leading solution is Terrain-Aided Navigation (TAN) and which utilizes an on-board map and a combination of sensor systems in order to correlate measurements within the known map. The most significant drawback to this method is the requirement for different filter estimation methodologies that may not be designed to work together.

This research will be presented in several parts. The first is to implement a new Extended Kalman Filter (EKF) as a dead reckoning (DR) model on-board the Naval Post Graduate School's Remote Environmental Monitoring UnitS 100 (REMUS) vehicle to improve its estimation during times of inaccurate velocity estimation. Secondly, this research seeks to build a Particle Filter (PF) for use within Active Terrain-Aided Navigation (ATAN) based on information theory. Finally, and perhaps most important, this research seeks to implement a novel information theoretic federation process between the PF and EKF to improve over all state estimation.

THIS PAGE INTENTIONALLY LEFT BLANK

# Acknowledgments

I would like to express my gratitude and thanks my thesis advisor, Dr. Douglas Horner, who was always available and whose mentorship, assistance, and direction are the reason that I am able to complete a meaningful contribution. This work would not be what it is without his guidance and I am truly thankful for his help.

All my love and appreciation to my family, Lauren, Kaia, and Theo. Thank you for all the support you have given me through these long nights. I would not have been able to complete this work without you by my side.

THIS PAGE INTENTIONALLY LEFT BLANK

# CHAPTER 1:
# Introduction

## 1.1 Motivation

As unmanned systems continue to become a more important component of U.S. Naval strategy, robust and accurate pose estimation remains a fundamental task. Current techniques often rely on a single filter which is expected to handle multiple and potentially sporadic sensor measurements. This can create estimation errors, especially when the system and measurement models have unknown uncertainty. The thesis investigates using multiple simultaneous filters and the methodology by which to combine these in order to produce a more robust position estimation approach.

There are many approaches to accurate underwater navigation that include a combination dead reckoning (DR) and external localization methods. They frequently rely on external beacon systems such as global positioning system (GPS), long baseline (LBL), and ultra short baseline (USBL) for position measurements. Despite their reliability and ability to conduct accurate navigation relative to the positioned beacons, there is growing concern [1] about their limitations and vulnerabilities.

They fundamentally are subject to jamming and spoofing vulnerabilities. For example, GPS jamming can be accomplished by producing a radio frequency (RF) signal which is powerful enough that it may "drown out" the transmissions which are being produced from the constellation of satellites [2]. This has the impact of increasing the overall noise level at the GPS receiver can cause a complete loss of GPS estimation. This is significant as many adversaries and near-peer competitors such as Russia and the People's Republic of China (PRC) have demonstrated jamming capabilities [3]. This technique presents the potential to be utilized both offensively or defensively [2] but would present sudden and obvious symptoms that indicate something is wrong with the system.

GPS spoofing involves deliberately mimicking transmissions from satellites which can deceive the receiver into believing that information was sent correctly [4]. This transmission would be indistinguishable from the true GPS signal and can be used by an adversary to

provide false position information or even take control of the system by directing it towards their desired location [5]. Countries such as Iran have used spoofing against the United States and in 2011 captured a U.S. drone by spoofing its GPS to land in Iran when it was believed, by the U.S., to have landed in Afghanistan [6].

There are also limitations associated with using LBL and USBL systems to include the necessity for near-operation installation as well as error associated with the intended position or watch circle due to installation. Although these systems can provide extremely accurate estimates, often within 1 m [7], the proximity requirement severely limits flexibility. These systems would also need to be installed well before an operation and would still require a precise understanding of their error to be used accurately. It is precisely these vulnerabilities and limitations that fuel the motivation for the U.S. Navy to reduce its dependence on external beacon and localization methods for position, navigation, and timing (PNT).

Because of the challenges associated with these system vulnerabilities, the Department of Defense (DOD) has directed policy, responsibility, and procedures for the DOD PNT enterprise [8]. The United States Navy's Program Executive Office Command, Control, Communications, Computers and Intelligence (PEO C4I) has a program to develop state-of-the-art capabilities for accurate and assured operational maneuver of surface and subsurface vehicles [9]. The focus of the program addresses various issues and technologies associated with PNT with the goal being to provide naval assets with assured PNT alternatives to GPS [9]. Figure 1.1, adapted from [9], shows the desired focus on non-GPS aided PNT alternatives for surface and subsurface sensors as an increasing threat environment demands the need to operate in extended periods of total GPS denial.

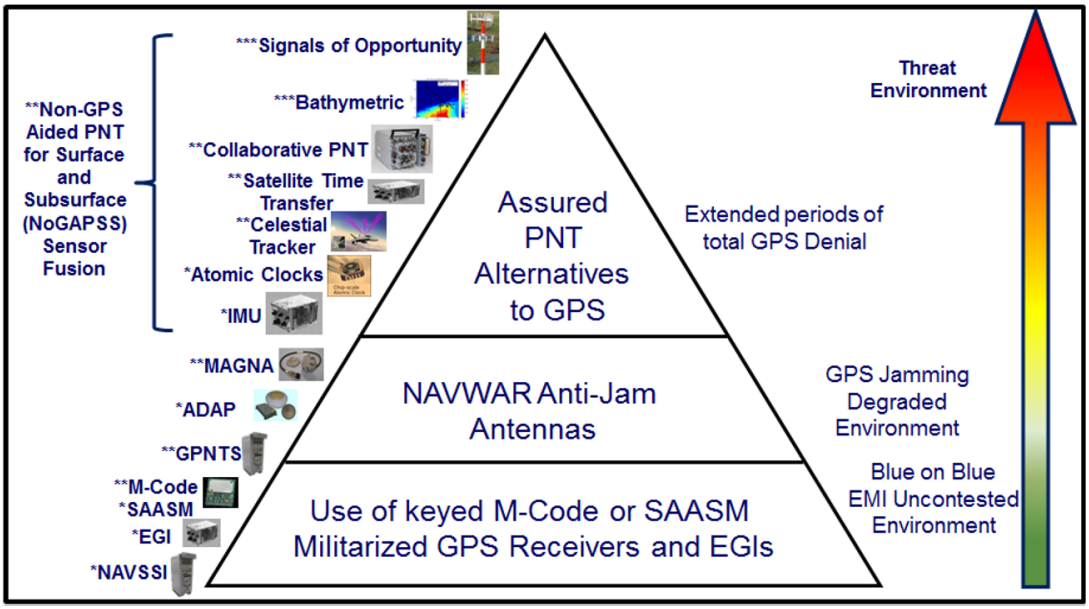



Figure 1.1. Position, Navigation, and Timing Alternatives Hierarchy. Adapted from: [9]

At the lowest threat level are environments which are completely Electromagnetic Interference (EMI) uncontested. In this environment, the Navy can utilize common navigational methods such as the shipboard Navigation Sensor System Interface (NAVSSI) which allows vessels to automatically accept GPS data for real-time velocity, positioning, and time data to shipboard navigational and combat Command, Control, Communications, Computers, and Intelligence (C4I) systems [10]. As the threat level increases to that of GPS jamming and degraded environments, there will be a requirement to use technologies that can defeat jamming threats such as the NavGuard 502 Multi-platform Anti-jamming GPS Navigation Antenna (MAGNA) which can place a null in the direction of jamming and reduce jamming noise. Finally, when GPS is not available, assured PNT alternatives to GPS must be used such signals of opportunity from systems such as LBL, collaborative PNT, or Terrain-Aided Navigation (TAN).

Autonomous underwater vehicle (AUV) operations present significant issues for position estimation techniques. Receipt of a GPS signal requires the vehicle to periodically surface

and limit operational flexibility. Imaging sensors, such as sound navigation and ranging (sonar) and cameras, and doppler velocity log (DVL)/acoustic doppler current profiler (ADCP) have limited range. Figure 1.2, adapted from [5], gives an example from a mission profile from Lake Crescent, WA where four different regimes can impact the filtering process at different times during the mission.

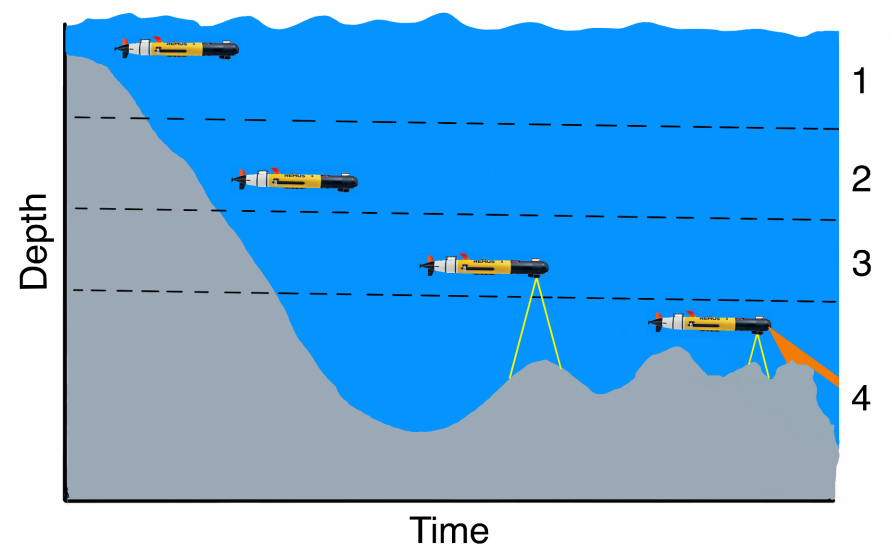



Figure 1.2. Regimes where different filtering techniques may be necessary. Regime 1, AUV operates near the surface and can leverage GPS data. Due to the depth the AUV is unable to utilize any other sensors and must use a DR model. Regime 2 is too deep to quickly access GPS data and not yet deep enough for seafloor-oriented sensors. Regime 3 can use the DVL/ADCP and inertial navigation system (INS) which can provide more accurate motion estimates. Regime 4 can utilize imaging sensors to further improve navigational accuracy. Adapted from: [5]

Table 1.1. Scenario Requirements and Limitations: A - Available, NA - Not Available, L - Limited

| Regime | GPS | DVL/ADCP | Camera | INS | Model |
|---|---|---|---|---|---|
| 1 | A | NA | NA | L | A |
| 2 | NA | NA | NA | L | A |
| 3 | NA | A | NA | A | A |
| 4 | NA | A | A | A | A |

In regime 1, the AUV operates at or near the surface and can leverage GPS and DR models for navigation. Within this example, the depth of the water is too deep for the use of DVL/ADCP since the integrated INS position estimation solution which requires the DVL/ADCP for velocity measurements. This limits the use of the INS. For AUVs this regime is a factor of the size of the instrumentation onboard the vehicle as well as its mission profile. Moving to regime 2, the only available tool for navigation is a model with limited INS use since surfacing for GPS is time and energy inefficient. Regime 3 becomes within range of systems such as the DVL/ADCP and the AUV can begin to more accurately use a model now that there are velocity estimates being provided. In regime 4, the AUV can begin to fully implement all on-board navigation methodologies as there is a wide variety of sensor information. Here an AUV can accurately navigate but must do so with any accrued error from transitioning through the other regimes.

Sensors, such as those listed in Table 1.1, are commonplace for AUVs and provide many of the measurements that produce accurate motion and localization estimates. When these sensors are not available, an accurate hydrodynamic model can be used to transition between the intermediate regimes. An accurate model can provide the AUV with estimates for its forward and side-slip velocity, heading, and heading rate that can be used as a part of an effective DR filter.

The Lake Crescent bathymetry, an example of which is shown in Figure 1.3, became a prime candidate of the necessity of an AUV to operate through different regimes.

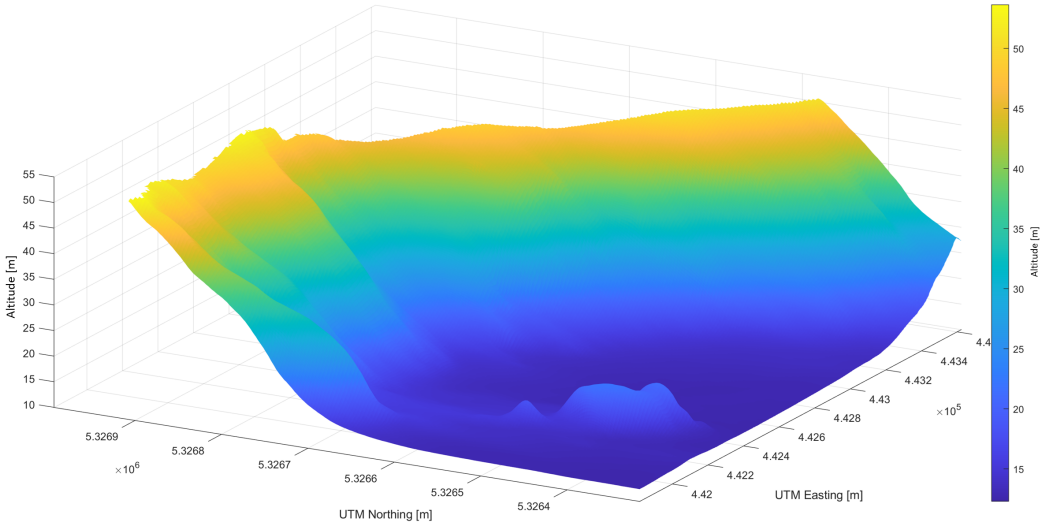



Figure 1.3. Multiple Regimes at Lake Crescent

Figure 1.3 shows one section of an operational area at Lake Crescent with altitudes relative to a datum set at the deepest point of the lake. The depth of water in this figure range from about 10 m to 55 m from the surface. As the AUV moves from south to north within Figure 1.3, there are points at which it must transition between different regimes and will only have certain sensor systems available until it reaches the most northern section. Due to the performance of the DR filter, some of the experiments conducted had final positional errors nearing one kilometer.

There are many limitations to some of the common filtering methods used for underwater navigation. Some filtering methods require uncertainty parameters associated with different measurements, such as depth data from imaging systems or velocity from the DVL/ADCP. These filter can be rendered inoperable without these. Some filters are designed to input external localization information to help bound error accumulated by a continuously running solution. Depending on the regime of operation, external localization signals may not be available or difficult to acquire within operation. Some AUVs have limited onboard processing power which can constrain the amount and characteristics of individual filter solutions. Finally, many AUVs are equipped with filters that are a part of a proprietary

solution that cannot be altered by an operator to meet specific issues.

Under the circumstances of the AUV operations discussed, it becomes difficult to implement only one filter, and so two central questions arise:

1. How many and what type of filters can provide a more robust solution?
2. Do these filters interact and if so, how?

This thesis proposes are general framework that can be used with two simultaneous Bayesian filters that combines the two estimates to provide a more robust and more accurate position estimation in all regimes of an AUV mission. The questions raised in this thesis stem from work conducted at Lake Crescent, WA. Because of the lake size and depth the AUV experienced multiple changes in regimes in which a single filter solution was either degraded or completely impractical.

The AUV experimentation testing was conducted with Unmanned Underwater Vehicle Squadron One (UUVRON 1) in preparation for Arctic operations. The NPS REMUS 100 AUV had on board two separate filters for position estimation. A manufacturer specific EKF filter was used that provided a proprietary solution that performed as a DR model when the AUV was out of ADCP/DVL range. The second filter was a PF based on correlation with NOAA surveyed, lake bathymetry [11]. There are limitations with the DR solution in that this was a standalone estimate that would accrue large errors over time and could not be utilized in conjunction with any other non-manufacturer specific filter.

A significant limitation to the Lake Crescent mission was that there was no method by which each filter could interact with one another or fuse data to provide more accurate estimates. A federative process, an approach by which two or more filters interact and combine estimates, will be presented in this thesis as a solution to the second central question. Filters can interact and fuse estimates in several different ways. Filters can act in parallel where each is run simultaneously but provides only one aspect of a solution and does not mitigate filter specific limitations. They can act in series where one filtering algorithm operates within the other in order to reduce estimation error but is also unable to mitigate filter specific limitations. They can interact through different logic switches that can be used to help mitigate the limitations of each filter. Finally, filters can interact through a model that is able to intelligently determine when to switch and fuse each estimate and

which mitigates the limitations of each filter providing robust estimates in situations that can degrade single filter performance. The latter will be investigated in this thesis as it produces a more robust PNT solution within GPS denied environments.

## 1.2 Thesis Organization

The thesis is organized as follows: Chapter 2 provides a literature review with a description of the equipment, necessary background in Bayesian Filtering (BF) and Information Theory, and an overview of the field experimentation. Chapter 3 will introduce position estimation filter techniques and an overview of Interacting Multiple Model (IMM). Chapter 4 will discuss development of the model-based Extended Kalman Filter (EKF). Chapter 5 will discuss the development and simulation results of the Particle Filter (PF). Chapter 6 will discuss the development and simulation results of the Information Theoretic Interacting Multiple Model (IT-IMM). The thesis will conclude and discuss future work in chapter 7.

## 1.3 Contributions of This Thesis

- Introduction of a novel IT-IMM estimation that federates the model-based EKF and PF through the combined use of Shannon entropy from the posterior probability distribution and a measure of terrain suitability for predictive PF performance.
- A model-based EKF for estimating forward and sideslip velocities when the ADCP/DVL is not available.
- A PF algorithm that implements an information theoretic framework for particle redistribution.

# CHAPTER 2:
# Background

This thesis seeks to build on the work done by Darren Kurt and Douglas Horner, who implemented Monte Carlo localization (MCL) for use in Active Terrain-Aided Navigation (ATAN). ATAN combines reinforcement learning (RL) with TAN to greatly reduce the dependency on an *a priori* bathymetry map [12]. Their focus was on the problem of *exploration* versus *exploitation* in that the simulated AUV had to balance "the desire to explore all map areas while being able to exploit known areas in order to minimize the positional error" [5].

During times when the AUV is exploiting known areas on a map, there will be instances when the terrain is outside the range of the acoustic and imaging sensor systems. Therefore, this thesis will focus on two techniques for position estimation; a model-based EKF and a PF, that can provide robust estimates in many different circumstances. This thesis will also focus on the development of an IT-IMM algorithm to intelligently mix the position estimates. This chapter provides the background for remaining thesis chapters and includes a literature review, AUV equipment and sensor overview, an introduction to Generalized Bayesian Filtering (GBF), and field experimentation.

## 2.1 Related Works

Although there are many applications to the filters discussed in this thesis [13]–[24], this section covers literature review on position estimation filters and methods that are pertinent to this thesis. The review includes literature on the PF, model-based EKF, and IMM estimation.

### 2.1.1 Particle Filters

The PF is a state estimation algorithm that can be used when the dynamics of the observed system are non-linear and non-Gaussian. They, belong to the class of mean-field type interacting particle methodologies which are a class of Monte Carlo algorithms [25]. These algorithms are methods that use random sampling to obtain numerical results. Unlike other

filters, the strength of the PF is in its ability to handle these types of systems without requiring model assumptions. The limitation associated with this ability is that there is much higher computational complexity when compared to other methods.

One of first instances of the PF was in a statistical methodology proposed by Hammersley et al., in 1954 [26]. In the context of underwater navigation, the PF is a numerical method that utilizes a large number, on the order of up to the tens of thousands, of samples of a state vector according to its probability density function (PDF) [27]. Palmier presents a survey paper on PFs for underwater TAN with multi-sensor fusion and presents important concepts directly related to the unique challenge of TAN-based PF implementation [28].

An early implementation of the PF comes from a team at the Robotics Institute at Carnegie Mellon University conducted research for the International Conference on Robotics and Automation (ICRA) on MCL for mobile robotics. The team explored probabilistic approaches as they believed that these to be the most promising in providing a real-time localization solution. The team presented the MCL method using a PF which provided the ability to handle multi-modal distributions and thus the robot could globally localize its position, in real time, without any prior knowledge of its starting position. This was an improvement in accuracy, time, and required processing computation then earlier grid-based methods and provided the bases of the algorithm used in this thesis [29].

Gustafsson et al. implemented for the first time a MCL framework with a map matching technique for aerial navigation. They introduce non-linear models and non-Gaussian noise, which yield an improvement in estimation accuracy of the PF. Real-time implementation yielded accurate results comparable with satellite navigation [30]. Perez-Grau et al. improve upon MCL for aerial vehicles but more importantly, utilize visual odometry and a prior map which demonstrate strong system performance for navigation in a 3-D space [31]. As it applies to future work, Perez-Grau show that camera depth information can be used effectively within the PF algorithm.

Donovan describes autonomous AUV navigation using a PF framework without the use of GPS or LBL localization methods. The novelty of Donovan's work was the implementation of PF resampling which allowed the PF to "consistently recover a positional fix over a search area" and highlights the importance of re-sampling within MCL [32].

Karlsson and Gustafsson propose a PF for underwater navigation and implement the Cramér-Rao Lower Bound (CRLB), the lower bound an estimator must achieve to be considered fully efficient, along an experimental trajectory. Rao-Blackwellization is applied and was found to decrease the overall computational complexity of the filter [33], [34]. Alexander Bachmann and Stefan Williams at the University of Sydney propose a method for TAN in underwater environments using a PF and a digitized seafloor map with sonar measurements. In order to increase the performance of the PF, they used an adaptive particle sample size, which changed the size of the particle distribution depending on the state uncertainty [35].

Salavasidis et al. were one of the first to consider energy conversation in developing a PF-based TAN algorithm designed for a deep-rated, long-range, underwater navigation. Since energy-expensive, dense ranging, multi-beam sonar and high-grade INS use must be limited, the team implemented simple DR sensors such as an ADCP which they found provided sufficient measurements for PF implementation [36]. These works on PF-based TAN provided foundational knowledge in the development of the PF.

Osertman and Rhén propose a PF-based TAN algorithm using DVL altitude data fused with forward looking sonar (FLS) information to explore the sensor requirements for robust PF estimation. They conclude that dedicated TAN-sensors are undesirable and that accurate PF estimates can be made with common on-board equipment such as a DVL and FLS [37]. Lager et al also highlight the effectiveness of utilizing affordable sensor systems, both navigational and perceptive, as well as publicly available charts and bathymetry maps within a robust PF-based TAN algorithm with sensor fusion [38]. The work done on sensor fusion provides important insight as the PF will be developed to be utilized onboard an AUV without dedicated TAN imagery sensors.

### 2.1.2 Model-based Extended Kalman Filters

The EKF is also a state estimation algorithm that uses a Taylor Series expansion to extend the Kalman Filter (KF) to non-linear process and observation models. The EKF, belongs to a family of optimal estimators for systems in which the model dynamics are well known. The strength of the filter, however, is the ability to make state estimates with much lower computational complexity when compared to PFs giving it an advantage when an AUVs processing power is limited.

A model-based EKF is one that utilizes an accurate model of the system dynamics. Dardanelli shows that such an approach can more accurately estimate state variables [39]. Ranjan et al. use a model-based EKF to merge altitude, velocity, heading, and GPS information in order to produce a single state vector utilizing a sensor model that helps reduce the overall error [40]. As it pertains to this thesis, a hydrodynamic model will be developed for estimating forward and sideslip velocities that provides more accurate estimates when the ADCP/DVL is not available.

There has been a tremendous amount of research into the development of hydrodynamic models for undersea vehicles. Fossen developed the 6 degrees of freedom (DOF) model for undersea vehicles, of which many of the simplified equations will be used in this thesis. Work done by Taylor [41] builds an accurate 6DOF hydrodynamic model of a Remote Environmental Monitoring UnitS 100 (REMUS) 100 AUV which will be the platform for testing within this thesis. Prestero and Sgarioto developed the hydrodynamic coefficients and equations from which the hydrodynamic model is formulated [42], [43].

### 2.1.3 Interacting Multiple Model Estimation

The IMM approach, or Classical Interacting Multiple Model (C-IMM), developed by Bar-Shalom and Li, is a sub-optimal model for system estimation for combining together two or more filtering methods into a single estimate [44]–[49]. C-IMM has seen success when combined with a PF for state estimation. Li and Dou utilize a C-IMM with two PFs for visual tracking based on multiple switching cue states of a dynamic system and blends three models: Corrected Background Weighted Histogram (CBWH), Completed Local Ternary Patterns (CLTP) and Histogram of Oriented Gradients (HOG) [50]. Kim and Jeong developed a tracking algorithm utilizing IMM PF that can process crash probability data as *a priori* knowledge and use that with Monte Carlo simulations to track vehicles [51].

Genovese provides a robust C-IMM algorithm that fuses five filter models and shows that the overall performance of the state estimates improved as the complexity and number of the filter models increased. This work provides significant insight into the fundamental algorithm of the C-IMM [52]. Johnston and Krishnamurthy developed an Re-weighted Interacting Multiple Model (R-IMM) which is a recursive implementation of a *maximum a posteriori* state sequence and presented as an improvement in computational speed over the

C-IMM [53]. Li and Jia develop an IT-IMM approach for combining state estimates which this thesis will use [54].

## 2.2 Equipment

This section provides an overview of the Naval Postgraduate School (NPS) REMUS 100 AUV and its associated sensor systems that was used during field experimentation at Lake Crescent, WA, in December 2022.

### 2.2.1 Remote Environmental Monitoring UnitS 100

The NPS REMUS is a modified REMUS 100 AUV with fore and aft cross-tunnel thrusters, which permit hovering, translation, and rotation. It has the flexibility to add module sensors on the front section of the vehicle and features a backseat driver architecture for testing new autonomy concepts. The backseat driver provides commands to the main vehicle control system which executes the commands. A description of the system and its capabilities are shown in Table 2.1.


Figure 2.1. REMUS 100 Vehicle. Source: [5]

Work by Eric Bermudez [55] and Ian Taylor [41] on the hydrodynamic characteristics of a 6 DOF model were adapted for this thesis to model the movement of the REMUS to improve the accuracy of the EKF. The equations and coefficients adapted from Bermudez and Taylor will be discussed in more detail in Chapter 4. Table 2.1, adapted from [5, Table 1.1] and shown below, was adapted from the work done by Darren Kurt [5].

Table 2.1. REMUS Configuration Parameters. Source: [5]

| **Part/Module** | **Used Setup/Utilized Implementation** |
|---|---|
| Robot | REMUS 100. Length 2.28 m. Diameter 0.19 m. Weight 40 kg. Speed 0.25-2.57 m/s. Depth range 3 m-100 m. Battery duration 9 hours [56]. |
| Propulsion | Direct drive DC brush-less motor directly connected to open three bladed propeller [56]. |
| Locomotion Control | (2) coupled yaw and pitch fins. CTT [56]. |
| INS | SeaDeViL INS navigational uncertainty equal to 0.43 percent distance traveled [56]. |
| ADCP | 900 kHz, four downward-looking, transducers measure forward velocity, side velocity, and heave of vehicle [56]. |
| Sonar | BlueView MB2250-N downward-looking, ultra-high resolution sonar |
| 3DX Camera | Embedded IMU and GPU. NVIDIA's Jetson TX2 embedded AI-engine. Built-in ROS based data and control interfaces. An on-board RUI for local setup and management as well as NEPI for remote data and system management over a secure smart IoT Backbone [57]. |
| Computational Resources | (2) intel Core i9 processors for the secondary controller and sensor processing. (2) Hydroid CPUs and hard drives for the main controller and the side scan sonar. |

### 2.2.2 Sensors

Pertinent to this thesis, the modified REMUS utilizes a Kearfott SeaDeViL INS system, and either the BlueView MB2250 FLS, the Numurus 3DX camera system or the ADCP/DVL for TAN.

### 2.2.3 Kearfott SeaDeViL INS

The Kearfott SeaDeViL INS is an integrated system and merges accelerometer sensors, the ADCP/DVL, and GPS into a pose estimate [55]. While surfaced, this system uses GPS to obtain position and velocity estimates. While submerged, the system uses the ADCP/DVL to obtain the speed over ground of the REMUS. A description of the system specifications are shown in Table 2.2:

Table 2.2. Kearfott SeaDeViL INS System. Adapted from: [55]

| **Attribute** | **Specification** |
|---|---|
| Size | 8850 $m^3$ |
| Weight | < 7.25 kg |
| Power | < 30 watts Vdc |
| **Operational Range** | |
| Acceleration | > 30 g's |
| Attitude | Unlimited |
| Attitude Acceleration | > 10,000 $deg/s^2$ |
| Temperature | -40°C to +55°C |
| **Performance** | |
| Position Accuracy | 0.5% DT CEP |
| Heading Accuracy | 5.0 mils RMS |
| Roll/Pitch | 0.5 mils RMS |

**BlueView MB2250 Sonar**

The BlueView MB2250 sonar is mounted to the REMUS via a "wet-mateable, power ethernet connection" [56]. The sonar mounted to REMUS is the BlueView 900 kHz FLS and BlueView 2250 kHz downward-looking, ultra-high resolution sonar. Each system employs methods that "turn a single acoustic signal into a larger swath of beams" [58]. A description of MB2250 sonar is shown in Table 2.3, from [56] .

Table 2.3. BlueView MB2250 Sonar Package. Source: [5]

| **Attribute** | **Specification** |
|---|---|
| FOV | 76°x1° |
| Minimum Range | 0.5 m |
| Maximum Range | 10 m |
| Beam Width | 1°x1° |
| Number of Beams | 256 |
| Horizontal Sonar Angles | [-30°, -15°, 0°, 15°, 30°] |
| Vertical Sonar Angles | [-15°, -7.5°, 0°, 7.5°, 15°] |
| Max Update Rate | 40 Hz |
| Frequency | 2.25 MHz |

Figure 2.2, from [5], shows an example of a BlueView MB2250 downward-looking image.

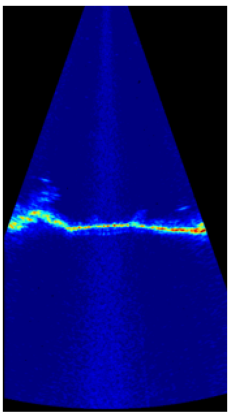


Figure 2.2. BlueView MB2250 Sonar image taken from the NPS REMUS 100. The light blue indicates the ocean floor. Source: [5]

### 2.2.4 Numurus 3DX-CI-1K-MKS Camera System

The Numurus 3DX-CI-MKS Camera System, shown in Figure 2.3.

Figure 2.3. Numurus 3DX-CI-MKS Camera System

The 3DX is mounted as a downward-facing unit on the REMUS and can provide imagery, depth maps, and point clouds [59]. An example from data collected at Lake Crescent is shown in Figure 2.4.

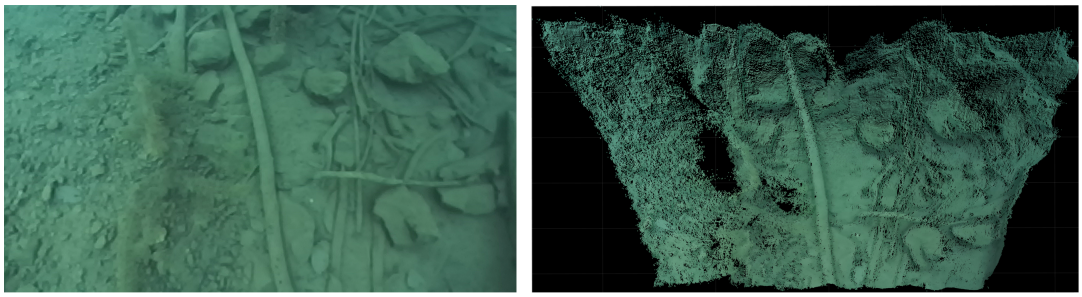


Figure 2.4. Camera imagery (left) and Point Cloud Image (right).

The 3DX sensors combine 2D/3D imaging/mapping within an IMU and GPU that is housed with a compact underwater housing. This system gives the REMUS the ability for on-board bathymetric data collection to be used within the PF while reducing the overall acoustic footprint of the system. The 3DX uses the ROS sensor message type PointCloud2. The message contains a collection an N-dimensional points which is organized as either a 2D or 1D collection of data. A system description for the 3DX is provided in Table 2.4.

Table 2.4. Numurus 3DX-CI-1K-MKS Camera Specifications. Source: [57]

| **Hardware** | **Specification** |
|---|---|
| Sensor Angles | Air: 90° (H) x 60° (V), Water: 67° (H) x 45° (V) |
| Image Resolution | 2.2K/1080p/720p/VGA |
| Image Rates | Up to 20 fps (Resolution bandwidth dependent) |
| Depth Range | 0.1 - 15 m |
| Point Cloud Rates | Up to 30 fps (Resolution Dependent) |
| LEDs | Up to 3500 Lumens |
| IMU | Translation 1.0% Rotation 0.013°/m |
| Internal Storage | 256 GB SD Card |
| Interface I/O | Ethernet 100 or 1000 Gbps |
| Supply Voltage Range | 10 - 36 VDC |
| Power Estimates | 12 - 17 W |
| Connector Options | Impulse MKS 310 (100 Mbits/s) SubConn DBCR2013M (GigE) |
| Housing Dimensions | 18 cm x 143 cm x 76 cm |
| Weight in Air/Water | 1.11 kg/0.363 kg |
| Depth Rating | 1 km |
| Operating Range | -20°C to +65°C |

## 2.3 State Space Representation

The following information provides the state space representation that will be implemented into the filters developed in this thesis. This provides the dynamics of our system as a set of first-order differential equations with state variables and algebraic equations [60]. The state space model, given time $k$, is presented as:

$$\dot{\boldsymbol{x}}(k) = A\boldsymbol{x}(k) + B\boldsymbol{u}(k) + \boldsymbol{\omega}(k), \tag{2.1}$$

$$\boldsymbol{z}(k) = H\boldsymbol{x}(k) + D\boldsymbol{u}(k) + \boldsymbol{\upsilon}(k), \tag{2.2}$$

where

- $A$ is the state matrix
- $B$ is the input matrix
- $H$ is the output matrix
- $D$ is the direct transmission matrix
- $\boldsymbol{x(k)}$ is the vehicle position and orientation state vector
- $\boldsymbol{u(k)}$ is the control vector using rudder deflection
- $\boldsymbol{z(k)}$ is the measurement vector
- $\boldsymbol{\omega(k)}$ is the zero mean, white process noise $\sim \mathcal{N}(0, \mathcal{Q})$
- $\boldsymbol{\upsilon(k)}$ is the zero mean, white process noise $\sim \mathcal{N}(0, \mathcal{R})$

The state space model is linear time invariant (LTI) so at any time $k$ the input relates to the output. The block diagram for the state space representation is shown in Figure 2.5.

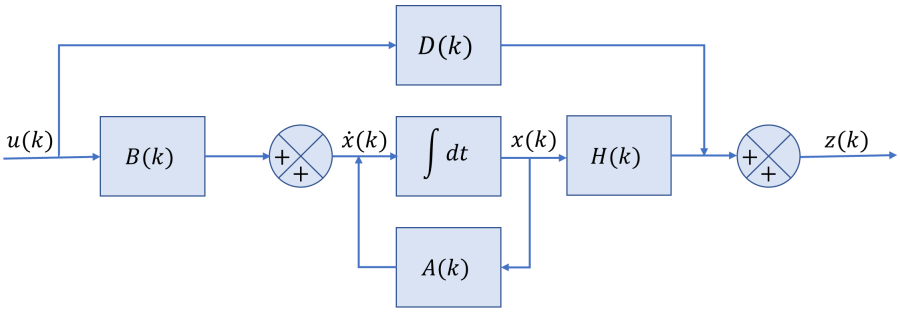



Figure 2.5. State space model block diagram. The state vector is a stochastic variable representing the estimate of the state of the vehicle. $A$ relates the state vector to the state estimate. $D = 0$ for this case as it is a feed through term. $H$ relates the state vector to the measurement vector and, if only one filter is implemented, must be able to change over time to accommodate different measurements.

The state space of a linear system is represented by Figure 2.5. Since the dynamic system is LTI and finite-dimensional then the algebraic equations can be written in matrix form. This will be the model implemented within this thesis. The input matrix and control vector are significant to the implementation of the state space model to the model-based EKF. This pair will provide the hydrodynamic changes of the REMUS due to the deflection in the rudder. Specific implementation will be discussed in further detail in Chapter 4.

## 2.4 Generalized Bayesian Filtering

This section describes Bayesian techniques that underlie the remaining thesis for position estimation and Optimal Spatial Estimation (OSE). In prior ATAN work, the OSE is used to calculate a stochastic estimate of the terrain. It permits the ability to plan dynamic, optimal paths based on evaluation of rewards associated with *exploration* (conducting the main coverage mission) and *exploitation* (using good undersea features for position localization) [5].

### 2.4.1 Position Estimation

The prediction, Equation 2.3, determines the probability of a vehicle's state, $\boldsymbol{x_k}$, given previous measurements as a function of time.

GBF uses the *a priori* probability distribution $p(\boldsymbol{x_{k-1}}|\boldsymbol{Z_{k-1}})$ and a state transition model represented by $p(\boldsymbol{x_k}|\boldsymbol{x_{k-1}}, \boldsymbol{u_k})$. The prediction is then used by the correction step, Equation 2.4, along with the measurement probability given all previous measurements, represented by $p(z_k|\boldsymbol{Z_{k-1}})$, as well as the current measurement probability given the current state, represented by, $p(\boldsymbol{x_{k-1}}|\boldsymbol{Z_{k-1}})$ [56]. The GBF approach are represented by:

$$p(\boldsymbol{x_k}|\boldsymbol{Z_{k-1}}) = \int p(\boldsymbol{x_k}|\boldsymbol{x_{k-1}}, \boldsymbol{u_k}) p(\boldsymbol{x_{k-1}}|\boldsymbol{Z_{k-1}}), \tag{2.3}$$

$$p(\boldsymbol{x_k}|\boldsymbol{Z_k}) = \frac{p(z_k|\boldsymbol{x_k}) p(\boldsymbol{x_k}|\boldsymbol{Z_{k-1}})}{p(z_k|\boldsymbol{Z_{k-1}})}, \tag{2.4}$$

where:

- $p(\boldsymbol{x_{k-1}})$ is the probability of the vehicle pose at time $k-1$.
- $p(\boldsymbol{x_k}|\boldsymbol{x_{k-1}}, \boldsymbol{u_k})$ is the state transition probability.
- $p(z_k|\boldsymbol{x_k})$ is the measurement probability, observing $z_k$ at state $\boldsymbol{x_k}$
- $\boldsymbol{Z_k} = \{z_i : i = 1, ..., k\}$ is the set of all measurements until time $k$.

Although this general framework for state estimation cannot be used directly, there are recursive solutions such as the model-based EKF and PF that will be developed in this thesis for position estimation that adhere to this Bayesian approach.

### 2.4.2 Optimal Spatial Estimation

Optimal Spatial Estimation (OSE) is used by ATAN to build a stochastic "terrain map based on the sensor measurements" of the REMUS [61]. This process of OSE is a technique that can be used to map a bathymetric surface using only a limited number of points [61]. Tydingco uses Kriging, a method of spatial interpolation of a scattered set of data around an unknown point to produce an optimal estimate, within OSE to improve terrain estimation for use within TAN.

Underlying the approach is a Bayesian estimator of the semivariogram function that is combined together with the distance of the measurements from the estimation point to determine the influence of the measurement on the overall estimate. The advantage of Kriging for OSE is that it provides a variance estimate for each point of interest in the stochastic map. [61].

The prediction of the estimation point is represented by:

$$C^* = \sum_{i=1}^{N} \boldsymbol{\lambda}_i C(s_i), , \tag{2.5}$$

where $N$ is the number of observed values, $\lambda_i$ is the weighted value at the $i^{th}$ location, and $Z(s_i)$ is the measured value at the $i^{th}$ location [61]. In order to determine Kriging weights the relationship, $K\lambda = \boldsymbol{k}$ is used where $K$ is the matrix covariance between data points, $\boldsymbol{k}$ is the vector covariance between data points and the estimation points, and $\boldsymbol{\lambda}$ is the Kriging weight vector for known data points [61]. $K$ and $\boldsymbol{k}$ are represented as:

$$K = \begin{bmatrix} \lambda\left(d\left(c_1, c_1\right)\right) & \cdots & \lambda\left(d\left(c_1, c_i\right)\right) \\ \vdots & \ddots & \vdots \\ \lambda\left(d\left(c_i, c_1\right)\right) & \cdots & \lambda\left(d\left(c_i, c_i\right)\right) \end{bmatrix}, , \tag{2.6}$$

$$k = \begin{bmatrix} \lambda\left(d\left(c_1, c^*\right)\right) \\ \vdots \\ \lambda\left(d\left(c_i, c^*\right)\right) \end{bmatrix}, \tag{2.7}$$

where $K$ is an $n \times n$ matrix, based on the number of points used in estimation, $\lambda$ can be derived from the semivariogram, $c^*$ is the estimation point, and $d\left(c_i, c_j\right)$ is distance between two points.

Once the Kriging weights have been calculated, the prediction can be made:

$$C = \sum_{i=1}^{N} C(s_i). \tag{2.8}$$

This process continues until the map is fully known. As it applies to this thesis, it is assumed that the IMM developed in this thesis is running on a fully known terrain map and can therefore have an expectation of the information within that terrain.

## 2.5 Information Theory

This thesis will develop a PF using Shannon Entropy for re-distribution of its particle population as well as an IT-IMM that will use Shannon and Boltzmann Entropy for determining when to shift between filter estimates.

### 2.5.1 Shannon Entropy

Information entropy was first introduced by Claude Shannon in 1948 and describes the "average of uncertainty of a random variable as the entropy, or information, which are characteristic to the variable's possible outcomes" [62]. This is also referred to as Shannon Entropy, $H$. Given a discrete random variable, $X$, the entropy can be defined below where $x_1, ..., x_n$ represent the possible outcomes that occur with probability $P(x_1), ..., P(x_n)$ [62]:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i). \tag{2.9}$$

This thesis seeks to implement the Shannon entropy of the particle distribution as a metric to inform state estimation combination of the IT-IMM.

### 2.5.2 Boltzmann Entropy

The work by Darren Kurt [5] explored Boltzman Entropy (BE) and how it could be used for the classification of terrain. Kurt showed that BE is directly proportional to the variability of the ocean floor that a sensor can detect. BE was shown to compute the number of permutations of a sample and can be defined as [5]:

$$W = k_b \log(W), \tag{2.10}$$

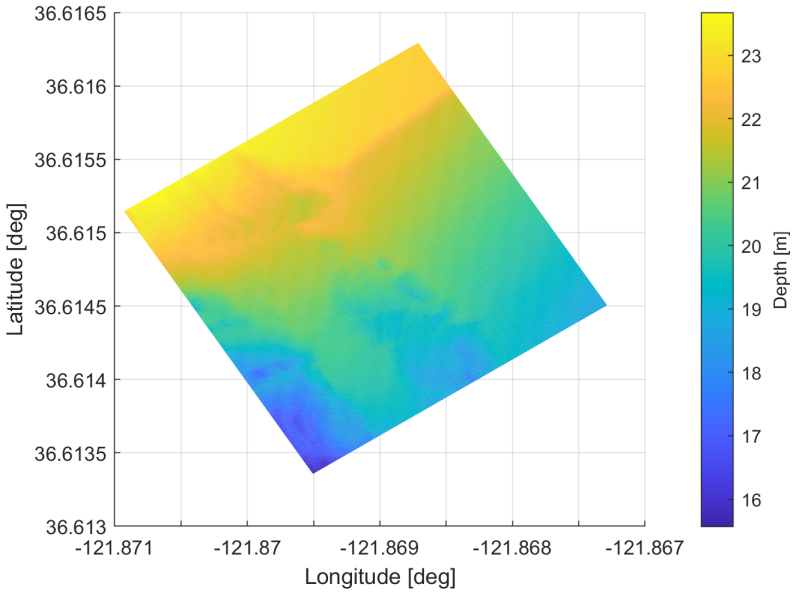
where $W$ are the number of permutations and $k_b$ is the Boltzmann constant. In Kurt's work, as in this thesis, the BE at position $\xi(m_n)$, is proportional to the number of permutations:

$$\xi(m_n) \propto \log(W). \tag{2.11}$$

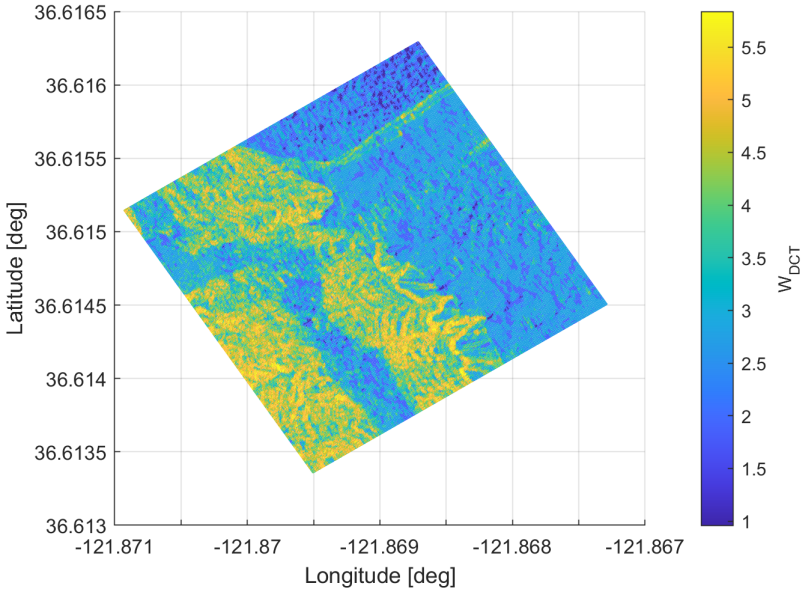
The two-dimensional discrete cosine transform (DCT) is calculated from a small section of the bathymetry map and then assigned as a part of the particle distribution. The DCT expresses the bathymetry data as a sum of the sinusoidal functions with different frequencies and amplitudes and, like a discrete Fourier transform (DFT), the DCT can only operate as a function of discrete data points [63]. The 2-D DCT or DCT-II is defined over the domain for $k = 0, ..., N - 1$:

$$X_t = \sum_{n=0}^{N-1} x_n cos[\frac{\pi}{N}(n + \frac{1}{2})t]. \tag{2.12}$$

Figure 2.6, from [5], shows a terrain map and DCT-BE map.




(a)




(b)

Figure 2.6. Ocean Floor Bathymetry Map (a) and Boltzmann Entropy Map (b). Source: [5]

As shown, significant changes within the terrain yield much higher DCT-BE values. Terrain that shows little to no changes in topology yields significantly lower values of DCT-BE. Values of 6 are considered high DCT-BE while values close to 0 are considered very low. This highlights the significance of the DCT-BE map and the motivation for its use within this thesis.

### 2.5.3 Kullback–Leibler Divergence

The Kullback-Leibler Divergence (KLD) is a part of Ali-Silvey class information theoretic measures [64] and is used to compute the difference of mutual information between two probability distributions [65]. The KLD can be used to determine which probability distribution best approximates a candidate probability distribution by minimizing the KLD [54]. The KLD between distribution $p(x)$ and $q(x)$ is defined as:

$$D_{KL}(p||q) = \int_{\mathbb{R}^n} p(x) \log \frac{p(x)}{q(x)} dx, \tag{2.13}$$

where $D_{KL}(p||q)$ is the divergence of $q(x)$ from $p(x)$ over $\mathbb{R}^n$.

Li and Jia utilize a weighted KLD defined given $N$ probability density functions $p^i(x)$ with relative weights $\lambda_i$ which satisfies [54]:

$$\sum_{i=1}^{N} \lambda_i = 1, \quad \lambda \geq 0, \tag{2.14}$$

with a weighted Kullback-Leibler average (KLA), $\overline{p}(x)$ defined as:

$$\overline{p}(x) = \arg \inf_{p \in \mathcal{P}} \sum_{i=1}^{N} \lambda_i D_{KL}(p||p^i), \tag{2.15}$$

where the set of probability density functions, $\mathcal{P}$, are over $\mathbb{R}^n$.

Li and Jia utilize a weighted KLD as the metric by which the IT-IMM algorithm switches from different filters.

## 2.6 Field Experimentation

Field experimentation and data collection occurred at Lake Crescent, Olympic National Forest, Port Angeles, Washington in 2022 shown in Figure 2.7 [11]. Data was collected on-board the REMUS which will be the focus in this thesis.

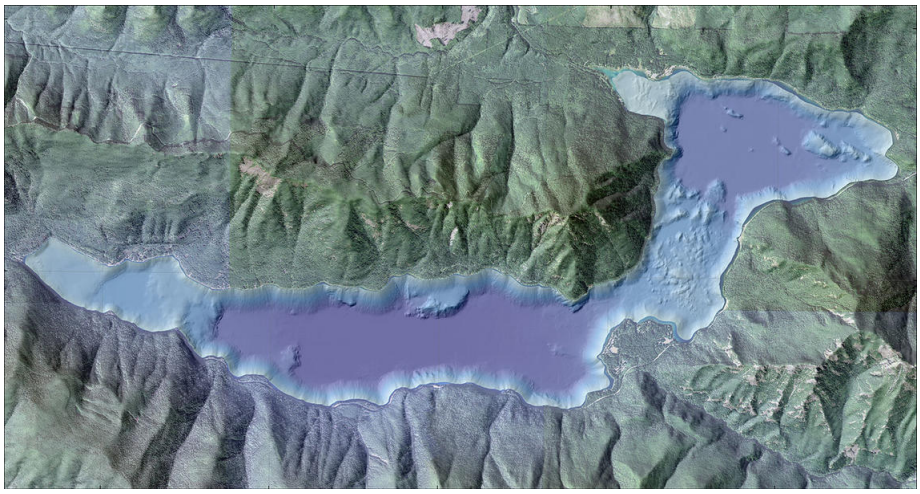


Figure 2.7. Bathymetric Map of Lake Crescent. Source: [11]

There were several different operational areas (OPAREA) chosen within the lake, shown in Figure 2.8. The centrally located OPAREA was chosen due as it provided a similar mission format to under-ice-conditions. The southern OPAREA was chosen for PF testing as the depth of water were within the constraints of the vehicle. Finally, the northern OPAREA was chosen for 3DX camera experimentation. Bathymetric ground truth was provided by a survey conducted by the United States Geological Survey (USGS) [11].

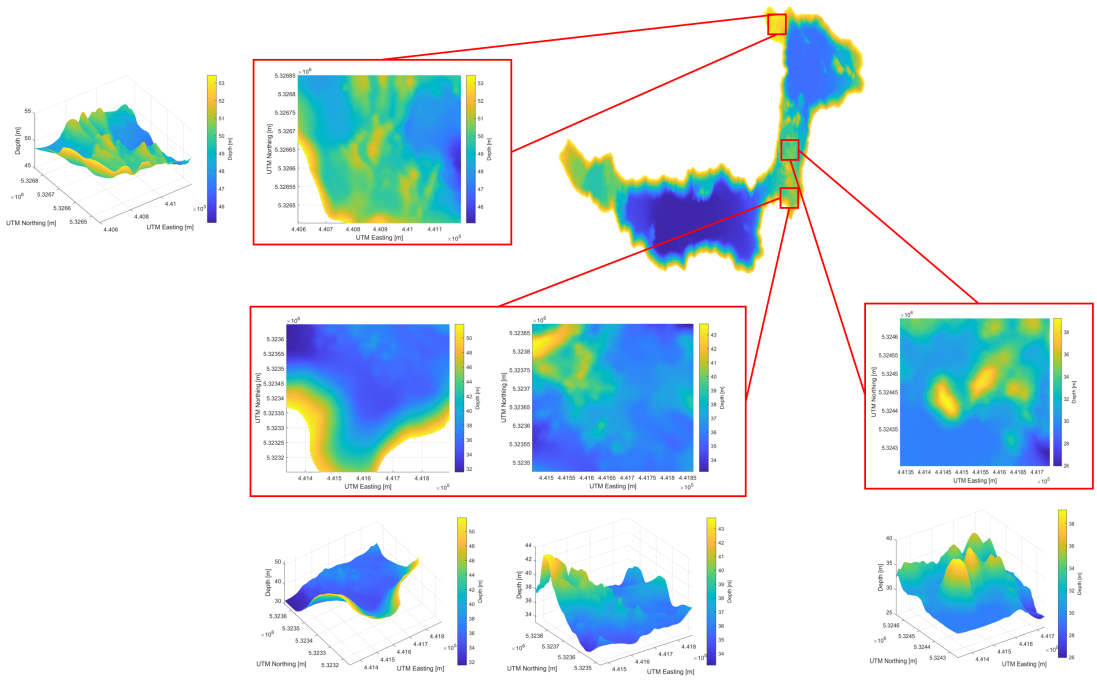


Figure 2.8. Lake Crescent Area of Operation

For further discussion in Chapter 4, the most northern OPAREA will be designated OA1, the mid area as OA2, and the most southern OPAREA as OA3.

Data was collected from the NPS REMUS over several runs, shown in Appendix A. This data included all available state information outputs from the REMUS conductivity temperature and depth (CTD) sensor, the ADCP/DVL, and the Kearfott SeaDeViL INS. The data collected from the missions at Lake Crescent will be used as inputs into the filters developed in this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

# CHAPTER 3:
# Review of Position Estimation Filters

This chapter provides an overview on position estimation filters that highlight techniques for increasingly non-linear, non-Gaussian estimation problems. The computational complexity of these algorithms, however, increases as the dynamics of the systems that they observe become more non-linear and non-Gaussian. These filters are the core selection of estimation techniques that can be used within a Markov jump process and include the KF, EKF, Unscented Kalman Filter (UKF), and PF. In relation to this thesis, the goal is to develop an EKF and PF and a method of switching between each.

## 3.1 Kalman Filter

The KF is an iterative process that uses consecutive inputs in order to estimate the true state of a vehicle when the state data contains uncertainties. The KF algorithm follows four main steps [66]:

- State Extrapolation.
- Error Covariance Extrapolation.
- Kalman Gain Matrix calculation.
- State Estimate and Error Covariance Update.

The KF operates on several significant assumptions; that the dynamics of the true state are linear, and that the probability distribution is Gaussian. Specifically the process noise and the measurement noise are considered to be zero-mean Gaussian ($\omega_k \sim \mathcal{N}(0, \mathcal{Q}_{k|k})$, $\upsilon_{k|k} \sim \mathcal{N}(0, \mathcal{R}_{k|k})$) [67].

The general discrete KF algorithm, shown in Algorithm 1 [66], follows two stages: a prediction stage in which the state estimate and error covariance are predicted, and an update stage at which the Kalman gain is calculated and the state and its error covariance are updated.

The goal of the KF is to estimate of the posterior state vector at time $k+1$, $\hat{\boldsymbol{x}}_{k+1|k+1}$ [66]. The nomenclature that will be used for the remainder of this thesis reflects Bayesian conditional

probability. This notation is a common way to show the conditional probability as the output for the update. It is important to note that $\boldsymbol{\omega}_k$ will also use this subscript notation. This could be deceiving as $\boldsymbol{\omega}$ will hold constant, however, has the flexibility to change. This is important because of the fact that the IMM algorithm can estimate what the error parameters are without prior knowledge.

**State Extrapolation**

The KF first extrapolates the state estimate, $\boldsymbol{x}_{k+1|k}$, using the state transition matrix, $A$, the control-input matrix $B$, the control vector $\boldsymbol{u}_k$ and the previous estimate $\hat{\boldsymbol{x}}_{k|k}$.

$$\boldsymbol{x}_{k+1|k} = A\hat{\boldsymbol{x}}_{k|k} + B\boldsymbol{u_k} + \omega_k. \tag{3.1}$$

The measurement state vector is represented by:

$$\boldsymbol{z}_{k|k} = H\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{\upsilon}_k, \tag{3.2}$$

where $H$ is the measurement transition matrix.

**Error Covariance Extrapolation**

Using the previous posterior error covariance matrix, $P_k$, $A$, and $A$ transpose, along with covariance of the process noise $Q_k$, the error covariance matrix $P_{k+1|k}$ can be found.

$$P_{k+1|k} = AP_kA^T + Q_k. \tag{3.3}$$

**Kalman Gain Matrix Calculation**

The Kalman Gain matrix $K_k$ is then calculated using $P_{k+1|k}$, $H_k$ and its transpose, and the covariance of the measurement noise $\mathcal{R}_k$.

$$K_k = P_{k+1|k}H_k^T[H_kP_{k+1|k}H^T + \mathcal{R}_k]^{-1}. \tag{3.4}$$

**State Estimate and Error Covariance Update**

Finally, the state estimate and error covariance matrix are updated using the measurement vector $z_k$ [66].

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \boldsymbol{x}_{k+1|k} + K_k[\boldsymbol{z}_k - H_k\boldsymbol{x}_{k+1|k}], \tag{3.5}$$

$$P_{k+1|k+1} = [I - K_k H_k]P_{k+1|k}. \tag{3.6}$$

**Algorithm 1** Kalman Filter

1: **Input**: $\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{z}_k, \boldsymbol{u}_k$
2: $\boldsymbol{x}_{k+1|k}$ = **state extrapolation**$(A, B, \hat{\boldsymbol{x}}_{k|k}, \boldsymbol{u}_k)$
3: $P_{k+1|k}$ = **error covariance extrapolation**$(A, P_k, Q_k)$
4: $K_k$ = **Kalman gain**$(P_{k+1|k}, H, \mathcal{R}_k)$
5: $\hat{\boldsymbol{x}}_{k+1|k+1}$ = **state est. update**$(\boldsymbol{x}_{k+1|k}, K_k, \boldsymbol{z}_k, H_k)$
6: $P_{k+1|k+1}$ = **error covariance update**$(I, K_k, H_k, P_{k+1|k})$
7: **Output**: $\hat{\boldsymbol{x}}_{k+1|k+1}, P_{k+1|k+1}$

The complexity of an algorithm is the amount of memory, time, and computational resources required to run it. The complexity of an algorithm can be found by analyzing the complexity of each step within it. The *state prediction* consists of multiplying two matrices, $m \times n$ and $n \times p$, and adding two $m \times n$ matrices. The complexity for each as $O(np)$ and $O(mn)$ respectively. The total time is therefore, $O(2n^2 + 3n)$ [68].

The *error covariance prediction* consists of multiplying two $n \times n$ matrices which run at $O(n^2.376)$. A transpose operations runs at $O(n^2)$ and with the addition of a matrix the total time is therefore, $O(2n^{2.376} + 2n^2$. The *Kalman Gain* can be represented in the same manner as $O(5n^{2.376} + 3n^2)$. The *state estimate update* is represented as $O(2n^2 + 2n)$ and the *error covariance update* as $O(2n^{2.376} + n^2)$ [68].

Therefore the total time complexity of the KF is represented as $O(9n^{2.376} + 10n^2 + 5n)$ [68].

The KF is a simple algorithm that is appropriate when the representation of the errors are

linear and Gaussian. In many situations, especially those associated with underwater TAN these assumptions are not always true. This is the most significant limitation of the KF, that when a systems dynamics are non-linear, the KF will not produce accurate estimates. The advantage of this filter is the computational simplicity which make it favorable in systems with limited processing power whose dynamics are known to be linear.

## 3.2 Extended Kalman Filter

The formulation of the EKF is almost identical to the KF with the exception that the system and measurement models are non-linear. Since the EKF can handle non-linear state dynamics the system and measurement model become differential functions. Since $f$ and $h$ cannot be directly applied to the covariance, the Jacobians $F$ and $H$ are computed at each time step $k$ [69]. The equations governing the EKF are a non-linear implementation of the equations of the KF.

**State Extrapolation**

$$\boldsymbol{x}_{k+1|k} = f(\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{u}_k) + \boldsymbol{\omega_k}, \tag{3.7}$$

$$\boldsymbol{z}_{k|k} = h(\hat{\boldsymbol{x}}_{k|k}) + \boldsymbol{\upsilon}_k, \tag{3.8}$$

$$F = \left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{u}_k}, \tag{3.9}$$

$$H = \left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k|k}}. \tag{3.10}$$

**Error Covariance Extrapolation**

$$P_{k+1|k} = F_k(\hat{\boldsymbol{x}}_{k|k})P_k + F_k^T(\hat{\boldsymbol{x}}_{k|k}) + Q_k. \tag{3.11}$$

**Kalman Gain Matrix Calculation**

The other significant difference in the EKF, shown in Algorithm 2, is that innovation covariance, $S_k$, is calculated to be utilized in the computation of the *near optimal* Kalman gain [69]. This step will be done within the Kalman Gain Matrix calculation in Algorithm 2.

$$S_k = H_k(\hat{\boldsymbol{x}}_{k|k})P_{k+1|k}H_k^T(\hat{\boldsymbol{x}}_{k|k}) + \mathcal{R}_k, \tag{3.12}$$

$$K_k = P_{k+1|k}H_k^T(\hat{\boldsymbol{x}}_{k|k})S_k^{-1}. \tag{3.13}$$

**State Estimate and Error Covariance Update**

The state estimate and covariance updates remain the same but now a function of the non-linear state.

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + K_k[\boldsymbol{z}_k - h_k(\hat{\boldsymbol{x}}_{k+1|k})], \tag{3.14}$$

$$P_{k+1|k+1} = [I - K_k H_k(\hat{\boldsymbol{x}}_{k+1|k})]P_{k+1|k}. \tag{3.15}$$

**Algorithm 2** Extended Kalman Filter

1: **Input**: $\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{z}_k, \boldsymbol{u}_k$
2: $\boldsymbol{x}_{k+1|k}$ = **state extrapolation**$(\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{u}_k)$
3: $P_{k+1|k}$ = **error covariance extrapolation**$(F_k, P_k, Q_k)$
4: $K_k$ = **Kalman gain calculation**$(P_{k+1|k}, H_k, S_k^{-1})$ ▷ *near optimal*
5: $\hat{\boldsymbol{x}}_{k+1|k+1}$ = **state update**$(\hat{\boldsymbol{x}}_{k+1|k}, K_k, \boldsymbol{z}_k)$
6: $P_{k+1|k+1}$ = **error covariance update**$(\hat{\boldsymbol{x}}_{k+1|k}, I, K_k, H_k, P_{k+1|k},)$
7: **Output**: $\hat{\boldsymbol{x}}_{k+1|k+1}, P_{k+1|k+1}$

As the algorithm of the EKF follows the algorithm for the KF, so too does it complexity. The key difference within its complexity, however, are that the non-linear functions $f$ and $h$ are system dependent. This means that it cannot be exactly stated

what the computational complexity is, rather that the complexity can be represented by $O(n^{2.376}, O(\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{u}_k), O(z_{k+1|k})$ [68].

As the dynamics of the system move from linear to non-linear, as in the case of the dynamics of an AUV model, the EKF becomes a more appropriate algorithm. Although more complex than the KF, due to the calculation of system dependent Jacobians, the EKF can still provide state estimations with a favorable level of computational complexity. The limitation associated with the EKF is that is linearizes about a non-linear system which can still produce inaccurate estimates if the dynamics become highly non-linear.

## 3.3 Unscented Kalman Filter

When the dynamics of the system become more highly non-linear, the EKF can have poor performance due to the fact that the covariance is propagated through linearization of the non-linear model [70]. An UKF is a filter technique which addresses this limitation by using a deterministic sampling function that picks a minimal set of sigma points (sample points) around a mean. This technique is called the unscented transform (UT) [70]. These sigma points are propagated through $f$ and $h$ and a new mean and covariance estimation is calculated.

Julier et al. discuss that the sigma points should be chosen so that they "capture the same mean and covariance irrespective of the choice of matrix square root which is used" [70]. The UKF bears a resemblance to a PF in that sigma points are a similar concept to particles. The underlying difference is that while particles are drawn at random, sigma points are drawn according to a deterministic algorithm [70] not discussed in this thesis. The UKF is available at [70] and will not be discussed in subsequent chapters of this thesis. Although not used, it is provided for completeness of the continuum of position estimation filters. An outline of the algorithm UKF is as follows [71]:

- Define System and Measurement model.
- Sigma Point Selection and Transformation
- State Estimate and Error Covariance extrapolation
- Second Sigma Point Selection and Transformation
- Kalman Gain Matrix calculation

- State Estimate and Error Covariance Update

A more complete discussion on sigma points can be found in *A New Extension of the Kalman Filter to Nonlinear Systems* [70].

An advantage of the UKF is that it removes the need to explicitly calculate Jacobians which can be costly due to the nature of some complex functions. A drawback, however stems from the use of UT in that the filters performance is highly dependent on how exactly the transformed information of the UT is calculated as well as which sigma points are used [70]. Although more computationally efficient than the PF, a limited number of sigma points give a limited representation and understanding of the probability density function.

## 3.4 Particle Filter

Unlike the previous filters, the PF is not limited by assumptions of the model dynamics and can be used for estimation within a much wider range of filtering problems. The PF uses a distribution of particles, as a candidate probability distribution for the true state of a system. Due to this flexibility, the PF can be used for non-linear, non-Gaussian state estimation.

A general PF algorithm is given in Algorithm 3 and follows the general steps:

- Initialization and Particle Generation
- Motion Model
- Sensor Model and Particle Weight Normalization
- Re-sampling
- Posterior Estimation and Localization

**Algorithm 3** Particle Filter. Adapted from: [5]

1: **Initialization**: $\overline{\boldsymbol{B}}_k = \boldsymbol{B}_k = 0, M, \mathbb{TM}$
2: **Input**: $(\boldsymbol{B}_{k-1}, \boldsymbol{a}_k, z_k, M)$
3: $\boldsymbol{b}_k^{[n]} =$ **motion model**$(\boldsymbol{a}_k, \boldsymbol{b}_{k-1}^{[n]})$
4: $w_k^{[n]} =$ **sensor model**$(z_k, \boldsymbol{b}_k^{[n]}, \mathbb{T})$
5: $\tilde{w}_k^{[n]} =$ **normalize weights**$(w_k^{[n]})$
6: $\overline{\boldsymbol{B}}_k = \overline{\boldsymbol{B}}_k + < \boldsymbol{b}_k^{[n]}, w_k^{[n]} >$
7: **if** $ESS < \beta M$ **then**
8: resample from **multi-modal sample** $\mathcal{N}(n_\vartheta, \sigma_{\vartheta_n}^2)$
9: add to $\boldsymbol{b}_k^i$ to $\boldsymbol{B}_k$
10: **end if**
11: draw i from **multi-modal sample** $\mathcal{N}(n_\vartheta, \sigma_{\vartheta_n}^2)$
12: $P_k =$ **error covariance**$(\tilde{w}_k^i, \hat{\boldsymbol{x}}_k, \boldsymbol{x}_k^{[n]}])$
13: add $\boldsymbol{b}_k^{[n]}$ to $\boldsymbol{B}_k$
14: **Output**: $\boldsymbol{B}_k, P_k$

**Initialization and Particle Generation**

The initialization of the PF begins with the particle generation and the initial belief state [5], $\boldsymbol{b}_{k0}$, which is a state vector of the vehicle that may include any combination of position, velocity, and acceleration. Also initialized is an *a priori* terrain map, $\mathbb{TM}$ to be utilized within the sensor model, a particle distribution of size $M$ with belief states, $(b_{k1}, ..., b_{kM}) \in B_k$, and the posterior probability density of the current belief state [5]. For further discussion, the superscript $[n]$ will indicate a value of the particle distribution from 1 to $M$. The particle generation is initiated with a random sampling around an initial position estimate to produce a Gaussian distribution and initial equal weights.

The input to the PF algorithm is the prior belief state, $\boldsymbol{B}_{k-1}$, the action of the vehicle, $\boldsymbol{a}_k$, the sensor system input, $z_k$, and an *a priori* map.

**Motion Model**

The input to the motion model function is an action vector determined, in our case, by the motion of the AUV and the belief vector representing the particle distribution. The output

of the function is the updated belief vector $\boldsymbol{b}_k^{[n]}$. The motion model mimics the actions of the vehicle through the particle distribution so that each particle is a hypothesis of the state at time $k$.

**Sensor Model and Particle Weight Normalization**

The input to the sensor model function is the measurement vector $z_k$, and the updated belief state from the previous step $\boldsymbol{b}_k^{[n]}$. Each particle has an associated map measurement that is compared with the actual sensor measurements of the vehicle. As each particle is correlated with the sensor measurement a weight is calculated and assigned to the particle. Each particle's weight is then normalized such that:

$$\sum_{i=0} w_k^{[n]} = 1. \tag{3.16}$$

The effect of this process over time is that, particles that are not consistent with the vehicle's observations are given reduced weights and more particles are generated that are perceived as more consistent with its state. The output to the sensor model are normalized weights that are added to the belief state.

**Re-sampling**

Next, the PF determines if re-sampling must occur using a re-sampling metric $\beta$ that is compared with the effective sampling size (ESS), shown in Equation 3.17. If the conditions for re-sampling are met, then a new random sample of particles is drawn based on the particles weights.

$$ESS = \frac{1}{\sum_{i=1}^{M} (\tilde{w}_t^i)^2}. \tag{3.17}$$

If re-sampling occurs, the output is a newly sampling particle distribution.

**Posterior Estimation and Localization**

The input to the final step is the particle distribution and from it, the highest weighted particle to be used as the estimate of the true belief state. Finally, the covariance is found given equation 3.18 [72]. The final output of the PF is the state estimate and the covariance.

$$P_k = \sum_{i=1}^{M} \tilde{w}_k^i [\boldsymbol{B}_k - \boldsymbol{b}_k^{[n]}][\boldsymbol{B}_k - \boldsymbol{b}_k^{[n]}]^T. \tag{3.18}$$

It is not as easy to compare the complexity of the PF algorithm with the KF and EKF due to the size of the state, $M$. The number of particles that are needed, grow exponentially with the size of the state space and so the complexity can be represented as $O(M^n)$.

The PF has a unique ability to handle non-linear, non-Gaussian distributions which is a critical advantage over other filtering methods. Through particle correlation, the PF can generate particles that are more consistent with the true state of the system and thus converge the true probability distribution including multi-model distributions. The drawback, however, is in the size of the distribution necessary to achieve this. The accuracy of the PF approximation is a function of the particle distribution size. The higher $M$, the higher the likelihood of good particle correlation. As $M$ increases, however, the computational cost necessary to run the PF also increases. This can become an issue for the on-board processing power of many unmanned systems when performing real-time localization [28]. For these reasons, the PF has become a favorite for underwater near-bottom navigation as the topology of the seafloor is often "highly dynamic, non-linear, and non-Gaussian" [36].

Presented in this thesis are a continuum of options for filtering techniques for position estimation. Beginning with simple linear assumptions and going to non-linear, from Gaussian to non-Gaussian, and from the computationally simple to complex, this continuum provides candidate filters that are application dependent. For the application of this thesis, a model-based EKF and a PF are the best tools for position estimation of underwater TAN.

### 3.4.1 Rao-Blackwellized Particle Filter

Due to the potentially large number of particles used in the PF, an approached called the Rao-Blackwellized Particle Filter (RBPF) focuses on sampling the probability distribution

as efficiently as possible. In order to improve sampling within RBPF, some of the variables within the PF must be marginalized according to the Rao-Blackwell Theorem (RBT).

The basis of the RBT is a process that can improve the efficiency of an estimator by taking its conditional expectation with respect to a sufficient statistic [73], [74]. More importantly, however, the improved estimator is better than the original. The RBPF uses the RBT in order to find an estimator of the conditional distribution with the intended goal of reducing the amount of particles necessary to reach the same level of accuracy as a conventional PF. The RBPF provides a significant advantaged over general PF algorithms in that it can greatly reduce the computational demand required to run [75].

The algorithm for RBPF differs from the conventional PF with respect to its Sequential Importance Sampling (SIS), selection, and transition steps [75]:

- Initialization and Particle Generation
- Motion Model
- Sensor Model
- Particle Weight Normalization.
- Sequential Importance Sampling
- Selection
- Markov chain Monte Carlo step

The distribution of $M$ particles is sampled, the weights of each particle are then calculated, normalized and re-sampled based on the weights determined in previous step. The selection step is applied which eliminates the hypotheses with low importance and multiplies those with high importance. Finally, a Markov chain Monte Carlo (MCMC) step is applied [28] which applies a Markov transition kernel, a type of Markov transition matrix, that reduces the total variation of the current distribution [28], [75]. The RBPF will not be discussed further in this thesis, however, is presented here for completeness as it is an extremely robust method for higher dimensional systems.

### 3.4.2 Cramér-Rao Lower Bound

It is important to utilize efficient estimators and equally as important to understand if an estimator is fully efficient. The CRLB expresses a lower bound on the estimation error

covariance of unbiased estimators of a deterministic parameter [76]. This calculated covariance is greater than or equal to the inverse of the information of the distribution [76]. Work done by Cramér and Rao show that if an unbiased estimator that achieves this lower bound, it is considered efficient.

The CRLB has been used to describe the limitations in navigational uncertainty of a PF for state estimation within underwater navigation [33]. Karlsson interprets the CRLB in terms of the INS error. Using methods by Niclas Bergman [77] a standard INS system can be modeled according to the following:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{u}_k + \omega_k, \tag{3.19}$$

$$\boldsymbol{y}_k = h(\boldsymbol{x}_k) + e_k, \tag{3.20}$$

where $\boldsymbol{x}_{k+1}$ is the horizontal position state estimate at $k+1$, $\boldsymbol{x}_k \epsilon \mathbb{R}^2$ is the horizontal position state vector, $\boldsymbol{u}_k$ is the velocity vector, and $\omega_k$ is the process noise due to drift. $\boldsymbol{y}_k$ is the measurement vector, $h(\boldsymbol{x}_k)$ is the measurement function at the current position, and $e_k$ is the measurement noise [33]. Karlsson et al derived the covariance for the CRLB as:

$$\overline{P} R^{-1} Z(\boldsymbol{x}) \overline{P} = Q, \tag{3.21}$$

where $\overline{P}$ is the covariance matrix, $Z(\boldsymbol{x})$ is the expectation of the stationary pose of the vehicle, $R$ is the measurement noise matrix, and $Q$ is the process noise matrix [33]. The significance of finding the CRLB of a PF is how much terrain or depth information is needed for use in positioning and navigation [33].

### 3.4.3 Sequential Importance Sampling

The goal of SIS is to estimate the properties of the probability distribution of the PF, however, it is assumed that there are only samples generated from a different distribution than the one of interest [78]. SIS has been shown to become impractical when applied within a PF due to the variance of the normalized particle weights becoming too large

and consequently, one normalized weight within the particle distribution will dominate while the other particle weights are approximately zero [79]. This phenomenon is called *particle degeneracy*. The danger in this phenomenon is having a distribution that is entirely dominated by one hypothesis which may inaccurate.

### 3.4.4 Sequential Importance Sampling with Re-sampling

Sequential Importance Sample with Re-sampling (SIR) was a later derivative of SIS which added a re-sampling step which facilitates the PF in avoiding particle degeneracy by removing particles that have negligible weights and concentrating on the particles with more significant weights [56].

When re-sampling occurs, the PF re-distributes the same number of particles but with an emphasis on states that had high correlations. The combination of sampling and re-sampling will consequently result in the particle distribution converging on the true belief state of the system. SIR PF can experience convergence on an incorrect belief state which this thesis will discuss in further detail in Ch 5. The re-sampling step is essential for the PF and can prevent a degenerating distribution. For this reason, particle re-sampling has become standard in most MCL algorithms [80].

The adverse effect of re-sampling is that if the estimate of the belief state was correct and the algorithm re-sampled, then the PF must continue with a distribution of a much wider variance. The difficulty then becomes to balance the need to re-sample the distribution with how often it is deemed appropriate for the PF to experience distributions with large variances.

## 3.5 Federated Filtering Methods

This thesis has highlighted situations in which multiple filters are maintained. The fundamental task then becomes, how to combine multiple estimates to create a single best output. The method of combining two or more estimates through a designated process is called *filter federation* [52], [81]–[88]. A summary of federated filtering techniques can be broken down into three subcategories:

- Meta-Filters

- Selective Logic Filters
- IMM Estimation

For further discussion of federated filtering methods, the following nomenclature will be utilized where subscript $k$ refers to time, and superscripts 1 and 2 denote each filter.

- $\hat{\boldsymbol{x}}$ is the state vector.
- $\boldsymbol{G}$ is the innovation.
- $\boldsymbol{a}$ is an action vector.
- $\boldsymbol{l}$ is a Boolean logic vector.
- $\hat{}$ are estimated quantities.
- $\boldsymbol{\mu}$ is the conditional model probability.
- $\Lambda$ is the likelihood function of the filter.

The innovation is the difference between the measurement and the measurement prediction. A Boolean logic vector is a vector containing only either 1 or 0. The conditional model probability is a weighted vector quantity calculated from a state switching matrix which is defined *a priori*. The likelihood function of a filter indicates the likelihood of the filter to produce an observation given certain parameters [52].

The following nomenclature will be used within block diagram discussions for each method:

- State Interaction
- Filter
- Action Detection
- Rule-based logic
- Model Probability Update
- State Estimate Combination

The *State Interaction* uses the current state and applies a weighted matrix entry which mixes the state and covariance estimate. These new mixed state estimates are the input to each *Filter* block which updates the state estimate. The *Action Detection* block provides an action vector from a separate sensor. The purpose of this sensor is solely to identify certain actions or states of the system and is not used as an input to any filter. The *Selective Logic* block is codified Boolean logic that receives the action vector and outputs a Boolean logic

vector which is used to inform the *State Estimate Combination*. The *Model Probability Update* block, updates the conditional model probability based on the filter probability distributions and is also be used to inform the *State Estimate Combination*. The *State Estimate Combination* uses the updated filter state estimates and either the Boolean logic vector or the conditional model probability to output a single best state estimate.

### 3.5.1 Meta-Filters

A Meta-Filter (MF) is a combination of two or more filters placed in either series or parallel. The filters used within a MF are run continuously regardless of the systems state or sensor status. An attribute of this method is that the accuracy of the estimate can improve since filters can be used for different aspects of the state estimation. A limitation, however, is that the quality of this state estimate requires accurate performance of both filters.

#### Parallel

A parallel MF uses two or more filters run independent of each other and whose estimates are combined into a single state estimate. Won et al. proposed a parallel MF that implemented a KF and PF. The KF was used to estimate the three-dimensional position, while the PF was used to estimate orientation. An attribute of this parallel MF is the computational efficiency as each filter was used for a specific type of estimate. Won et al. showed that a federated KF and PF had a significant decrease in estimation error when compared with a single EKF [88]. Figure 3.1 shows a block diagram of a MF with two filters in parallel.

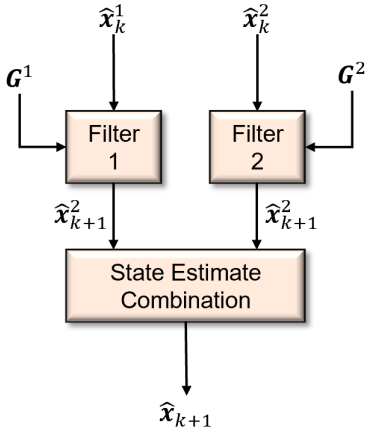



Figure 3.1. Block Diagram of a Meta-Filter with Two Filters in Parallel

A parallel MF begins with the current state estimates and the innovation as inputs to each filter which output an updated state estimate. The state estimate combination block uses the output of each filter and produces a single best estimate.

**Series**

A series MF has an initial filter output that is used by a subsequent filter as an input. The subsequent filter's output, in the case of only two filters, is the final estimate. The limitation of this method is that if a filter outputs a bad estimate, that estimate will be used by any subsequent filters. Shariati et al. proposed a series MF which utilized a PF whose particles were updated using an EKF [89]. A limitation of the series MF developed by Won et al. is the computational effort as the number of particles updated by the EKF was 1000. The attribute, however, was a more accurate state estimate when compared to a single EKF. Figure 3.2 shows a block diagram of a MF with two filters in series.

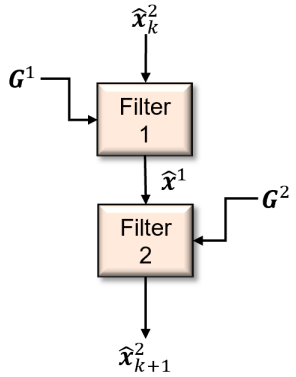



Figure 3.2. Block Diagram of a Meta-Filter with Two Filters in Series

The change from a parallel MF to series is that the state estimate from Filter 1 is the input to Filter 2. Filter 2 then outputs its state estimate which is the final estimate of the MF.

### 3.5.2 Selective Logic Filtering

The filters used in a Selective Logic Filter (SLF) continue to run simultaneously and a "rule-based logic system" [82] is used to inform the state estimation combination step. A SLF may be used, for example, when two filters are used to estimate a dynamic state of a system that is stationary as a part of its the normal mode of operation. Also necessary within a SLF is a system used for action detection which provides an action vector to inform the "rule-based logic" [82]. The limitation of a SLF is that the federation relies heavily on a robust "rule-based logic system" [82]. The attribute, however, is the ability for this system to federate estimates depending on the actions of the system.

Won et al. utilized a rule-based logic system in order to federate a KF and PF. The logic system had the ability to identify a stationary state and inform the KF and PF accordingly [82]. Figure 3.3 shows a block diagram of a SLF with two filters.

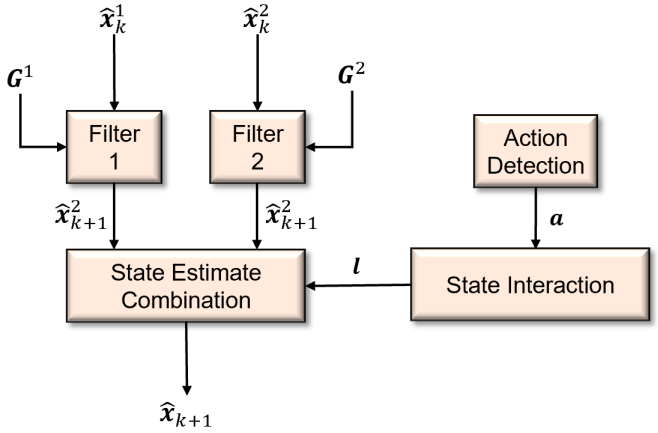



Figure 3.3. Block Diagram of a Selective Logic Filter with Two Filters

The SLF block diagram flows similarly to the parallel MF diagram. The addition is the action detection block that outputs an action vector. The rule-based logic uses this action vector to output a Boolean logic vector. The state estimate combination block uses the filter estimates, and the Boolean logic, to output a single state estimate.

### 3.5.3 Interactive Multiple Model Estimation

The IMM federative process developed by Bar-Shalom and Li [44], implements a Bayesian framework for state estimation through a conditional model probability represented as:

$$\boldsymbol{\mu}_j(k) \triangleq P\left\{N_j|\boldsymbol{Z}_k\right\} = P\left\{N_j|z(k), \boldsymbol{Z}_{k-1}\right\}, \tag{3.22}$$

where, after implementation of Bayes' formula yields [44]:

$$\boldsymbol{\mu}_j(k) = \frac{p[z(k)|\boldsymbol{Z}_{k-1}, N_j]\boldsymbol{\mu}(k-1)}{\sum_{i=1}^{r} p[z(k)|\boldsymbol{Z}_{k-1}, N_i]\boldsymbol{\mu}_i(k-1)}, \tag{3.23}$$

where the filter $N_j$ is one of $r$ possible filters within a system of $r$ modes:

$$N \in \{N_j\}_{j=1}^{r}, \tag{3.24}$$

and where the likelihood function of the model $j$, $\Lambda_j$ is represented by:

$$\Lambda_j(k) = p[z(k)|\boldsymbol{Z}_{k-1}, N_j]. \tag{3.25}$$

The IMM provides a *sub-optimal* approach to the estimation of a Markov jump process that only uses a certain number of filters and is an extremely cost effective algorithm in managing hypothesis [54]. For Markov jump systems, "dynamics are represented by a number of modes", $\mathcal{M}$, "governed by a finite state Markov chain" [54]. Within each mode the continuous state is "described by a set of stochastic differential or difference equations" [54]. An *optimal* filter becomes impractical for state estimation of a Markov jump system since "the number of possible mode sequences exponentially increase" [54].

Consider the following Markov jump system:

$$\dot{\boldsymbol{x}}_k = A_{k-1}(r_k)\boldsymbol{x}_k + B_{k-1}(r_k)\boldsymbol{u}_k + \boldsymbol{\omega}_k(r_k), \tag{3.26}$$

$$\boldsymbol{z}_k = H_k(r_k)\boldsymbol{x}_k + \boldsymbol{\upsilon}_k(r_k), \tag{3.27}$$

where $A_k(r_k)$, $B_{k-1}(r_k)$, and $H_k(r_k)$ are the system matrices that correspond to the mode $r_k$ at time $k$. $\boldsymbol{x}_k \in \mathbb{R}^n$ and $\boldsymbol{z}_k \in \mathbb{R}^m$ are the state and measurement vectors of the system respectively. $n$ and $m$ are positive integers and $r_k$ is a finite state Markov chain where $r_k = \{1, 2, ..., \mathcal{M}\}$. $\boldsymbol{\omega}_k(r_k)$ and $\boldsymbol{\upsilon}_k(r_k)$ are the state and measurement noise, respectively, that correspond to mode $r_k$, and are "assumed to be a zero-mean Gaussian" process with covariance matrices $Q_{k-1}(r_k)$ and $R_k(r_k)$ [54]. For further discussion, the notation $A_k(r_k)$ will be shortened to $A_k^j$ for $r_k = j$ [54].

The conditional probability density function for an "optimal state estimate for a Markov jump system" is given by [54]:

$$p(\boldsymbol{x}_k|\boldsymbol{Z}_k) = \sum_{i=1}^{M^k} p(\boldsymbol{x}_k|\mathcal{M}_k^i, \boldsymbol{Z}_k) p\left\{\mathcal{M}_k^i|\boldsymbol{Z}_k\right\}, \tag{3.28}$$

where $\mathcal{M}_k^i$ denotes the $i$-th possible mode sequence, and where $\boldsymbol{Z}_k = \{z_1, z_2, .., z_k\}$ denotes the cumulative set of measurements.

The IMM estimator has been shown to be "an extremely cost effective federative filter methodology for Markov jump systems" [54]. The attribute of the IMM algorithm is the conditional mode probability, which can be dynamically calculated. This ability gives the IMM the advantage over other federated filtering methods in that it can dynamically adjust its estimation combination. Figure 3.4, adapted from Genovese [52], shows the block diagram of the IMM algorithm with two filters.

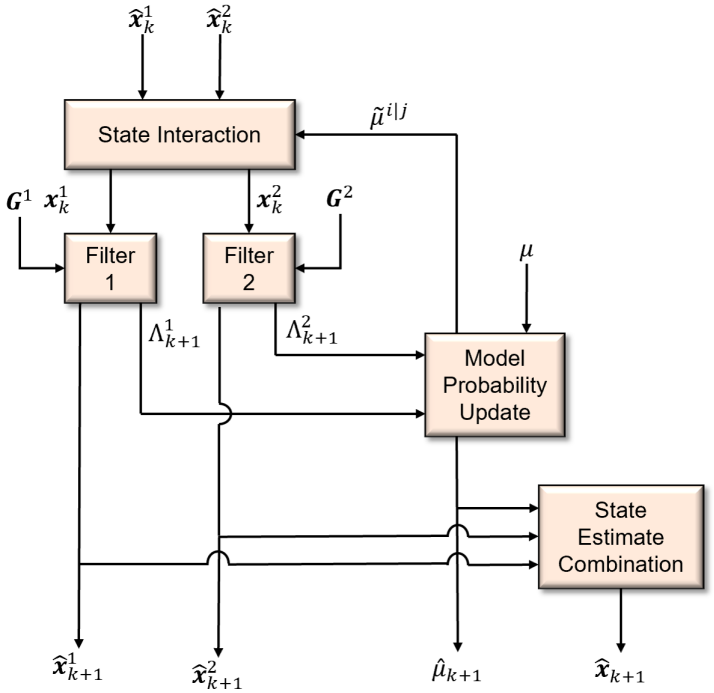



Figure 3.4. Block diagram of an IMM With Two Filters. Adapted from: [52]

The IMM begins with the state estimates and the conditional model probability estimate as inputs to the state interaction. The likelihood functions are input into the model probability update which outputs an updated model probability. The state estimate combination block uses each updated state estimates and the model probability update to produce a single best state estimate [52].

The criterion for state estimate combination can be divided into "*local* and *global* methods" [54]. Local methods only consider "lower order statistics" of the filters when evaluating their similarity such as mean and covariance. Global methods consider all available information of the mixture such as the KLD implemented by Li and Jia [54]. As it applies to this thesis, a global method will be applied as a criteria for component selection.

THIS PAGE INTENTIONALLY LEFT BLANK

# CHAPTER 4:
# Extended Kalman Filter Results

This chapter presents a model-based EKF that is used for position estimation on data collected from Lake Crescent. In the context of the overall thesis, the model-based EKF will utilize a hydrodynamic model as the system model. Since the system dynamics within an EKF are non-linear, a more accurate system model will yield more accurate estimation. The mission profile for the collected data that was used by the model-based EKF was in deep water out of range of the ADCP/DVL. This impacted the normal position estimates because the Kearfott Sea DeViL INS relied on accurate velocity inputs from these sensors. This profile was being tested because it represented conditions similar to those expected during under ice operations in the arctic. In the expected arctic area of operations, the depth of the water is too great for velocity estimates. Testing in Lake Crescent was used to investigate the anticipated behavior of the position estimates without velocity estimates.

For the remainder of this thesis, Table 4.1 shows the standard Society of Naval Architects and Marine Engineers (SNAME) terminology for vehicle position, orientation, velocities, forces, and moments from [90]. These variables will be used to discuss the state of the REMUS within the model-based EKF, the PF, and the IMM developed in this work.

Table 4.1. Notion of SNAME for variables. Source: [90]

| **Linear Variables** | **Positions** | **Linear Velocities** | **Forces** |
|---|---|---|---|
| Surge | $x$ | $u$ | $X$ |
| Sway | $y$ | $v$ | $Y$ |
| Heave | $z$ | $w$ | $Z$ |
| **Angular Variables** | **Euler angles** | **Angular velocities** | **Moments** |
| Roll | $\phi$ | $p$ | $K$ |
| Pitch | $\theta$ | $q$ | $M$ |
| Yaw | $\psi$ | $r$ | $N$ |

## 4.1 Model-based Extended Kalman Filter

The model-based EKF developed here is intended to function under conditions in which the ADCP/DVL are not available. The solution presented in this chapter is to estimate forward velocity ($u$) and sideslip velocity ($v$) using a model-based approach. This combines the propeller turns and the hydrodynamic properties of the REMUS derived by Taylor [41].

The first implementation of the model-based EKF utilized a basic thrust model derived by Bermudez [55] represented as:

$$u = 9.5e^{-14} * r^4 - 9.92e^{-10} * r^3 + 2.04e^{-6} * r^2 + 4.19e^{-5} * r - 0.000539, \tag{4.1}$$

where $r$ is the propeller turns in revolutions per minute (RPM). The limitation associated with this model is that it was derived using empirical data by Bermudez [55]. The hydrodynamics of the REMUS are impacted by the vehicle configuration and this model does not account for the 3DX camera which was used for most of the Lake Crescent missions. This is a model that can be improved and is an area for future work.

The second implementation of the model-based EKF will utilize a linear approximation of

$u$ and estimate $v$ based on rudder deflection. Although this approximation for $u$ is mission specific, it does take into account the 3DX camera. The linear approximation provided by the REMUS, 6.07 cm/rev, was calculated during missions at Lake Crescent. An estimate for sideslip velocity can then be found using simplified equations of motion (EOM) from Taylor [41] and Bermudez [55].

**Estimation of Sideslip Velocity**

First, the horizontal motions of the REMUS are decoupled from the vertical plane which results in:

$$[w, p, q, Z, \phi, \theta] = 0. \tag{4.2}$$

This simplifies the AUV EOM from Taylor and Bermudez [41], [55] to:

$$m\dot{v} = -mur + Y_{\dot{v}}\dot{v} + Y_v v + Y_{\dot{r}}\dot{r} + Y_r r + Y_\delta \delta_r(k), \tag{4.3}$$

$$I_{zz}\dot{r} = N_{\dot{v}}\dot{v} + N_v v + N_{\dot{r}}\dot{r} + N_r r + N_\delta \delta_r(k), \tag{4.4}$$

$$\dot{\psi} = r, \tag{4.5}$$

where the following are hydrodynamic coefficients derived by Taylor [41]:

- $Y_v$ and $Y_r$ are the coefficients of sway force induced by slip and yaw.
- $N_v$ and $N_r$ are the coefficients of yaw moment induced by slip and yaw.
- $Y_{\dot{v}}$ is the added mass in sway.
- $Y_{\dot{r}}$ and $N_{\dot{r}}$ is the added mass induced by yaw motion.
- $Y_\delta \delta_r(k)$ and $N_\delta \delta_r(k)$ are the force and moment produced by the action of the rudder.

These EOM can be re-written in state space form and solved for in terms of $\dot{v}$, $\dot{r}$, and $\dot{\psi}$:

$$\begin{bmatrix} m - Y_{\dot{v}} & -Y_{\dot{r}} & 0 \\ -N_{\dot{v}} & -I_{zz} - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} Y_v & Y_r - mu & 0 \\ N_v & N_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} Y_\delta \\ N_\delta \\ 0 \end{bmatrix} \delta_r(t), \tag{4.6}$$

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = A \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + B\delta_r(t), \tag{4.7}$$

$$A = \begin{bmatrix} m - Y_{\dot{v}} & -Y_{\dot{r}} & 0 \\ -N_{\dot{v}} & -I_{zz} - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} Y_v & Y_r - mu & 0 \\ N_v & N_r & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{4.8}$$

$$B = \begin{bmatrix} m - Y_{\dot{v}} & -Y_{\dot{r}} & 0 \\ -N_{\dot{v}} & -I_{zz} - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} Y_\delta \\ N_\delta \\ 0 \end{bmatrix}, \tag{4.9}$$

where $A$ is the state input matrix, $B$ is the control input matrix, and $\delta_r$ is the control vector. The above set of equations will be used to estimate forward and sideslip velocity based on the propeller turns and hydrodynamic coefficients. The EOM use a North-East and Down (NED) three-axis convention: north ($x$-direction), east ($y$-direction), and down ($z$-direction). The Jacobians $F$ and $H$ for the system and measurement model are represented as:

$$F = \begin{bmatrix} 1 & 0 & ucos(\psi) - vsin(\psi) & sin(\psi) & cos(\psi) & 0 \\ 0 & 1 & -vcos(\psi) - usin(\psi) & cos(\psi) & -sin(\psi) & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.10}$$

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.11}$$

The model-based EKF is designed to provide state estimates without inputs from external positioning sources or the ADCP/DVL and so $H$ will only consider heading and heading rate. This EKF will be federated within the IMM and so it has the ability to adapt $H$ if there are position or velocity measurements available. In the presence of these measurements, $H$ will change in the following manner:

$$Position : H(1,1) = H(2,2) = 1, \tag{4.12}$$

$$Velocity : H(4,4) = H(5,5) = 1. \tag{4.13}$$

For this chapter, the model-based EKF will still use the following sensor measurement vector, however, only heading and heading rate will be non-zero values:

$$\boldsymbol{z}_{k|k} = [x_{LBL}, y_{LBL}, \psi, u, v, r]^T + \boldsymbol{\upsilon_{k|k}}. \tag{4.14}$$

$Q_{k|k}$ and $\mathcal{R}_{k|k}$ are assumed represented as:

$$Q_{k|k} = diag(c_1, ..., c_6),\ c = 0.001, \tag{4.15}$$

$$\mathcal{R}_{k|k} = diag(d_1, ..., d_6),\ d = 0.01. \tag{4.16}$$

The values chosen were kept consistent for all runs and were chosen according to the system and measurement model. First, the state is extrapolated following the form:

$$\boldsymbol{x}_{k+1|k} = f(\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{u}_k) + \boldsymbol{\omega_k}, \tag{4.17}$$

where non-linear equations governing the motion of an AUV represented as:

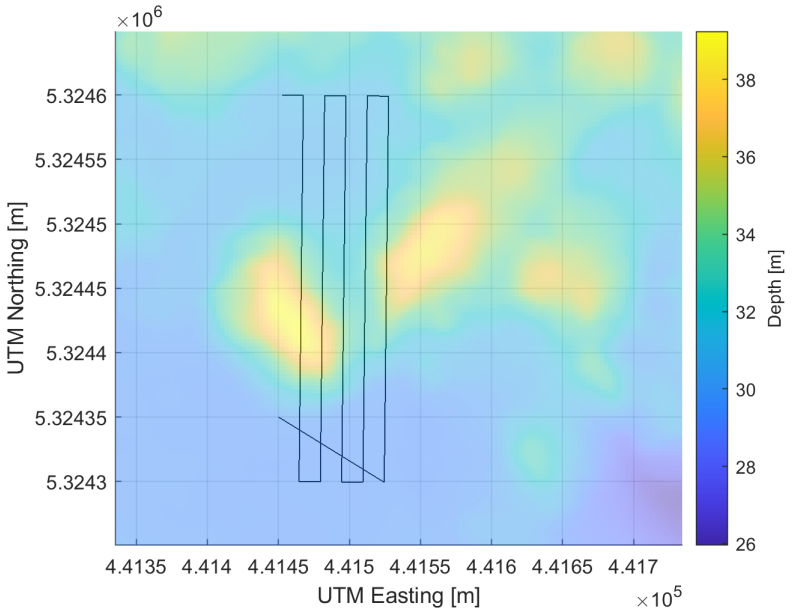
$$\begin{bmatrix} x \\ y \\ \psi \\ u \\ v \\ r \end{bmatrix} = \begin{bmatrix} \hat{x}_{k|k} \\ \hat{y}_{k|k} \\ \hat{\psi}_{k|k} \\ \hat{u}_{k|k} \\ \hat{v}_{k|k} \\ \hat{r}_{k|k} \end{bmatrix} + \begin{bmatrix} u * cos(\psi) - v * sin(\psi) \\ v * cos(\psi) + u * sin(\psi) \\ r \\ 0 \\ \dot{v} \\ 0 \end{bmatrix}, \tag{4.18}$$

where the state vector, $\boldsymbol{x}_{k+1|k}$, is a function of the non-linear state, $\hat{\boldsymbol{x}}_{k|k}$, and the velocity control vector which includes the effects of rudder deflection on $v$. Since, forward velocity is a function of the linear approximation discussed earlier, it is not updated with any change in these equations. Heading rate is also not updated as it can be measured directly from the vehicle. Then the covariance is extrapolated and the Kalman gain is calculated. Finally, the state and covariance estimate are updated.

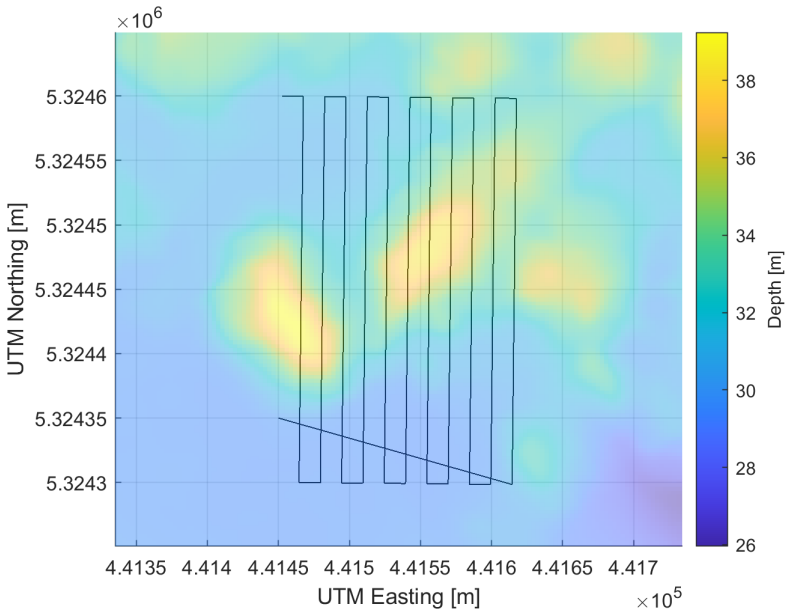
The first iteration of the model-based EKF utilized only the forward velocity derived by Bermudez while the second uses the linear approximation and estimation of sideslip. The results for each implementation are discussed with an emphasis on the improvement of the model-based EKF compared to the manufacturer specific estimates. For further discussion the first iteration of the model-based EKF will be EKF 1 and the second iteration will be EKF 2. It is expected that a more accurate thrust and hydrodynamic model will yield improved estimation performance.

## 4.2 Results of Model-based EKF

Figure 4.1 shows two REMUS missions that were conducted within OA2 of Lake Crescent as shown in Figure 2.8. A complete set of REMUS missions are available in Appendix B.




(a) Northing Parallel Track 1




(b) Northing Parallel Track 2

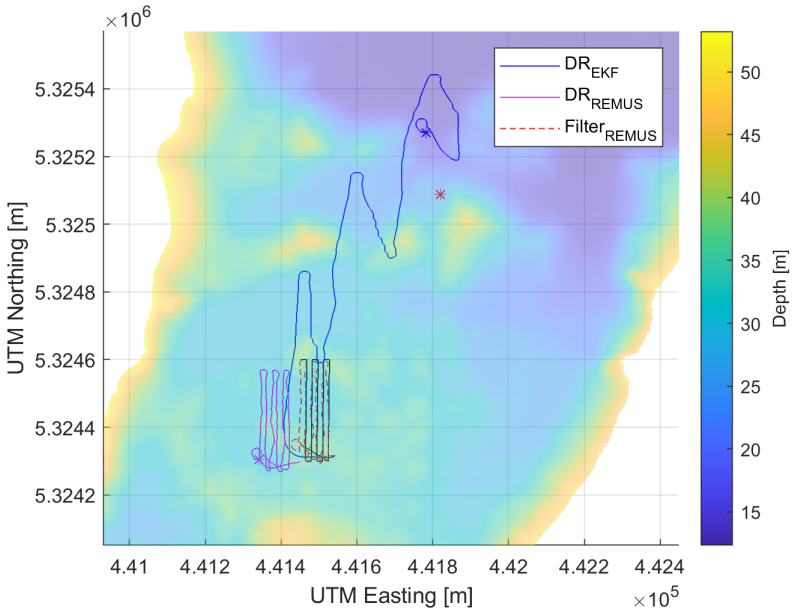
Figure 4.1. Lake Crescent Coverage Missions

During the these two mission sets, the REMUS did not receive velocity estimates. This

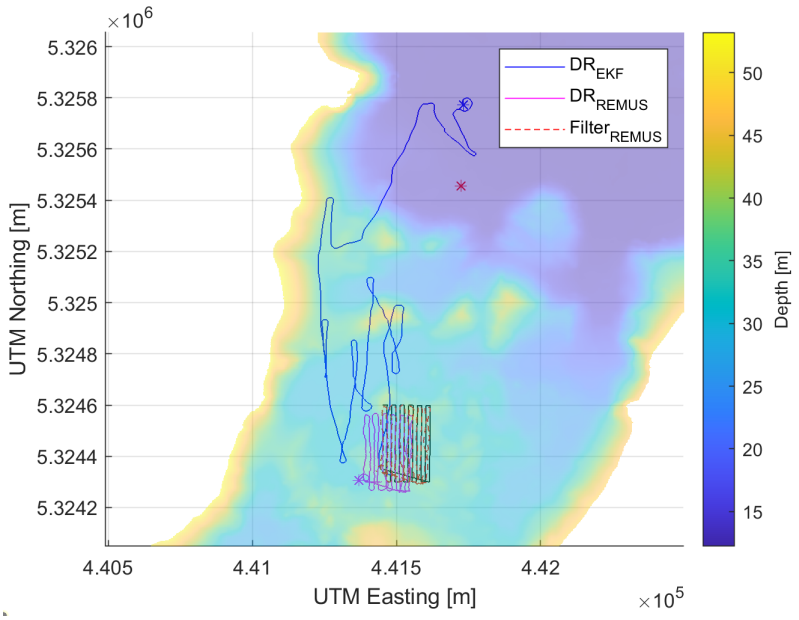
was due to the fact that the REMUS ADCP/DVL was out of range. This resulted a large amount of accrued error and inaccurate estimates from the INS. Under normal conditions, the Kearfott INS can produce accurate estimates of position using a manufacturer filter. Without velocity estimates, the Kearfott INS could only rely on heading and heading rate which resulted in a final position extremely distant from the intended surfacing position. In both missions this error was between 800m $-$1000m.

**EKF 1 Results**

Figure 4.2 show the performance of the manufacturer DR model, $DR_{REMUS}$, the manufacturer filter, $Filter_{REMUS}$, and the first iteration of the model-based EKF, $DR_{EKF}$. The distance between the surfaced point, designated with an asterisk, and the final point of the model-based EKF was reduced to below 200m. Tables 4.2 and 4.3 show the significant decrease the final positional error for both missions.




(a) Northing Parallel Track 1

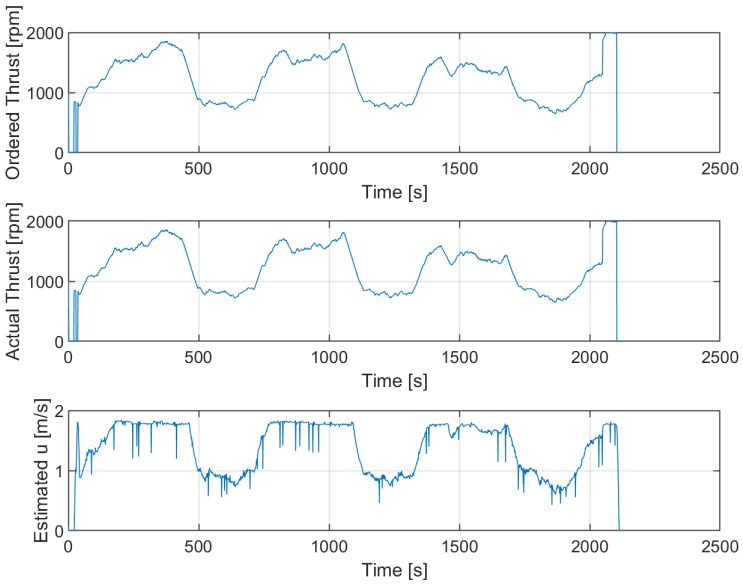



(b) Northing Parallel Track 2

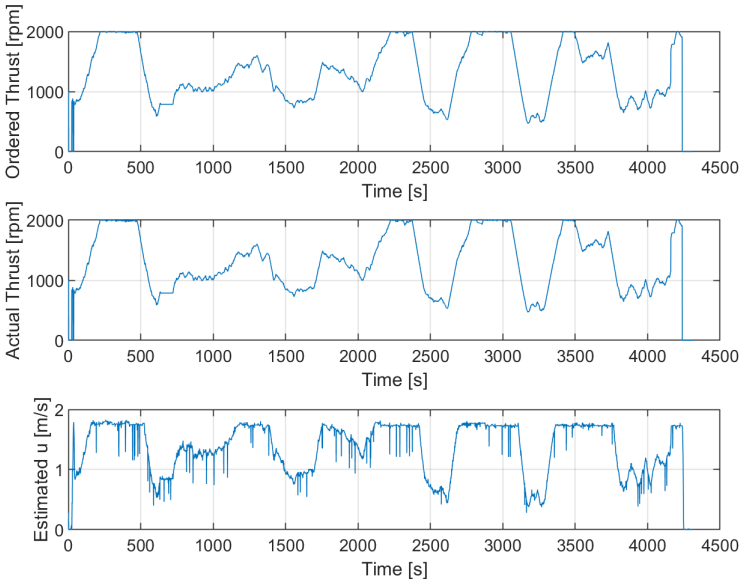
Figure 4.2. Lake Crescent EKF 1 Results

The performance of the ordered thrust command, the actual thrust command, and the

estimated forward velocity are shown in Figure 4.3.




(a) Northing Parallel Track 1




(b) Northing Parallel Track 2

Figure 4.3. Thrust and Model Velocity Estimation

Table 4.2. Lake Crescent Northing Mission 1 EKF 1 Results

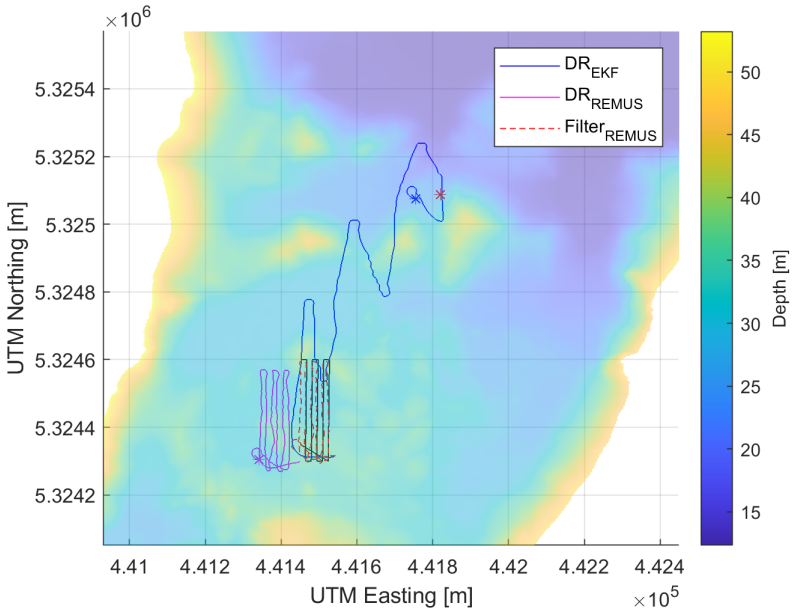
| **Estimation Model** | **UTM North (10) [m]** | **UTM East (10) [m]** | **Dist. from True [m]** | **% of Dist. Traveled** |
|---|---|---|---|---|
| REMUS DR | 5324303.359 | 441341.304 | 918.30 | 39.91 |
| REMUS Filter | 5324329.200 | 441445.089 | 845.53 | 36.75 |
| EKF | 5325269.427 | 441781.738 | 186.06 | 8.09 |

Table 4.3. Lake Crescent Northing Mission 2 EKF 1 Results

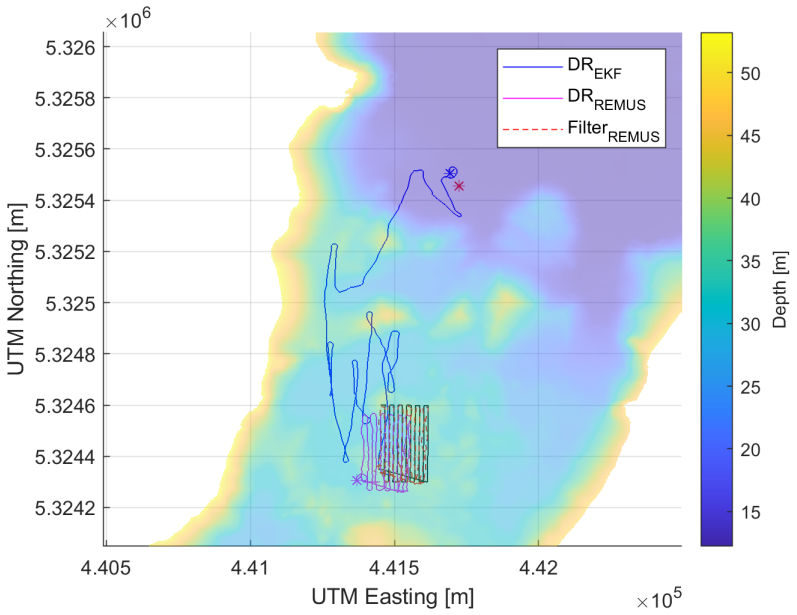
| **Estimation Model** | **UTM North (10) [m]** | **UTM East (10) [m]** | **Dist. from True [m]** | **% of Dist. Traveled** |
|---|---|---|---|---|
| REMUS DR | 5324304.420 | 441369.953 | 1205.44 | 26.65 |
| REMUS Filter | 5324337.434 | 441430.436 | 1157.09 | 25.58 |
| EKF | 5325772.593 | 441731.333 | 316.14 | 6.99 |

**EKF 2 Results**

Figure 4.4 shows the significant improvement of EKF 2 using a model-based estimate for sideslip. The important distinction is that within EKF 1, the system model did not account for sideslip ($v$) was assumed to be zero. It is clear from the results of EKF 2 that a more accurate model yields more accurate results in positional estimation.




(a) Northing Parallel Track 1

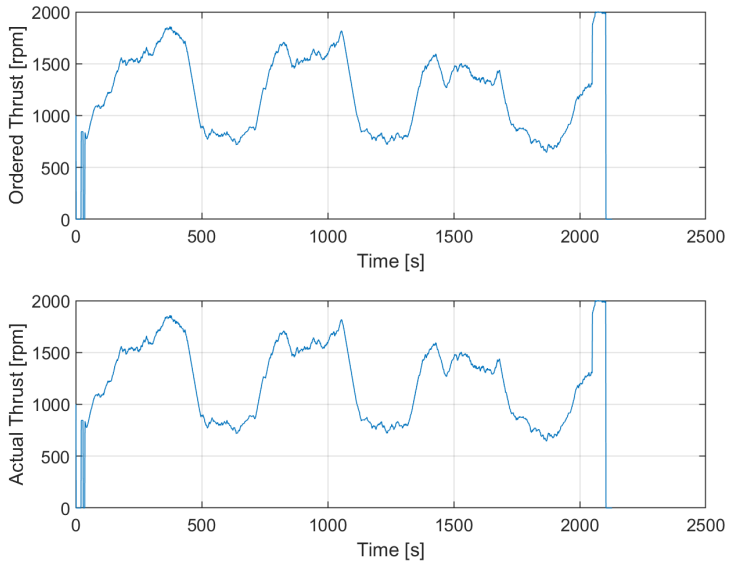



(b) Northing Parallel Track 2

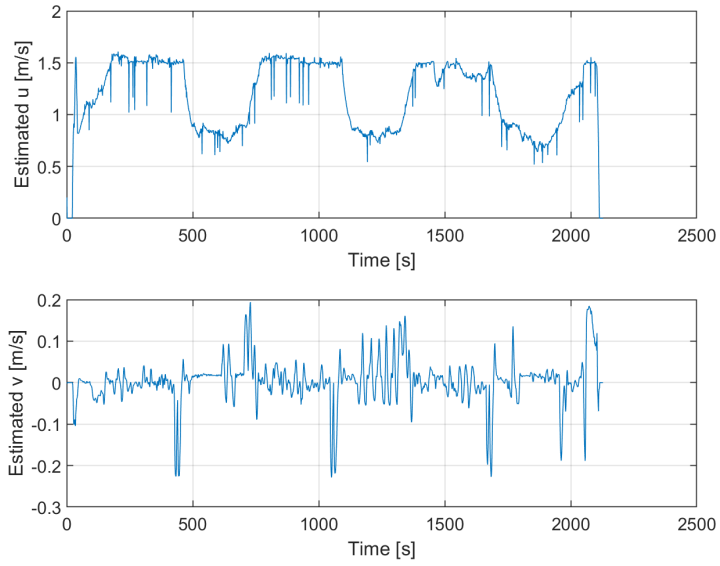
Figure 4.4. Lake Crescent EKF 2 Results

The performance of the ordered thruster command, the actual thruster command, and the

estimated forward and sideslip velocities for each mission are shown in Figure 4.5 and Figure 4.6 respectively.

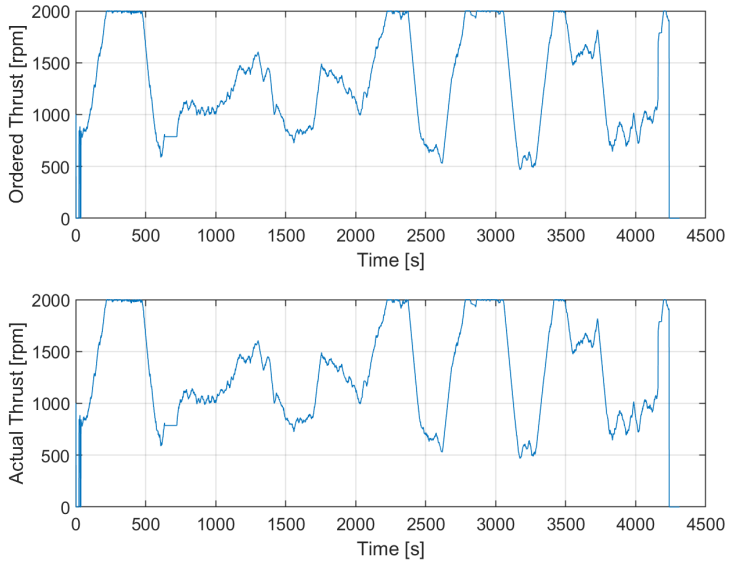



(a) Thrust Commands

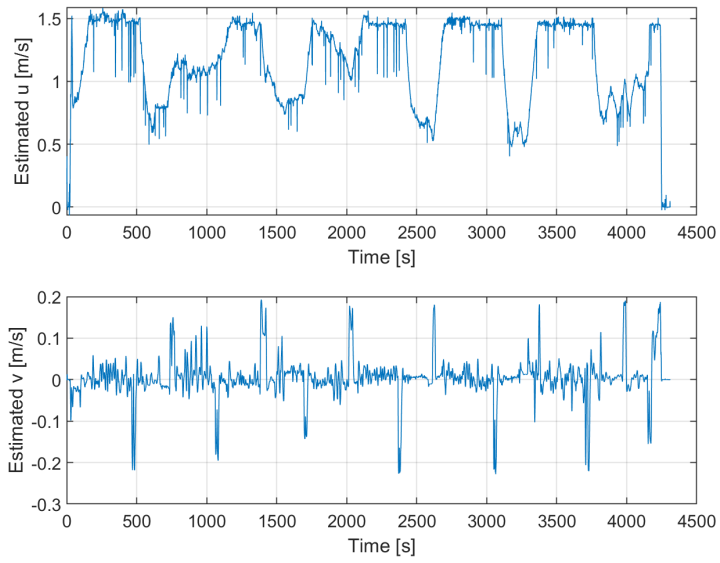



(b) Estimated Forward and Sideslip Velocity

Figure 4.5. Northing Parallel Track 1




(a) Thrust Commands




(b) Estimated Forward and Sideslip Velocity

Figure 4.6. Northing Parallel Track 2

Table 4.4. Lake Crescent Northing Missions EKF 2 Results

| **Northing Parallel Track** | **UTM North (10) [m]** | **UTM East (10) [m]** | **Dist. from True [m]** | **% of Dist. Traveled** |
|---|---|---|---|---|
| 1 | 5325074.847 | 441755.038 | 65.75 | 3.07 |
| 2 | 5325506.122 | 441691.634 | 59.51 | 1.42 |

EKF 2 increased the positional estimation accuracy for each mission by approximately 5% and 5.6% respectively. This result highlights the utility of the model-based EKF developed in this thesis. The strength of this filter is in its ability to adapt to operating either with or without position and velocity inputs. The greatest limitation of the model-based EKF operating under these conditions, however, is that it is not an observable system. This system is not observable because then we cannot determine the state vector solely from the measurements. This means that although the model-based EKF can be a good approach under these conditions, there is no guarantee of its performance. Chapter 6 will federate this filter within an IMM which will create situations in which the model-based EKF will be observable.

THIS PAGE INTENTIONALLY LEFT BLANK

# CHAPTER 5:
# Particle Filter Results

In the last chapter, it was shown how a model-based EKF can be effective when an ADCP/DVL is not available. This chapter presents results based on a PF simulation using a BlueView Sonar input. In the context of the IMM, this is a second filter which is expected to work effectively when there is highly informative bathymetry. It is assumed that that PF is running while an AUV is operating within a regime in which its sensor systems are within range of the bottom topology. It is assumed that because of this, the AUV is receiving accurate velocity estimates. These sensor systems that are effective within a regime like this can include the ADCP/DVL, sonar, and camera systems such as the 3DX.

Within the context of underwater navigation the PF will generate hypotheses (particles) about its state that are scattered randomly based on an initial, Gaussian distribution. At each iteration, these particles are moved based on the motion model and are meant to represent a discretized version of the probability density function for position. After the motion model there is a correlation step where each of the hypotheses are compared to a bathymetric map and assigned weights based upon the strength of their correlation. The PF then makes a determination on which hypothesis represents the true belief state of the AUV.

Within this chapter, an emphasis will be placed on the aspects of this PF that are significant within underwater navigation and are important within the IMM. This includes the particle correlation, when particle re-sampling should occur, how many particles are ideal for estimation, and an introduction of Shannon entropy re-distribution. Figure 5.1, adapted from Salavasidis [36], shows how the Shannon entropy can describe the uncertainty of the particle distribution.

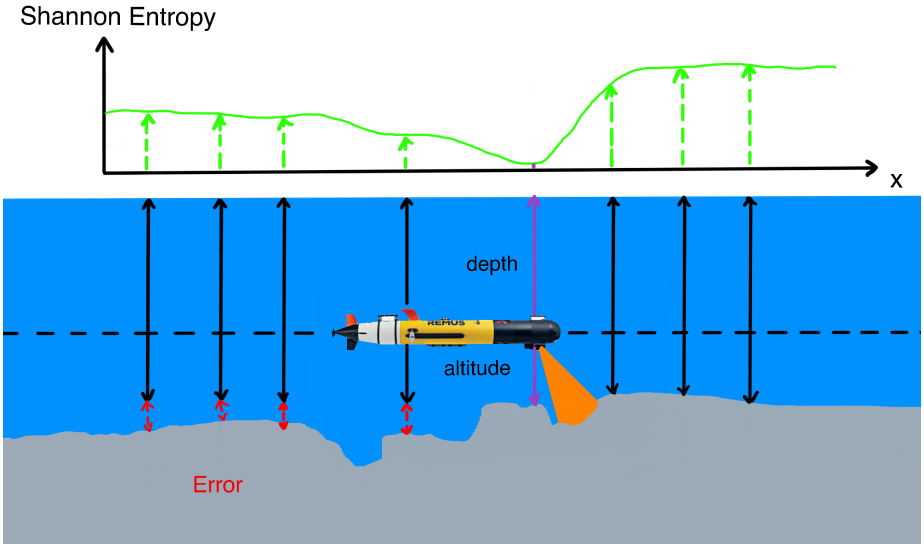



Figure 5.1. On-board water-depth and imaging sensors provide measurements that are correlated against the particle distribution. The Shannon entropy of this distribution shows the uncertainty within the particle distribution with high values indicating that the distribution is uncertain about the position. Since there is a time delay from when the AUV experiences the terrain and when the Shannon entropy is calculated, the scalar value will not be perfectly aligned. It will, however, give good indication that the distribution has a low level of uncertainty. Adapted from [36]

Figure 5.1 shows the non-linearity of the seafloor and how there may be multiple instances of near-identical terrain can be present within the map. Given frequent changes in terrain, a low Shannon entropy can be used to describe the uncertainty of the particle distribution. Conversely, flat or unchanging terrain will yield a high Shannon entropy representing a higher level of uncertainty within the PF distribution. The PF developed in this thesis is an extension of the work done by Darren Kurt [5].

## 5.1 Particle Filter Methodology

The algorithm consists of the following steps:

- Initialization and Particle Generation
- Motion Model
- Sensor Model and Particle Weight Normalization
- Re-sampling
- Re-distribution
- Posterior Estimation and Localization

**Algorithm 4** MCL With Entropy Redistribution

**Initialization**: $\overline{\boldsymbol{B}}_k = \boldsymbol{B}_k = 0, M, \mathbb{TM}$
**Input**: $(\boldsymbol{B}_{k-1}, \boldsymbol{a}_k, \boldsymbol{z}_k, M)$
$\boldsymbol{b}_k^{[n]} =$ **motion model**$(\boldsymbol{a}_k, \boldsymbol{b}_{k-1}^{[n]})$
$w_k^{[n]} =$ **sensor model**$(\boldsymbol{z}_k, \boldsymbol{b}_k^{[n]}, \mathbb{TM})$
$\tilde{w}_k^{[n]} =$ **normalize weights**$(w_k^{[n]})$
$\overline{\boldsymbol{B}}_k = \overline{\boldsymbol{B}}_k + < \boldsymbol{b}_k^{[n]}, \tilde{w}_k^{[n]} >$
**if** $ESS < \beta N \,||\, S_{ESS} > \zeta$ **then**
    **re-sample** from distribution
    add to $\boldsymbol{b}_k^{[n_\vartheta]}$ to $\overline{\boldsymbol{B}}_k$
**end if**
**if** $H_k < H_{LB}$ **then**
    $\overline{\boldsymbol{B}}_k =$ **particle re-distribution**$(\boldsymbol{B}_k, \boldsymbol{a}_k)$
**end if**
draw particle $i$ from distribution
add $\boldsymbol{b}_k^i$ to $\boldsymbol{B}_k$
$P_k =$ **error covariance**$(\tilde{w}_k^{[n]}, \boldsymbol{B}_k, \boldsymbol{b}_k^{[n]})$
**Output**: $\boldsymbol{B}_k, P_k$

### Initialization

The PF initializes the belief state of the particle distribution, $\boldsymbol{b}_{k0}$, which is a vector containing the three-dimensional position of the REMUS. The particle generation utilizes the bootstrap method [5] in which a Gaussian particle distribution is initialized about the belief state, $\sim \mathcal{N}(\boldsymbol{b}_{k0}, \sigma^2)$. The REMUS action is initialized with the heading and velocity components of the vehicle.

**Motion Model**

The inputs to the motion model are $\boldsymbol{a}_k$ and $\boldsymbol{b}_k^{[n]}$. The motion model outputs an updated belief state, $\boldsymbol{b}_k^{[n]}$, using the control action, $\boldsymbol{a}_k$, of the vehicle.

**Sensor Model and Particle Weight Normalization**

The input to the sensor model is measurement vector, $z_k$, and the updated belief state. The sensor model used within this work is a model of the BlueView MB2250 forward looking bathymetry sonar and utilizes the parameters shown in Table 5.1:

Table 5.1. Particle Filter System Model Settings

| **Attribute** | **Specification** |
|---|---|
| Array Size | 10 m |
| Array Resolution | 0.01 m |
| Horizontal FOV | 60° |
| Vertical FOV | 30° |
| Mounting Angle | 45° Downward |
| Max Range | 30 m |

The sensor model correlates the map measurement associated with each particle with the observation. A weight is computed from this correlation process. There are many different correlation methods to include cross-correlation (XCOR), normalized cross-correlation (NXCOR), mean absolute difference (MAD), and mean square difference (MSD). Juriga gives a more in-depth discussion of these methods [56] but the PF developed here will implement MAD as the correlation method shown below:

$$MAD(\boldsymbol{b}_k^{[n]}) = \frac{1}{M}\sum_{i=1}^{M}\left|\boldsymbol{z}_{k,i} - \boldsymbol{b}_{k+1}^{[n]}\right|, \tag{5.1}$$

where the MAD is calculated between the current sensor measurement and the measurement

associated with each particle. There are limitations associated with each correlation method but Juriga shows that "NXCOR, MAD, and MSD perform similarly well" [56]. XCOR was shown to be "ill-suited for correlation" as it is not scale variant which is to say that the "values from the prior map influence the result of the correlation" [56]. The particle weight is applied to the belief state. The weights $w_k^{[n]}$ are then normalized, $\tilde{w}_k^{[n]}$. The final state of each particle is represented as:

**Re-sampling**

An important aspect of this PF is when re-sampling should occur. Traditional PFs utilize the $ESS$ of the particle distribution and a re-sampling parameter, $\beta$, in order to determine if re-sampling should occur. In this method the PF will re-sample the distribution if the $ESS < \beta M$, where $\beta \in (0, 1)$ [5], [75], [80]. This standard method is implemented within this PF as this re-sampling can help bound the error of a distribution that is too small.

A second method introduced in this PF utilizes the covariance of the particle distribution, $S_{ESS}$, with respect to the northing and easting position. If the distribution grows too large, $S_{ESS} > \zeta$ where $\zeta \in (0, 1)$, then re-sampling occurs. Both parameters, $\beta$ and $\zeta$, are chosen *a priori* and are important in tuning this PF. Values that are not tuned well can cause re-sampling either too often or not often enough and can result in degraded performance of the PF.

**Informational Entropy Re-distribution**

This thesis implemented a new metric into the PF algorithm which utilizes Shannon entropy, $H$, as a lower bound to determine when re-distribution should occur. As the particle distribution converges on an estimate, and the covariance decreases, there is always the possibility that the particle distribution converges on an incorrect state. This could occur because of how the particles are correlated or calculated. The limitation associated with this method is that if the PF estimate was correct and re-distribution occurs, the particle population is then re-initialized and the PF must begin estimation from an initial distribution.

Shannon entropy can be applied to the particle distribution as a metric to describe its uncertainty. When $H$ decreases, the uncertainty of the particle distribution also decreases. When the distribution falls below a certain threshold, $H_{LB}$, the particles can be re-distributed.

The Shannon entropy of the particle distribution will be utilized at each time $k$ represented by:

$$H_k = -\sum_{n=1}^{M} p(w_k^{[n]}) ln\left(p(w_k^{[n]})\right). \tag{5.2}$$

The range of the Shannon entropy of the particle distribution is a function of both the terrain as well as the number of particles. As an example, when $M = 5000$, a value of Shannon entropy that represents a probability distribution of high covariance is $H \approx 700$ bits. Conversely, a value of Shannon entropy that represents a probability distribution of low covariance is $H \approx 35$ bits. An understanding of $H$ can help tune the re-distribution step in the same way that the re-sampling parameters can be tuned.

**Localization and Posterior Estimation**

The final step within the algorithm is the determination of the posterior belief state, $\boldsymbol{B}_k$. The PF must be able to make this determination in the best manner possible given either uni-modal or multi-modal distributions. When a PF experiences a uni-modal distribution, the particle weights or mean position of the distribution may be sufficient in determining $\boldsymbol{B}_k$. For multi-modal distributions, the PF may also have to take into account other metrics such as the vehicle dynamics and which distribution is actually more likely given how far the vehicle could have reasonably traveled.

The PF developed in this thesis was run in simulation using terrain data sets for Monterey Bay and Lake Crescent. Verification of the PF was conducted in several parts: its performance as a filter for localization given an *a priori* map (discussed in this chapter), and its performance within the IT-IMM (discussed in Chapter 6).

## 5.2 Monterey Bay Simulation

The simulation for Monterey Bay utilized a bathymetry map, shown in Figure 5.2 and Figure 5.3 [56], and utilizes the Universal Transverse Mercator (UTM) coordinate system with a map resolution of 1m. Figure 5.3 is the two-dimensional overhead view of Figure 5.2 and will be used in further discussion for the missions within the simulation.

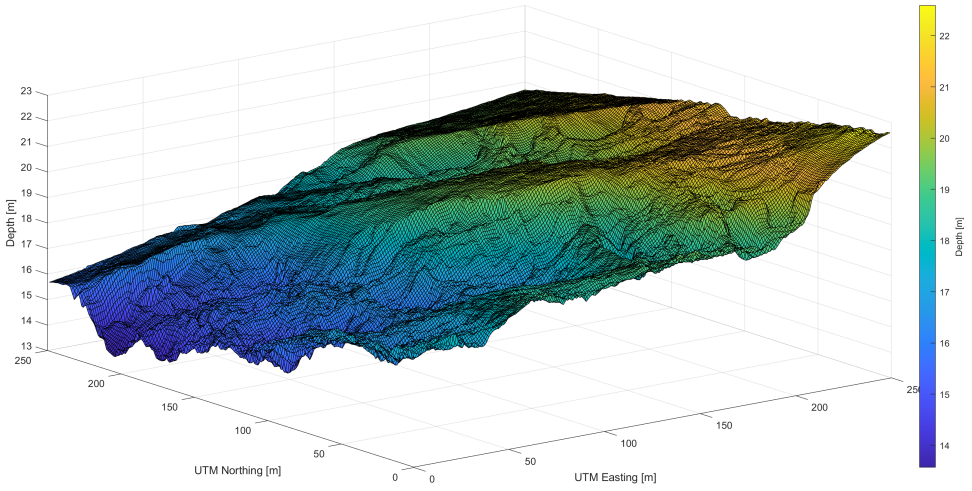



Figure 5.2. Monterey Bay Bathymetry Map

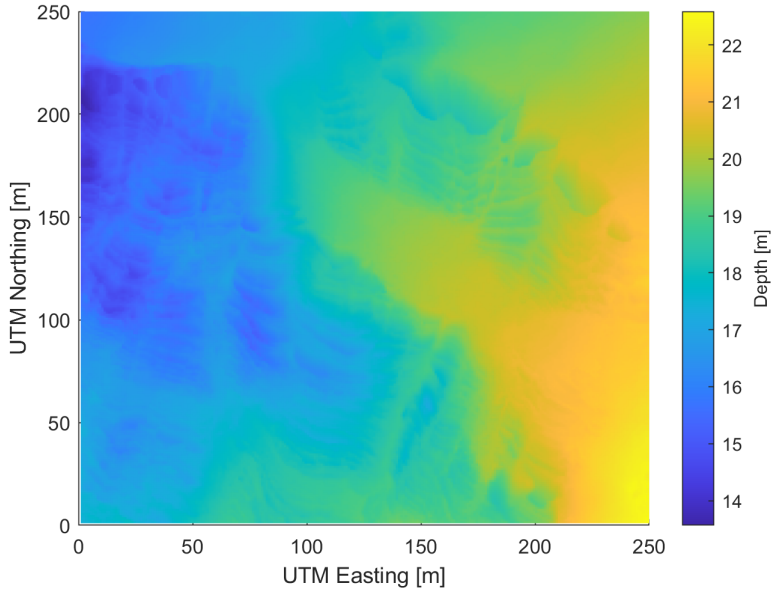


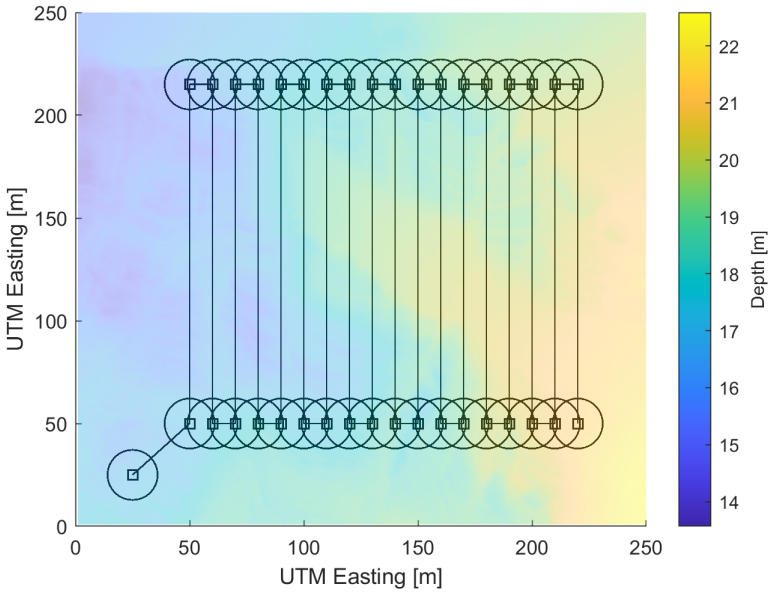

Figure 5.3. Monterey Bay Bathymetry Map

Several full coverage missions were used in the simulation in order to gain a better understanding of the performance of the PF as a function of the terrain and informational entropy. A north/south parallel track mission, an east/west parallel track mission, and a shrinking square mission were chosen for testing. The north/south parallel track is shown in Figure 5.4 with the latter two tracks shown in Appendix A. The PF was run with the following settings:

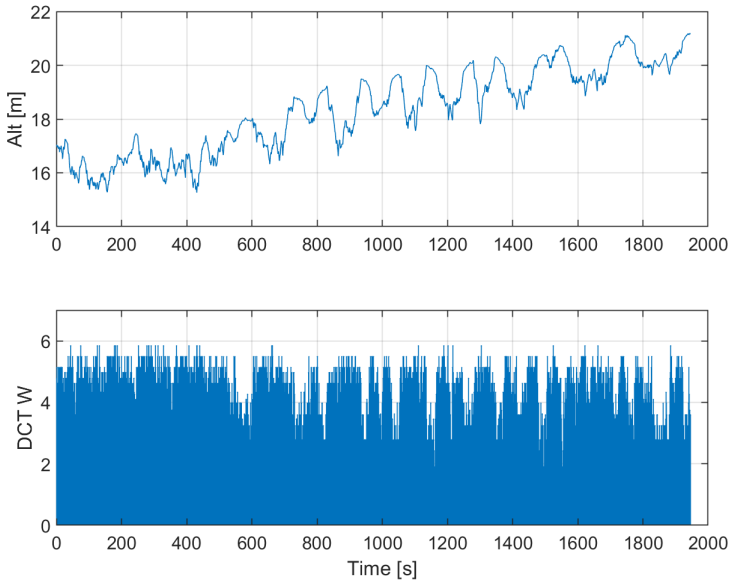
Table 5.2. Particle Filter Settings

| **Nomenclature** | **Setting** | **Value** |
|---|---|---|
| M | Number of Particles | 2000 |
| $\beta$ | Re-sampling Parameter | 0.5 |
| $\zeta$ | Re-sampling Parameter | 0.003 |
| $H_{LB}$ | Shannon Entropy Re-distribution Parameter | 15 |
| | Sensor Model | BlueView MB2250 |

The north/south parallel track mission was chosen for continued PF testing as it provided a slightly higher mean DCT BE shown in Figure 5.4. It is important to note that each mission experienced significant altitude and DCT BE changes, in both frequency and magnitude. The change in altitude and DCT-BE experienced by the simulated REMUS for each coverage mission is shown in Appendix A.




(a) North/South Parallel Track




(b) North/South Parallel Alt and BE

Figure 5.4. North/South Parallel Track Change in Altitude

It is important to note that the change in in altitude is only in one dimension while the

DCT-BE takes into account the change in terrain in two dimensions. Figure 5.4 shows the significant changes in terrain which is extremely important for the PF's correlation process. If the REMUS experiences terrain that does not have any significant changes then there will be a higher level on uncertainty within the particle distribution. Both the change in terrain and frequency of change improve the PF performance.

## 5.3 Results of PF Performance

Simulations were then conducted according to the coverage mission in Figure 5.4 with an increasing amount of particles in order to determine the ideal number. The number of particles is a function of each step of the PF algorithm. It is important to determine an ideal number as the PF is the most computationally dense within the IMM. Every effort must be made to appropriately save on computational cost within the PF, to include the size of the distribution, without sacrificing performance. An ideal number of particles will be such that any increase does not yield any significant improvement in state estimates.

Each was run was conducted 30 times in order to build a normal distribution of PF estimation error data for proper analysis [91]. Following the Central Limit Theorem (CLT), a sample size over 30 begins to develop a population of data whose mean and standard deviation are more accurate to the true estimation error. The CLT shows that the distribution of the data will be normal even if the population is not necessarily normal [91].

The results in Figure 5.5 show that as $M$ increases the accumulated error decreases to approximately 800 m and the mean error converges to about 0.4 m. No significant accuracy is gained in the PF performance, given this simulation, when $M > 5000$. As the complexity of belief state increases in future testing, this should increase $M$ required to achieve accurate results. For the PF developed here further testing will be conducted at $M = 5000$.

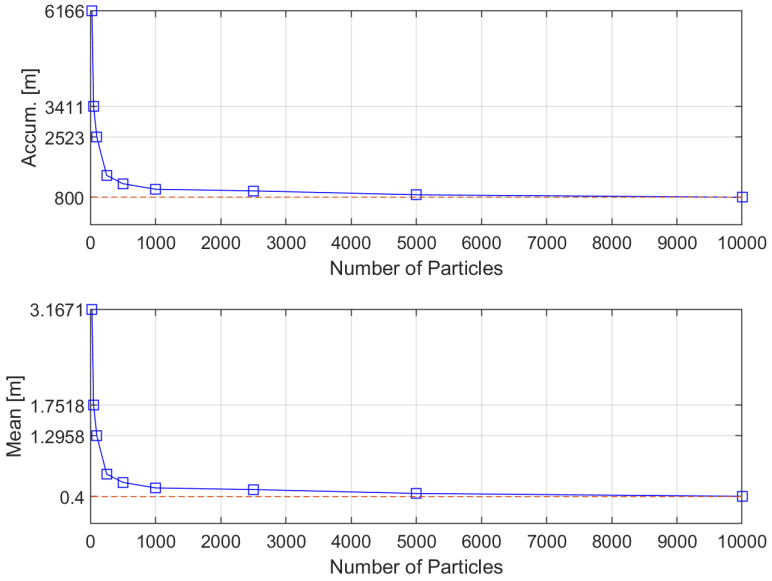



Figure 5.5. PF Estimation Error

Accumulated error also drops approximately 87% from $M = 25$ to $M = 5000$ and only 8% from $M = 5000$ to $M = 10000$. Again this is significant because there is tremendous change in the accumulated error up until $M = 5000$ and computational efficiency can be maintained by fixing M at 5000.

There is also significant decrease in the variability of the estimation error as shown in figure 5.6. At low values of $M$, the PF estimates have a relatively high variance which shows that the estimates that the PF is choosing is more disperse from the mean value its estimates over the entire simulation. This is significant because at higher values of $M$ the PF estimates are much closer to the mean and thus are more consistently accurate over time.

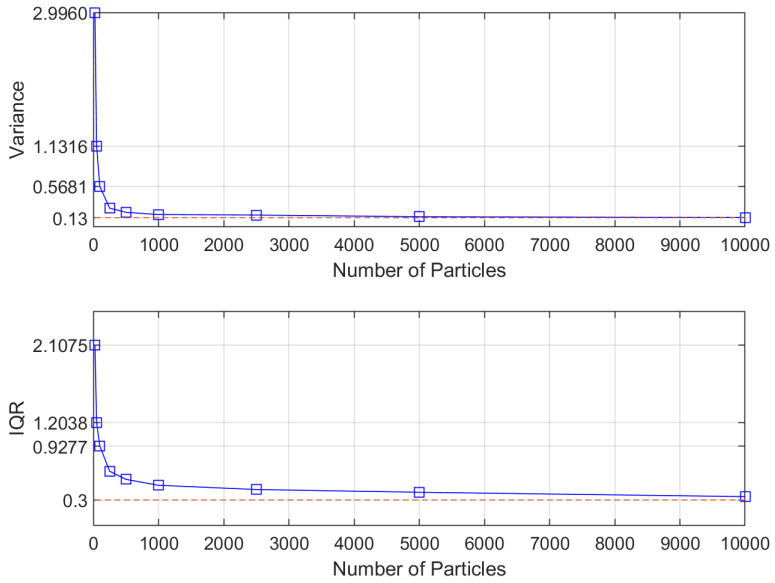
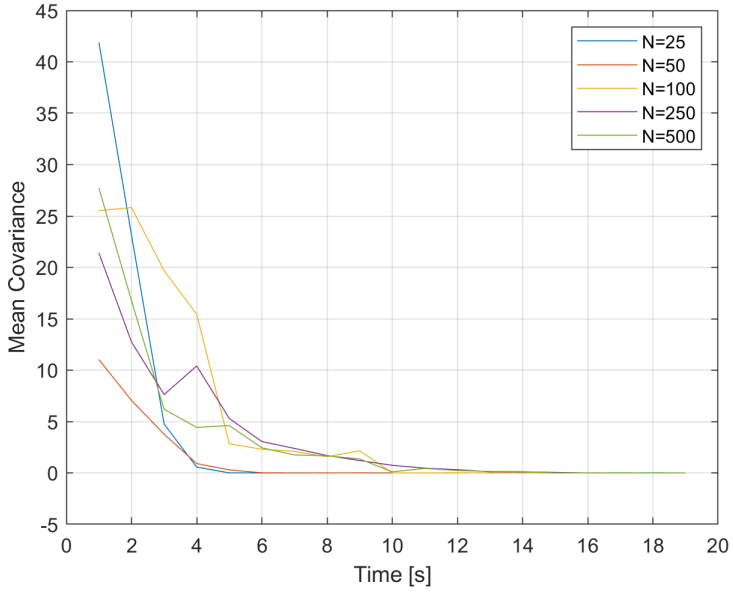
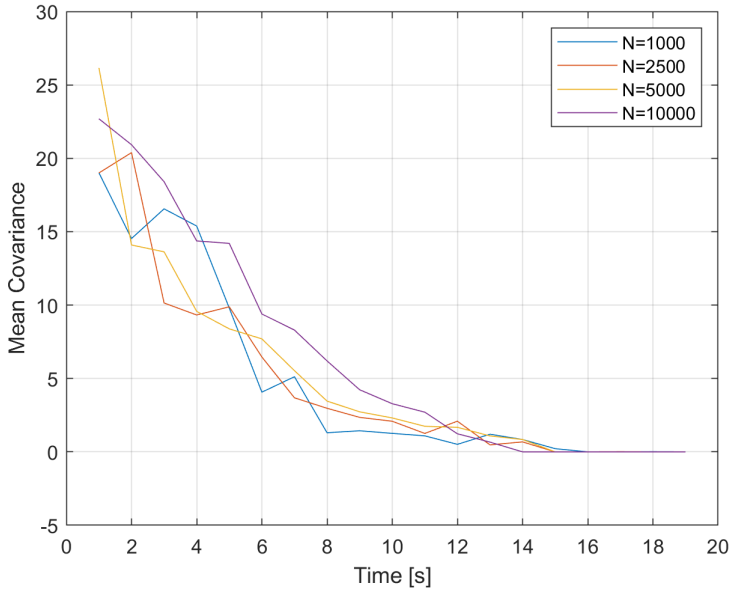



Figure 5.6. PF Estimation Variability

The interquartile range (IQR), when shown with the variance, is a strong indicator of the variability of the data [92]. The IQR shows that as $M$ increases there is a greater amount of estimates that are more concentrated about the mean. Figure 5.7 shows a single iteration the covariance of the particle distribution at each $M$.




(a) Small Distributions




(b) Large Distributions

Figure 5.7. Particle Filter Covariance Prior to Re-distribution

Within this iteration, the particles start at an initial distribution and thus have a large

covariance. As the PF converges on an estimate, the covariance decreases up until the point that the PF determines that re-distribution is necessary. This highlights one of the limitations associated with the Shannon entropy re-distribution. If the PF has an accurate estimate with a low covariance, it will need to begin this process over again and incur a large covariance.

Figure 5.8 shows the increase in the minimum Shannon entropy of the particle distribution as a function of $M$ and were derived through simulation. These values are used as the threshold for re-distribution. As $M$ increases, so does this approximate minimum value.

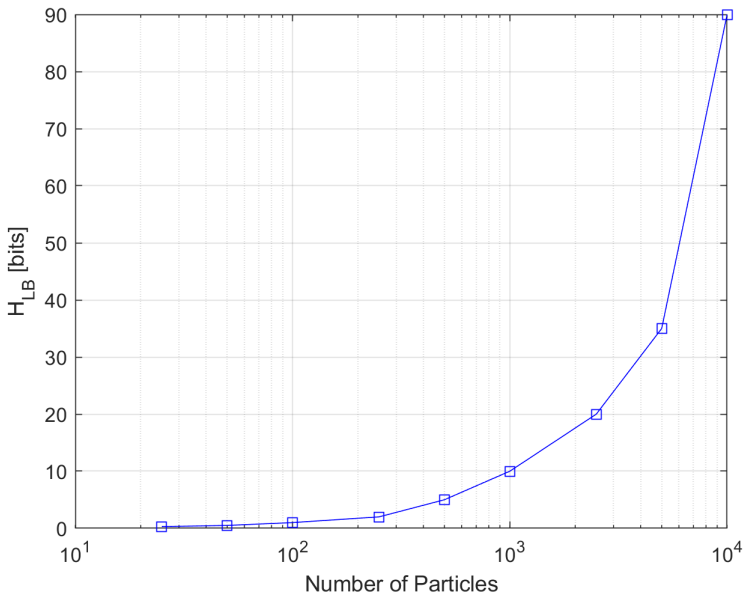



Figure 5.8. Minimum Shannon Entropy of Particle Distribution

The PF developed in this thesis verified, through simulation, that the $M$ proposed by Kurt [5] of $M = 1000$ provided an accurate state estimation for the AUV. The PF will be run at $M = 5000$ for the IT-IMM testing in Chapter 6 as it is more ideal. The PF focused on re-sampling, re-distribution, and an ideal particle size in order to provide the best estimates while saving on computational cost.

This filtering technique requires the AUV to operate within a regime in which its sensors are in range of the ocean floor. This filter was developed because it provides accurate position

estimation for underwater navigation which has a covariance that is bounded. Unlike the model-based EKF, an attribute of this filter is that the error can be bounded throughout a mission so long as there are accurate sensor measurements. As discussed earlier, it is common for AUVs to operate within or through several regimes in which not all sensor systems on-board are within range to provide the required inputs for these types of filters. It is necessary then to federate both the model-based EKF and the PF in order to provide a single best output throughout multiple regimes. In the next chapter, these two filters will be brought together using an IT-IMM.

THIS PAGE INTENTIONALLY LEFT BLANK

# CHAPTER 6:
# IT-IMM Results

In the previous chapters it was shown how a model-based EKF and a PF can be used for position estimation. The IT-IMM algorithm developed in this chapter will federate the estimates of each of these filters. Presented in this chapter is an information theoretic methodology that can inform a Markov Jump matrix (model probability update), to facilitate this federation. The information theoretic approach uses a combination of two measures to determine when to switch between the two filters. The first uses Shannon entropy of the AUV posteriori probability density function and the second uses a measure of terrain information (DCT-BE) for determining the ability of the filter to provide better position estimates. The chapter concludes with results from simulated runs using the USGS surveyed bathymetry from Lake Crescent, WA.

## 6.1 IT-IMM Methodology

Figure 3.4 from Chapter 3 shows a general IMM federation approach. Li and Jia extend this approach using the KLD to inform the federation process. The novel approach presented in this chapter takes into consideration the virtue of the AUV position estimate and the terrain variability for determining when to use the PF solution as a synthetic measurement for a second EKF. The block diagram is shown in Figure 6.1.

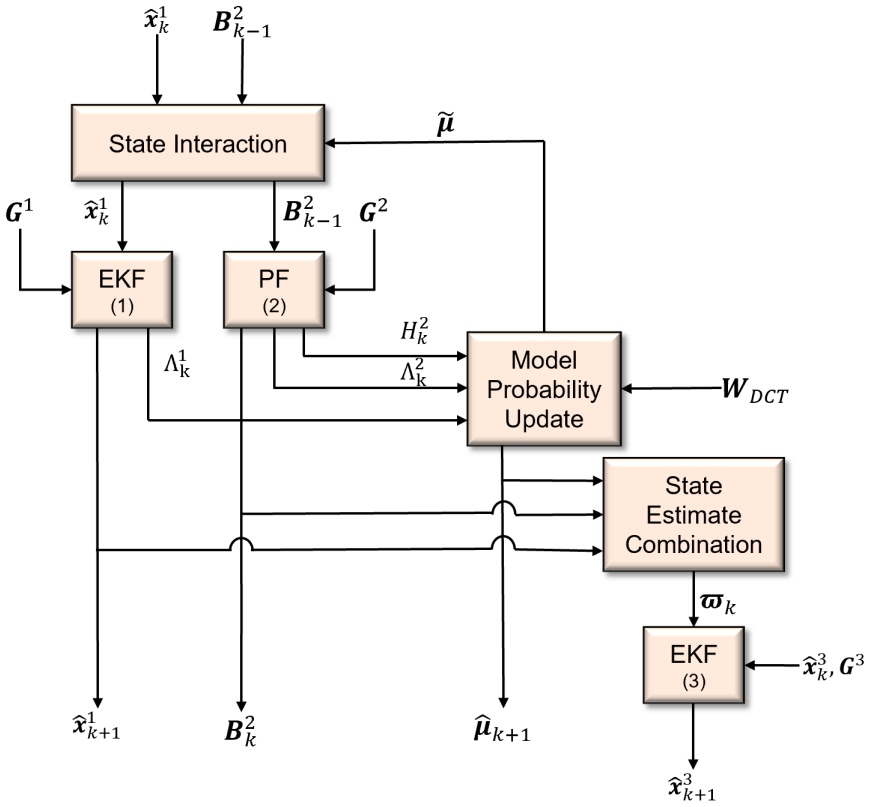



Figure 6.1. IT-IMM Block Diagram

For further discussion of Figure 6.1, the following new nomenclature will be utilized:

- $W_{DCT}$ is the DCT-BE Map.
- $\varpi$ is the synthetic federated position vector.

Two model-based EKFs and the PF are used and are labeled with a filter number for clarification. EKF (1) is run continuously during a mission is not impacted by any other filter. As discussed in Chapter 4 this can provide good estimates but will have significant unbounded error given time. The PF will be run when the AUV is within sensor range of the bottom topology and therefore will be used intermittently within certain regimes. Finally, EKF (3) will also be run throughout the mission but will obtain a federated output from the state estimation combination block which should bound its error. This output is in the form

of a synthetic position measurement to be used within the measurement model of the EKF. The term *synthetic* refers to the fact that this is not a true sensor measurement but one that will be used as such within the EKF.

The diagram begins with the *State Interaction* block which inputs the state estimates as well as a weighted matrix from the *Model Probability Update*. A simplification within this implementation is that this weighted probability matrix only contains either 1s or 0s to inform the interaction. Within the simulation run using the IT-IMM developed here, only 0s were used and so no mixing of the state or covariance estimates occurred. Next, the state estimates as well as the innovations input into each respective filter. For the model-based EKF this is the current state estimate and for the PF this is the previous belief state. The likelihood functions from each filter as well as the Shannon entropy from the PF are input into the *Model Probability Update*. The DCT-BE is also an input into this step. The Markov Jump matrix that is used for this Markov Jump process is output from the *Model Probability Update* which is used to inform the state estimation combination. The Markov Jump matrix is a weighted $2 \times 2$ probability matrix where the rows add to 1 with values (0,1) and the number of columns are the number of models in consideration. Within this thesis, however, only 1s and 0s will be utilized within the matrix. When the IT-IMM algorithm determines, through entropy, that there is a high level of confidence in the PF estimation, the Markov jump matrix shown below is used:

$$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \tag{6.1}$$

For all other iterations in which there is a low level of confidence then the matrix is represented as:

$$\begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}. \tag{6.2}$$

When the matrix in 6.2 is chosen, the *State Estimate Combination* will not federate estimates, and EKF (3) will only utilize its previous estimate. The attribute of this IT-IMM is its ability to combine Gaussian and non-Gaussian estimates using terrain information, to make future

predictions about how well the PF will work. A limitation of this current implementation, however, is that this Markov Jump matrix utilizes only the conditional probabilities of 1s or 0s. This creates a federative process that is either all or nothing. An investigation into a Markov Jump matrix that is more robust is an area for future work.

## 6.2 Entropy Behavior

The model probability update within this IT-IMM will be informed through Shannon entropy, $H$, and DCT-BE, $W$. It is important to understand the behavior of $H$ and its relationship with DCT-BE. As described in Chapter 2, the Shannon entropy and DCT-BE are scalar values which can be used to describe the uncertainty of the particle distribution and the value of terrain respectively. An assumption within the testing of this simplified IT-IMM, is that an OSE is fully known, and can be used to inform an expectation of $H$. An area for future work is further investigation when there is a stochastic spatial estimate.

Two samples of terrain, shown in Figure 6.2, were chosen from the Lake Crescent bathymetry map to show the different behavior of $H$ as a function of DCT-BE.

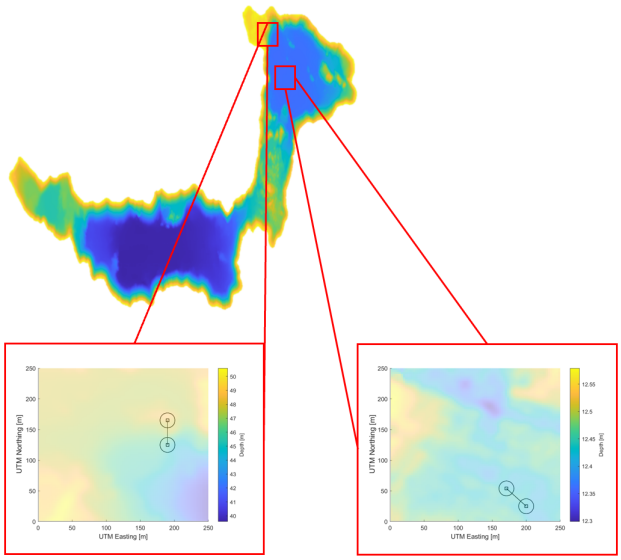
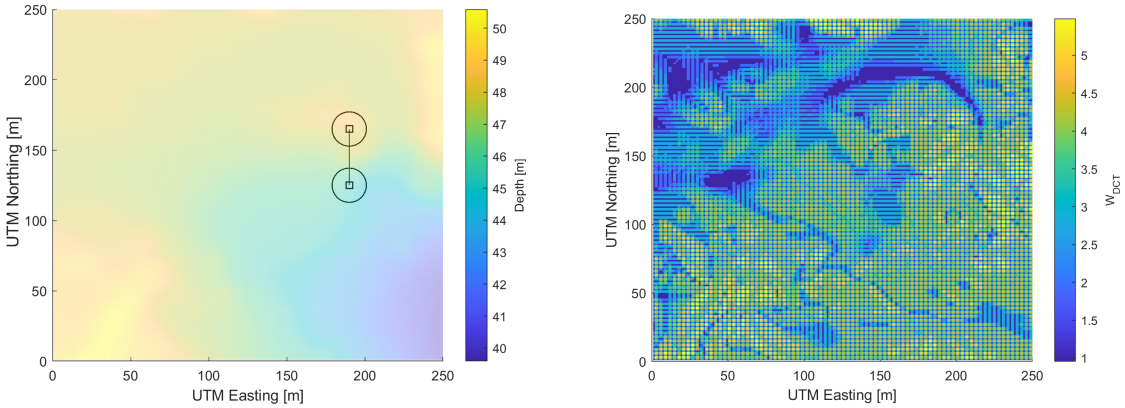


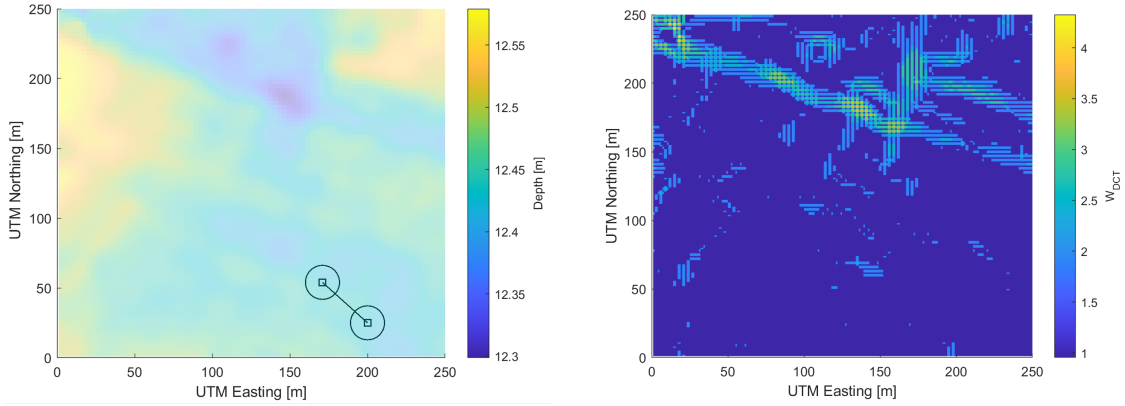

Figure 6.2. Terrain Selection of Lake Crescent

Figure 6.3 and 6.4 show each selection of terrain chosen and its corresponding DCT-BE map. The track in each bathymetry view was used to investigate the behavior $H$ over time given a high and low level of information.




(a) Bathymetry View (b) DCT W View

Figure 6.3. Informative Terrain



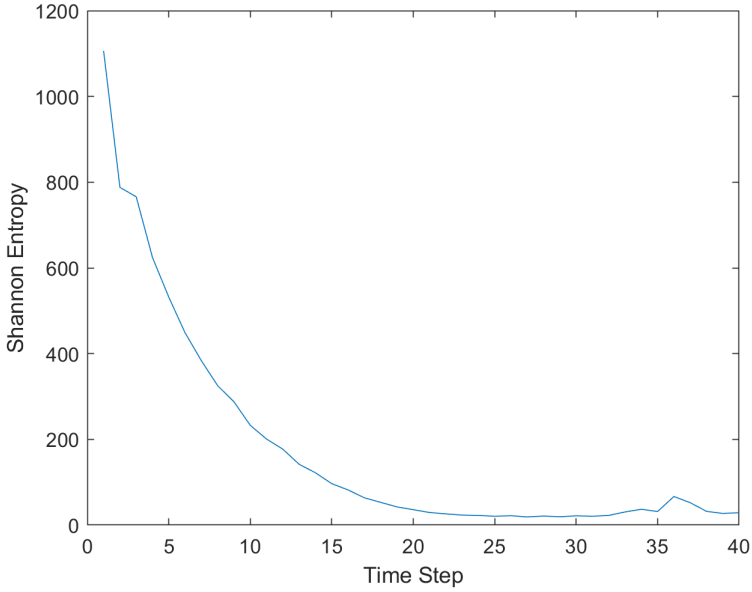

(a) Bathymetry View (b) DCT W View

Figure 6.4. Non-Informative Terrain

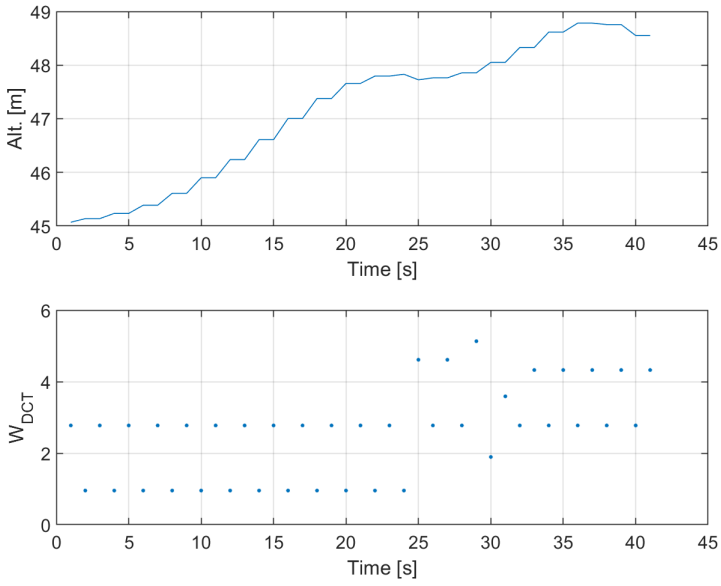
The informative bathymetry in Figure 6.3 may only have a difference in its actual terrain of 10 m, but the DCT-BE map gives a more illuminating view of the high amount of information

within the terrain itself. This is in contrast to Figure 6.4 which has a difference in elevation of only about 0.25 m and shows a much lower and much more uniform distribution of terrain information. When the particle distribution is over informative terrain there will be a significant difference in particle weights as there will be better map correlation. In non-informative terrain, particle correlation is more difficult will generate a higher degree of uncertainty within the distribution.

As the PF experiences terrain with high quality information there is an exponential decrease in $H$, shown in Figure 6.5. This is in contrast to terrain with poor information which presents a chaotic behavior. Figure 6.5 shows the relationship between the Shannon entropy of the particle distribution and the DCT-BE of the terrain.




(a) Change in Shannon Entropy

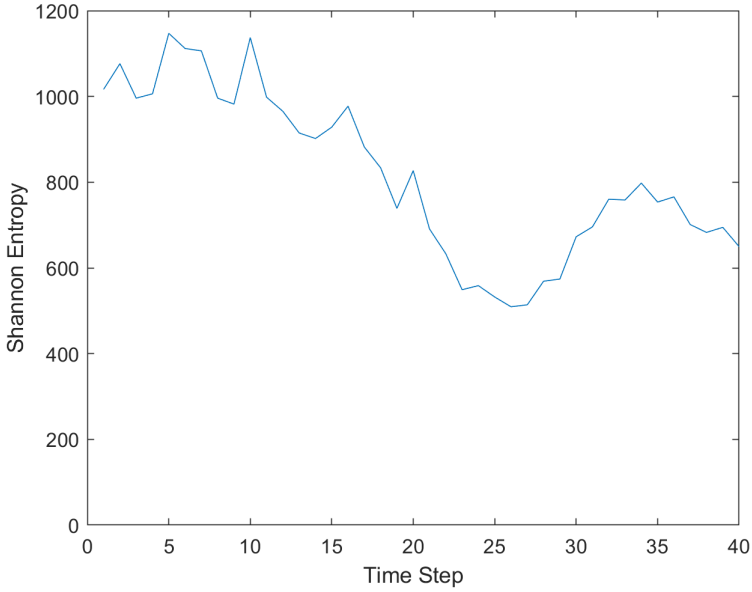



(b) Change in altitude (top) and DCT-BE (bottom)

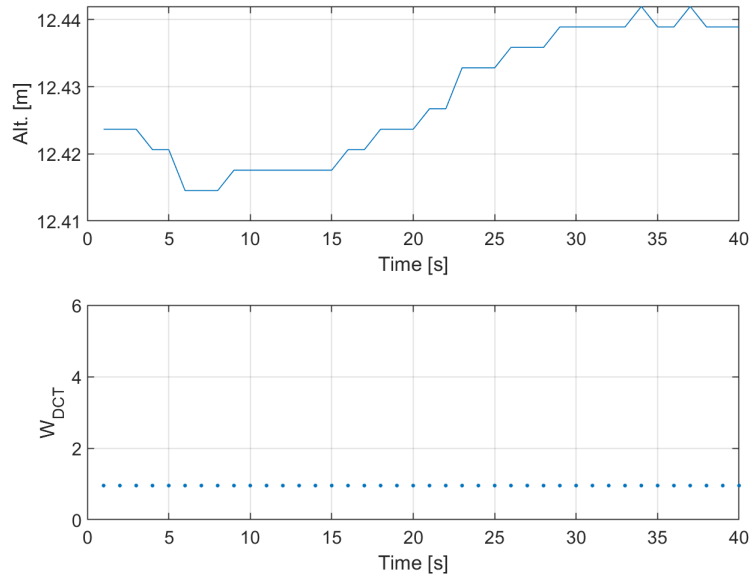
Figure 6.5. Informative Terrain

As shown in Figure 6.5, as the AUV experiences informative terrain, and as this information

increases, $H$ maintains a very low value indicating that the more informative the terrain, the less uncertainty there is within the particle distribution. The behavior of the Shannon entropy within non-informative terrain is chaotic and does not follow the exponential decrease shown in Figure 6.6.




(a) Change in Shannon Entropy




(b) Change in altitude (top) and DCT-BE (bottom)

Figure 6.6. Non-Informative Terrain

The interesting relationship within the non-informative terrain is that for very small and

constant values of DCT-BE, $H$ will experience random behavior. From Figure 6.5 it can be seen that, within informative terrain, we can build an expectation of the behavior of $H$ given changes in DCT-BE. With this behavior, we can have a strong level of confidence within the correlation process of the PF and thus inform the Markov Jump matrix in order to facilitate federation.

Several different expectations were investigated that informed the Markov jump matrix. The first, IT-IMM 1, implemented a future expectation of $H$ as a threshold value. The second, IT-IMM 2, used an expectation of the of the change in $H$ up until time $k$. IT-IMM 3, is a combination of 1 and 2. All three implementations also considered the DCT-BE as the change in $H$ was shown to be a function of the information within the terrain. IT-IMM 3 is hypothesized to have a higher performance, in terms of reducing navigational error, then IT-IMM 1 or 2 as it is the most robust in informing the algorithm of the PFs confidence.

## 6.3 Results

The IT-IMM developed in this thesis was implemented over a section of informative terrain from Lake Crescent, shown in Figure 6.3, and was chosen because of its realistic application to the overarching underwater navigational problem. The selected track, shown in Figure 6.7, was chosen as the simulated REMUS will experience different regimes in which its sensors will not always be in range. Next, upon conditions that inform the Markov jump matrix, the IT-IMM will begin to combine the estimates of both filters. This will continue throughout the coverage mission and highlight the utility of the IT-IMM.

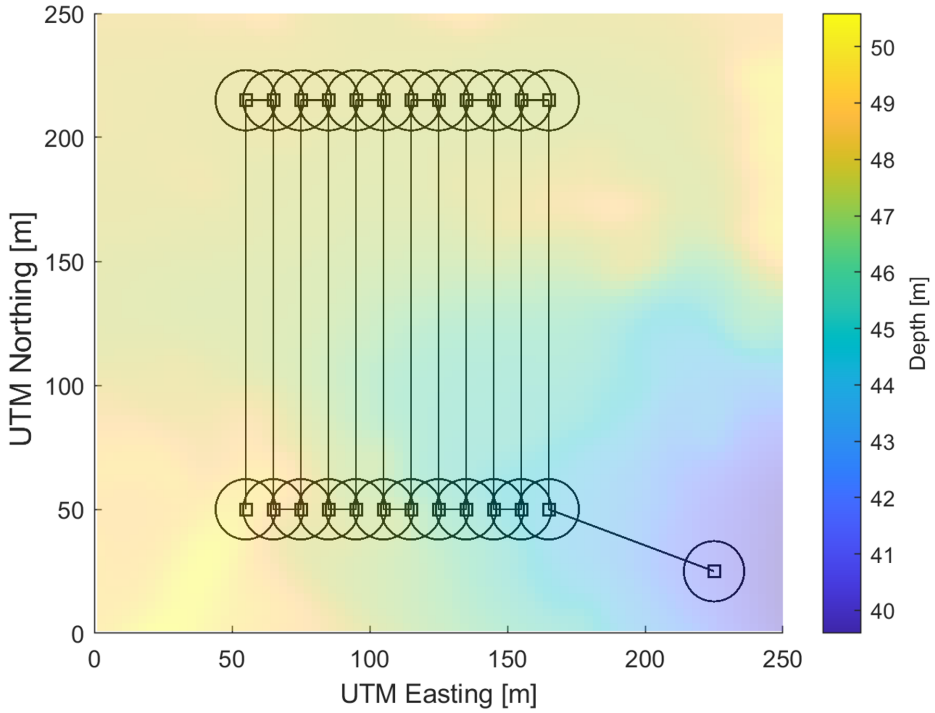



Figure 6.7. Lake Crescent IT-IMM Simulation Waypoints

The PF used within each IT-IMM implementation was tuned with the settings shown in Table 6.1.

Table 6.1. IT-IMM PF Settings

| Nomenclature | Setting | Value |
|---|---|---|
| M | Number of Particles | 5000 |
| $\beta$ | Re-sampling Parameter | 0.5 |
| $\zeta$ | Re-sampling Parameter | 0.003 |
| $H_{LB}$ | Shannon Entropy Re-distribution Parameter | 35 |
| | Sensor Model | BlueView MB2250 |

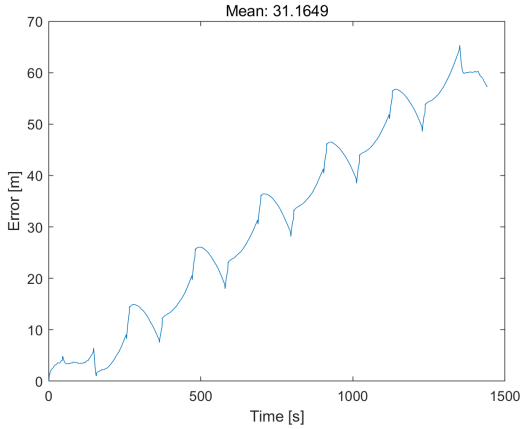
The results from each simulation are shown in Table 6.2. The EKF estimation model is estimate from EKF (1) from the block diagram. As expected, this filter accumulated a significant amount of error over time. IT-IMM 1, 2, and 3 estimation models are the estimates from EKF (3) in the block diagram. There is significant improvement in both the final positional error and the overall accumulated error from these estimates.

Table 6.2. Lake Crescent Northing Mission 2 Results

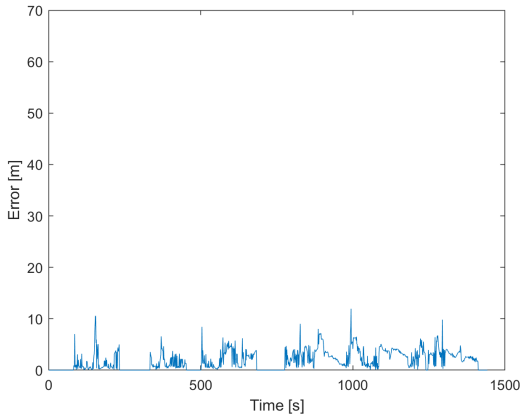
| Estimation Model | Accumulated Error [m] | Dist. from True [m] | % of Dist. Traveled |
|---|---|---|---|
| EKF | 44,940 | 55.67 | 3.5 |
| IT-IMM 1 | 8,040 | 1.68 | 0.1058 |
| IT-IMM 2 | 6,670 | 2.02 | 0.1268 |
| IT-IMM 3 | 6,123 | 1.5 | 0.0946 |

Figure 6.8 show the results of the error of the simulated REMUS at each time step. The error of the EKF is shown building over time, and is corrected slightly after each turn due to

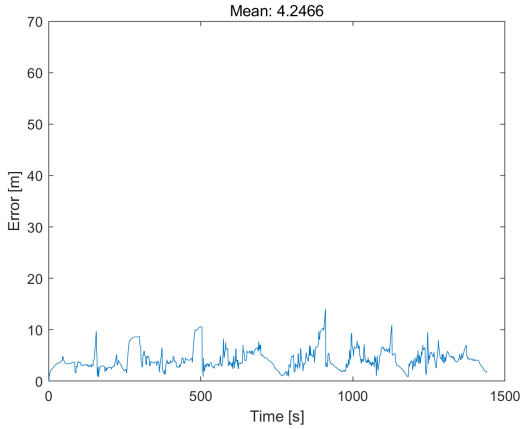
the bias in the heading. The error of the PF is much lower then the EKF but is intermittent because of the REMUS traveling through regimes in which the sensors are out of range. Finally, the error of the IT-IMM 3 show an overall lower estimation error then the EKF with no intermittent sections like the PF. The error plots of IT-IMM 1 and 2 are shown in Appendix C.




(a) EKF




(b) PF




(c) IT-IMM

Figure 6.8. Simulation Error Results for IT-IMM 3

These results highlights the ability of the IT-IMM to make intelligent decisions about federating each filter. Although simplified, the IT-IMM is a promising candidate solution for filter federation within underwater navigation and within GPS denied environments. The greatest limitation of this algorithm is that there are significant simplifications within the *State Interaction* and *Model Probability Update*. Given those simplifications, however, the results highlight a significant attribute of this algorithm which is its ability to federate two different filtering methods to provide a single best output.

THIS PAGE INTENTIONALLY LEFT BLANK

# CHAPTER 7:
# Conclusions and Future Work

The vulnerabilities and limitations of external localization methods are extremely apparent and are driving the U.S. Navy's desire for technologies that do not rely of these systems. Current underwater localization methodologies have limitations with sensor systems due to many of the regimes associated with AUV operations. This thesis proposes the solution of two different filtering methodologies federated by an IT-IMM estimation approach.

## 7.1 Thesis Conclusions

This thesis provides a model-based EKF that can provide position estimates during times that an AUV is outside of the range of its sensors. Although this method does provide accurate state estimates, it can accrue a significant amount of error if it operates under these conditions for an extended period of time.

Also developed is a PF for position estimation within ATAN which implements an information theoretic methodology for particle re-distribution. Re-sampling and an ideal number of particles were also investigated within this thesis.

Finally, a simplified IT-IMM estimation method was developed to intelligently blend two different filtering estimates. It was found that the IT-IMM improves state estimation accuracy while leveraging the attributes of each filtering method within its algorithm. The limitations of this IT-IMM are the simplifications within it.

## 7.2 Future Work

This thesis covers multiple aspects of underwater localization to include equipment, filtering methods, and filter federation. The areas for future work are organized into sections that mirror this thesis.

### 7.2.1 Numurus 3DX Camera System

The Numurus 3DX camera presents a promising sensor system that can be used to provide depth information into the PF. Future work will focus on field experimentation utilizing the 3DX as the main depth sensor system. Work must also be conducted on implementing the camera system within the PF. Finally, a more accurate estimation of velocity can be found considering the drag force added by the camera system. This can improve the estimates of the model-based EKF.

### 7.2.2 Model-based EKF

An area for future work within the model-based EKF developed in this thesis is the hydrodynamic model. Currently, the model only considers hydrodynamic forces in a two dimensional plane based on the deflection of the rudder. A more complete 6 DOF model within the EKF will yield more accurate state estimates due to more accurately modeling the non-linear system.

### 7.2.3 IT-IMM Implementation

Two major areas for future work within the IT-IMM are in removing several of the simplifications and assumptions. The simplifications within the *State Interaction* and *Model Probability Update* sections of the algorithm must be investigated. Removing these simplifications can create a more robust IT-IMM algorithm. The assumption of a fully known OSE can also be investigated.

### 7.2.4 Applications Outside of the Undersea Domain

The IT-IMM developed in this model has applicable approaches within other robotics localization and state tracking problems. A second aspect of future work is to incorporate the IT-IMM into different types of Unmanned Vehicles (UV)s.

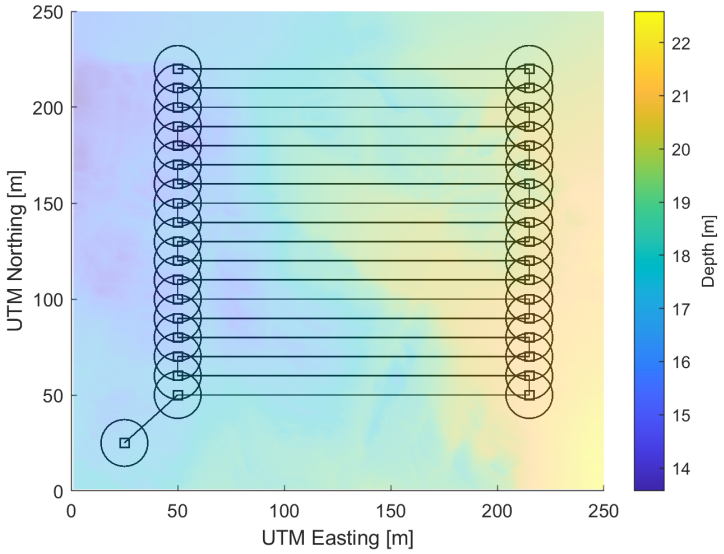
## 7.3 Concluding Remarks

An IT-IMM estimation takes advantage of the advantages of many of the attributes within its algorithm while mitigating their limitations. In this thesis we provide a contribution to

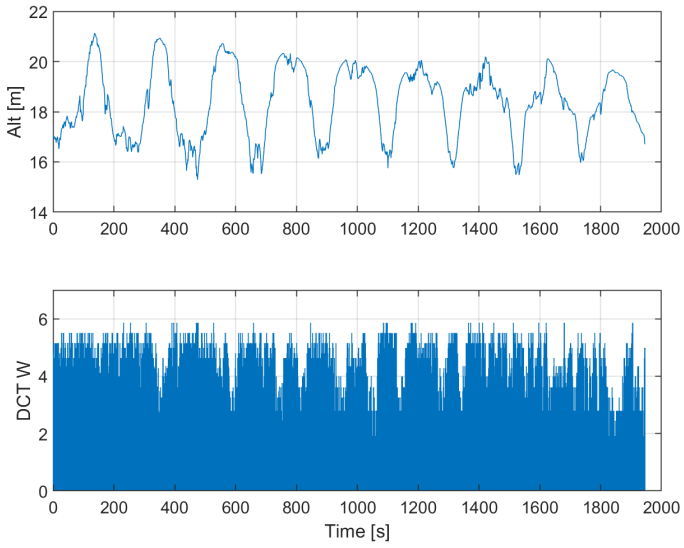
Bayesian filtering federation and a more robust AUV position estimation algorithm within GPS denied environments.

THIS PAGE INTENTIONALLY LEFT BLANK

# APPENDIX A:
# Monterey Bay AUV Mission Patterns

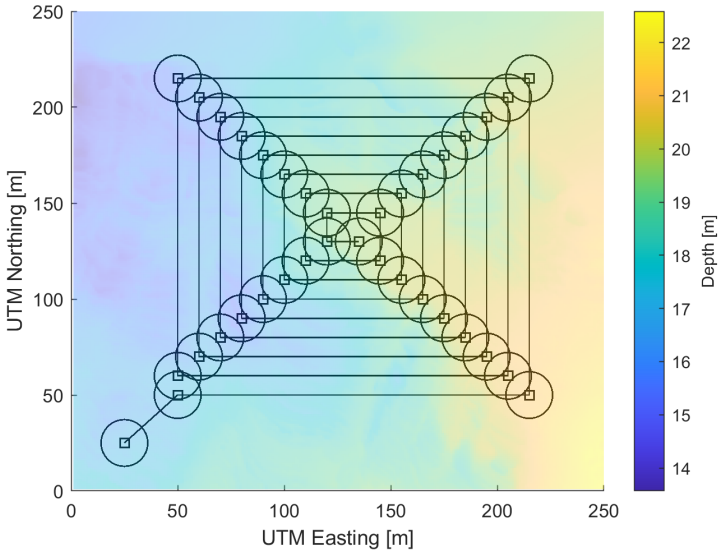



(a) East/West Parallel Track

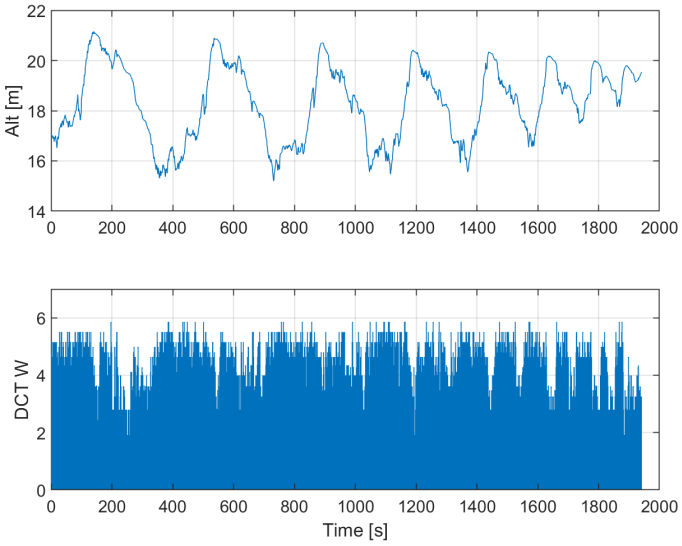



(b) East/West Parallel Alt and BE

Figure A.1. Coverage Missions




(a) Shrinking Square Track




(b) Shrinking Square Alt and BE

Figure A.2. Coverage Missions

THIS PAGE INTENTIONALLY LEFT BLANK

# APPENDIX B:
# Lake Crescent Mission Patterns

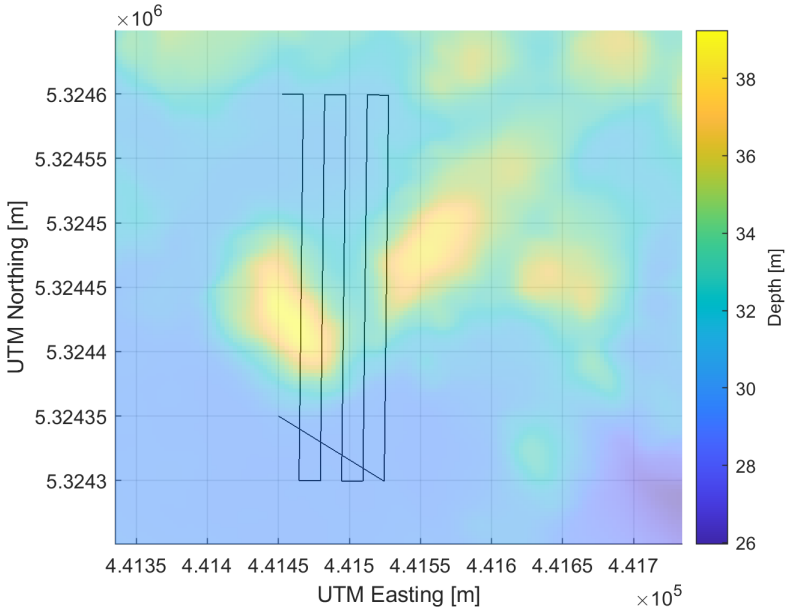



Figure B.1. Lake Crescent North/South Parallel Track 1

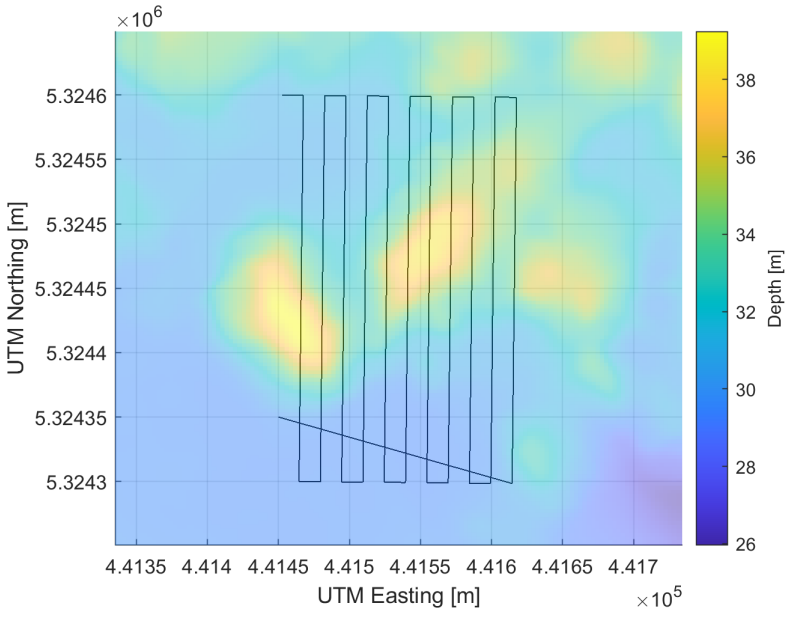



Figure B.2. Lake Crescent North/South Parallel Track 2

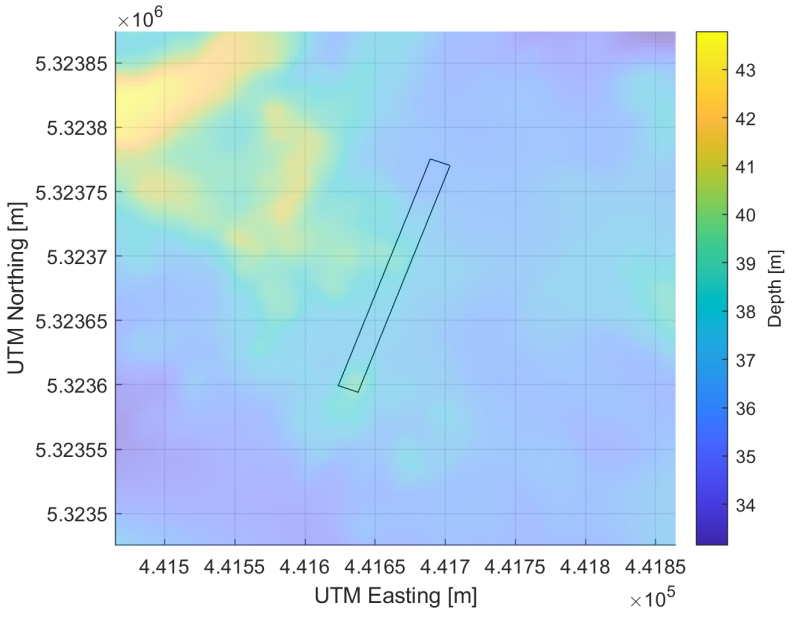



Figure B.3. L

ake Crescent ADCP TestLake Crescent ADCP Test

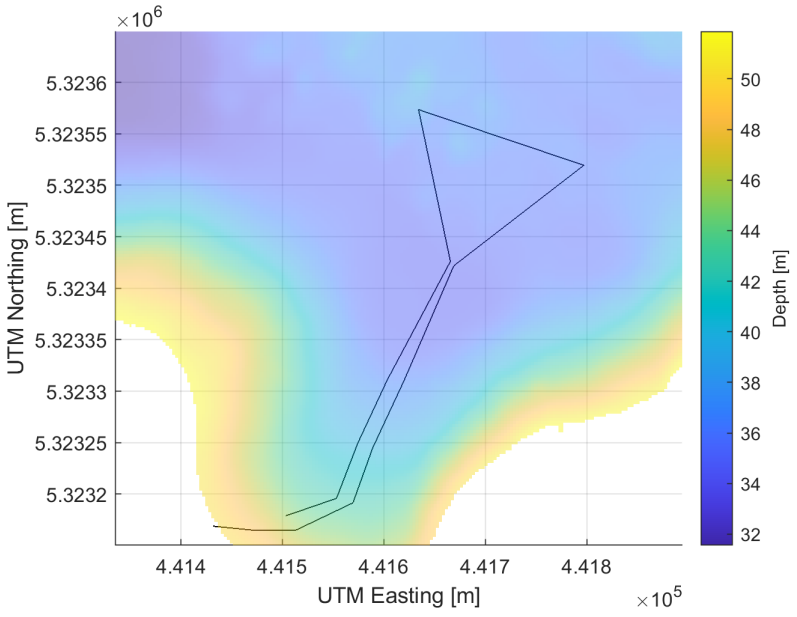



Figure B.4. Lake Crescent Triangle Mission

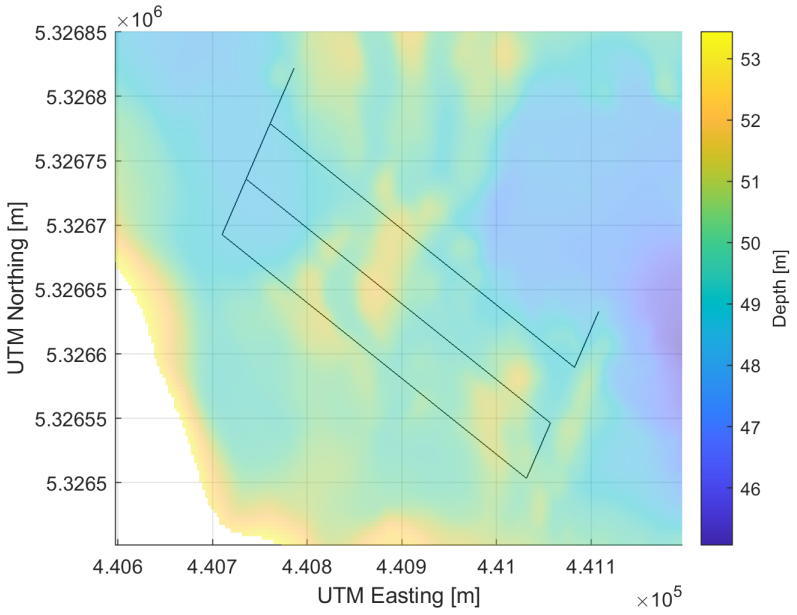
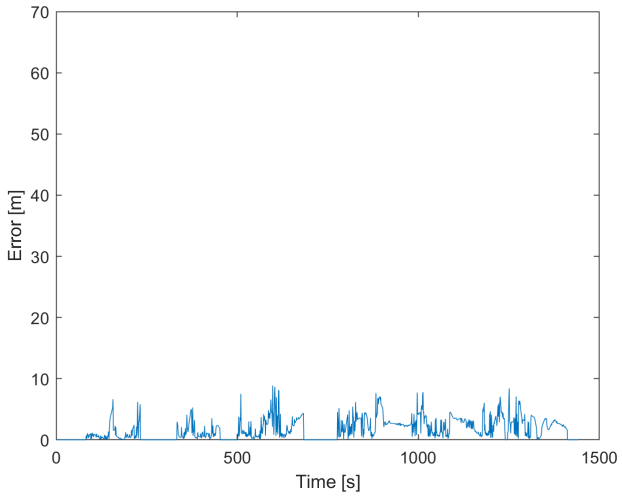


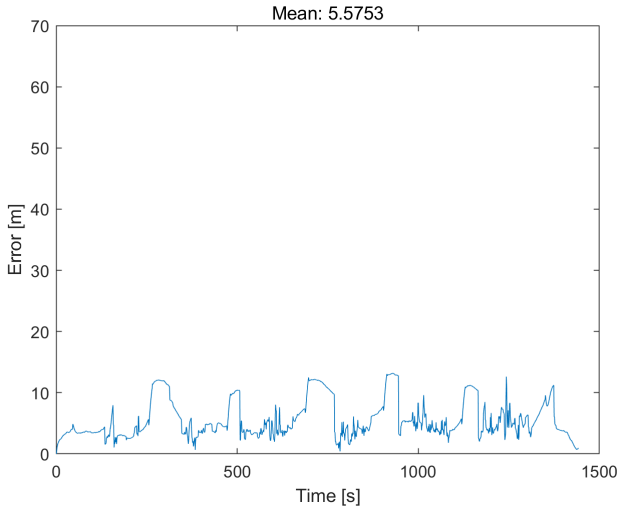

Figure B.5. Lake Crescent Box Mission

THIS PAGE INTENTIONALLY LEFT BLANK

# APPENDIX C:
# IT-IMM Results

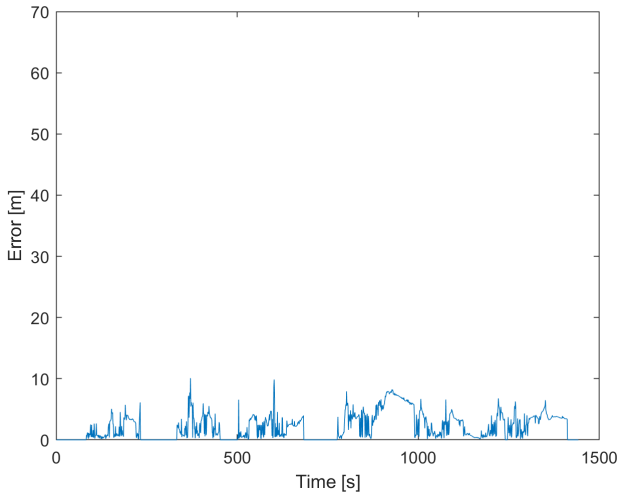



(a) PF

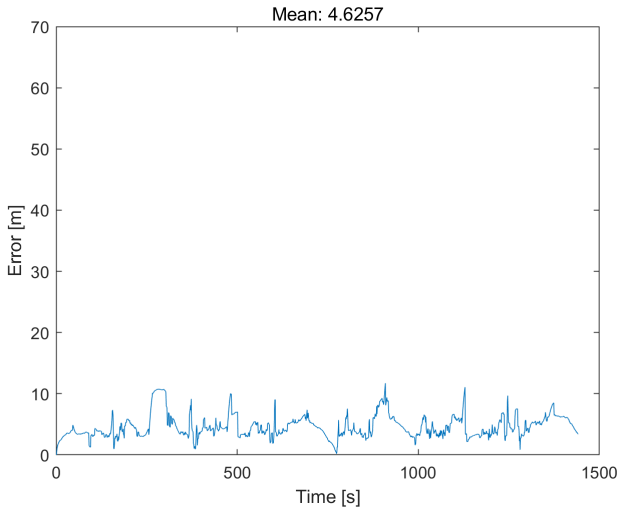



(b) IT-IMM

Figure C.1. Simulation Error Results for IT-IMM 1




(a) PF




(b) IT-IMM

Figure C.2. Simulation Error Results for IT-IMM 2

# List of References


[1] D. Glass, "What happens if GPS fails," *The Atlantic*, June 2016. [Online]. Available: https://www.theatlantic.com/technology/archive/2016/06/what-happens-if-gps-fails/486824/.

[2] K. Jindal, S. Dalal, and K. K. Sharma, "Analyzing spoofing attacks in wireless networks," *Fourth International Conference on Advanced Computing Communication Technologies*, pp. 398–402, 2014. [Online]. Available: https://doi.org/10.1109/ACCT.2014.46.

[3] T. Westbrook, "The global positioning system and military jamming: geographies of electronic warfare," *Journal of Strategic Security*, vol. 12, no. 2, pp. 1–16, 2019. [Online]. Available: https://www.jstor.org/stable/26696257.

[4] J. S. Berg, *Broadcasting on the Short Waves, 1945 to Today*. McFarland and Company, Inc Publishers, 2008.

[5] D. Kurt, "Active Terrain-Aided Navigation," M.S. thesis, Dept. of Elec. Eng., Naval Postgraduate School, Monterey, CA, USA, 2020.

[6] M. L. Psiaki, T. E. Humphreys, and B. Stauffer, "Attackers can spoof navigation signals without our knowledge. Here's how to fight back GPS lies," *IEEE Spectrum*, vol. 53, no. 8, pp. 26–53, 2016.

[7] *NOAA Diving Standards and Safety Manual*, National Oceanic and Atmospheric Administration, Washington, DC, USA, 2017. [Online]. Available: https://www.omao.noaa.gov/sites/default/files/documents/NDSSM$%$20Rev$%$2005$%$20May$%$202020$%$20clean.pdf

[8] *Positioning, Navigation, and Timing*, DOD Directive 4650.06, Office of the DoD Chief Information Officer, Washington, DC, USA, 2020. [Online]. Available: https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/465006p.pdf?ver=t6SO1Bw92MXCUnkeM2NKLg$%$3D$%$3D

[9] *Operate in absence of the global positioning system (GPS) - GPS independent navigation solutions*, Communications and GPS Navigation Program Office (PMA/A 170), Patuxent River, MD, USA, 2017.

[10] C. K. J., "Navigation sensor system interface (NAVSSI)," *Proceedings of the 1992 National Technical Meeting of The Institute of Navigation*, pp. 225–334, Jan. 1992.

[11] P. Dartnell, J. Warrick, and K. Wegmann, "Multibeam bathymetry and acoustic backscatter data," U.S. Geological Survey data release, Lake Crescent, Olympic National Park, Washington, 2016. [Online]. Available: https://www.sciencebase.gov/catalog/item/586d3165e4b0f5ce109faa51

[12] D. Kurt and D. Horner, "Undersea active terrain-aided navigation (ATAN)," *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, pp. 1–8, 2020. [Online]. Available: https://doi.org/10.1109/AUV50043.2020.9267899.

[13] W. S. Cooper, "Use of optimal estimation theory, in particular the Kalman Filter, in data analysis and signal processing," *Review of Scientific Instruments*, vol. 57, no. 2862, 1986. [Online]. Available: https://doi.org/10.1063/1.1139005.

[14] P. Del Moral, G. Rigal, and G. Salut, "Estimation and nonlinear optimal control: an unified framework for particle solutions," LAAS-CNRS, Toulouse, Research Report no. 91137, 1991.

[15] P. Zarchan and H. Musoff. *Fundamentals of Kalman Filtering: A Practical Approach*. American Institute of Aeronautics and Astronautics, Incorporated.

[16] S. J. Majumdar, C. H. Bishop, R. Buizza, and R. Gelaro, "A comparison of ensemble-transform Kalman Filter targeting guidance with ECMWF and NRL total-energy singular-vector guidance," *Quarterly Journal of the Royal Meteorological Society*, vol. 128, no. 585, pp. 2527–2549, 2002. [Online]. Available: https://doi.org/10.1256/qj.01.214.

[17] K. Nummiaro, E. Koller-Meier, and L. V. Gool, "An adaptive color-based Particle Filter," *Image and Vision Computing*, vol. 50, no. 1, pp. 99–110, 2003. [Online]. Available: https://doi.org/10.1016/S0262-8856(02)00129-4.

[18] S. Hainsworth and M. Macleod, "Particle Filtering applied to musical tempo tracking," *EURASIP Journal on Applied Signal Processing*, May 2004. [Online]. Available: https://doi.org/10.1155/S1110865704408099.

[19] K. Yao and T. Lee, "Time-varying noise estimation for speech enhancement and recognition using Sequential Monte Carlo Method," *EURASIP Journal on Applied Signal Processing*, Nov. 2004. [Online]. Available: https://doi.org/10.1155/S1110865704405022.

[20] K. Zindler, N. Geiß, K. Doll, and S. Heinlein, "Real-time ego-motion estimation using lidar and a vehicle model based Extended Kalman Filter," *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 431–438, 2014. [Online]. Available: https://10.1109/ITSC.2014.6957728.

[21] E. Herbst and F. Schorfheide, "Tempred Particle Filtering," *Penn Institute for Economic Research*, 2016. [Online]. Available: https://doi.org/10.3386/w23448.

[22] E. Ghysels and M. Marcellino, *Applied Economic Forecasting using Time Series Methods*, 2nd ed. New York, NY, USA: Oxford University Press, 2018.

[23] E. Hajiramezanali, M. Imani, and U. Braga-Neto, "Scalable optimal Bayesian classification of single-cell trajectories under regulatory model uncertainty," *BMC Genomics*, vol. 20, no. 435, June 2019. [Online]. Available: https://doi.org/10.1186/s12864-019-5720-3.

[24] S. Fuchigami, T. Niina, and S. Takada, "Particle Filter method to integrate high-speed atomic force microscopy measurements with biomolecular simulations," *Journal of Chemical Theory and Computation*, Sep. 2020. [Online]. Available: https://doi.org/10.1021/acs.jctc.0c00234.

[25] P. D. Moral, *Mean Field Simulation for Monte Carlo Integration*, 1st ed. London, UK: Chapman and Hall/CRC, 2013.

[26] J. M. Hammersley, "Poor man's Monte Carlo," *Journal of the Royal Statistical Society*, 1954. [Online]. Available: https://doi.org/10.1111/j.2517-6161.1954.tb00145x.

[27] I. Nygren, "Terrain navigation for underwater vehicles," Ph.D. dissertation, KTH, Royal Institute of Technology, School of Electrical Engineering (EES), Signal Processing, Stockholm, Sweden, 2005.

[28] C. Palmier, "New Particle Filters for underwater terrain-aided navigation using multi-sensor fusion," *HAL Open Science*, pp. 1–167, 2020. [Online]. Available: https://tel.archives-ouvertes.fr/tel-03521530.

[29] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo localization for mobile robots," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, vol. 2, 1999, pp. 1322–1328 vol. 2.

[30] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle Filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.

[31] F. J. Perez-Grau, F. Caballero, A. Viguria, and A. Ollero, "Multi-sensor three-dimensional Monte Carlo localization for long-term aerial robot navigation," *International Journal of Advanced Robotic Systems*, vol. 14, no. 5, 2017. Available: https://doi.org/10.1177/1729881417732757

[32] G. T. Donovan, "Position error correction for an autonomous underwater vehicle inertial navigation system (INS) using a Particle Filter," *IEEE Journal of Oceanic Engineering*, vol. 37, pp. 431–445, 2012. [Online]. Available: https://doi.org/10.1109/JOE.2012.2190810.

[33] R. Karlsson, F. Gusfafsson, and T. Karlsson, "Particle Filtering and Cramer-Rao lower bound for underwater navigation," *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, vol. 6, pp. VI–65, 2003.

[34] R. Karlsson and F. Gusfafsson, "Particle Filter for underwater navigation," *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, vol. 6, pp. 526–529, 2003.

[35] A. Bachmann and S. Williams, "Terrain-aided underwater navigation - a deeper insight into generic Monte Carlo localization," ARC Centre of Excellence for Autonomous Systems School of Aerospace, Mechanical and Mechatronic Engineering, University of Sydney, Sydney NSW, Australia, Tech. Rep., 2003. [Online]. Available: https://www.araa.asn.au/acra/acra2003/papers/22.pdf

[36] G. Salavasidis, A. Munfano, C. A. Harris, T. Prampart, R. Templeton, M. Smart, D. T. Roper, M. Pebody, S. D. McPhail, E. Rogers, and A. B. Phillips, "Terrain-aided navigation for long-endurance and deep-rated, autonomous underwater vehicles," *Wiley*, 2018. [Online]. Available: https://doi.org/10.1002/rob.21832

[37] N. Osterman and C. Rhén, "Exploring the sensor requirements for Particle Filter-based terrain-aided navigation in AUVs," *In 2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, pp. 1–6, 2020.

[38] M. Lager, E. A. Topp, and J. Malec, "Robust terrain-aided navigation through sensor fusion," *23rd International Conference on Information Fusion Virtual Conference*, pp. 1–8, 2020. [Online]. Available: https://doi.org/10.23919/FUSION45008.2020.9190578.

[39] A. Dardanelli, S. Corbetta, I. Boniolo, S. M. Savaresi, and S. Bittanti, "Model-based Kalman Filtering approaches for frequency tracking," *IFAC Proceedings Volumes*, vol. 43, no. 10, pp. 37–42, 2010. [Online]. Available: https://10.3182/20100826-3-TR-4015.00010.

[40] T. N. Ranjan, A. Nherakkol, and G. Navelkar, "Navigation of autonomous underwater vehicle using Extended Kalman Filter," *Trends in Intelligent Robotics: 13th FIRA Robot World Congress*, vol. 103, pp. 1–9, 2010.

[41] I. Taylor, "Variable speed hydrodynamic model of an AUV utilizing cross tunnel thrusters," M.S. thesis, Dept. of Mech. Eng., Naval Postgraduate School, Monterey, CA, USA, 2017.

[42] T. T. J. Prestero, "Verification of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle," M.S. thesis, Dept. Mech. Eng. and Ocean Eng., MIT, Cambridge, MA, USA, 2001.

[43] *Control system design and development for the REMUS autonomous underwater vehicle*, Defense Technology Agency, Auckland New Zealand, Tech. Rep., 2007.

[44] Y. Bar-Shalom and X. Li, *Estimation and Tracking: Principles, Techniques, and Software*. ARTECH House, Inc, 1993.

[45] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 103–123, 1998. [Online]. Available: https://doi.org/10.1109/7.640267.

[46] M. E. Farmer, R. Hsu, and A. K. Jain, "Interacting multiple model (IMM) Kalman Filters for robust high speed human motion tracking," *2002 International Conference on Pattern Recognition*, vol. 2, pp. 20–23, 2002. [Online]. Available: https://doi.org/10.1109/ICPR.2002.1048226.

[47] S. Vasuhi and V. Vaidehi, "Target tracking using Interactive Multiple Model for wireless sensor network," *Information Fusion*, vol. 27, pp. 41–53, 2016. [Online]. Available: https://doi.org/10.1016/j.inffus.2015.05.004.

[48] S. Li, X. Feng, Z. Deng, F. Pan, and S. Ge, "Minimum error entropy based multiple model estimation for multisensor hybrid uncertain target tracking systems," *The Institution of Engineering and Technology*, vol. 14, no. 4, pp. 199–213, 2020. [Online]. Available: https://doi.org/10.1049/iet-spr.2019.0178.

[49] T. L. Koller and U. Frese, "The Interacting Multiple Model Filter and Smoother on Boxplus-Manifolds," *Sensors*, pp. 1–19, 2021. [Online]. Available: https://doi.org/10.3390/s21124164.

[50] J. Dou and J. Li, "Robust visual tracking based on Interactive Multiple Model Particle Filter by integrating multiple cues," *Neurocomputing*, vol. 135, pp. 118–129, 2013. [Online]. Available: https://doi.org/10.1016/j.neucom.2013.12.049.

[51] T. Kim and H. Jeong, "A novel algorithm for crash detection under general road scenes using crash probabilities and an Interactive Multiple Model Particle Filter," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, pp. 2480–2490, 2014. [Online]. Available: https://doi.org/10.1109/TITS.2014.2320447.

[52] A. Genovese, "The Interacting Multiple Model algorithm for accurate state estimation of maneuvering targets," *John's Hopkins APL Technical Digest*, vol. 22, no. 4, pp. 614–623, 2001.

[53] L. A. Johnston and V. Krishnamurthy, "An improvement to the Interacting Multiple Model (IMM) algorithm," *IEEE Transactions on Signal Processing*, vol. 49, pp. 2909–2923, 2001. [Online]. Available: https://doi.org/10.1109/78.969500.

[54] W. Li and Y. Jia, "An information theoretic approach to Interacting Multiple Model Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 1811–1825, 2015.

[55] E. B. Bermudez, "Terminal homing for autonomous underwater vehicle docking," M.S. thesis, Dept. of Mech. Eng., Naval Postgraduate School, Monterey, CA, USA, 2016.

[56] J. T. Juriga, "Terrain-aided navigation for REMUS underwater autonomous vehicle," M.S. thesis, Dept. of Mech. Eng., Naval Postgraduate School, Monterey, CA, USA, 2014.

[57] *3DX-CI-1K-MKS User Manual*, 3rd ed., Numurus: Smart System Solutions, 2021.

[58] N. A. McChesney, "Three-dimensional feature reconstruction with dual forward looking sonars for unmanned underwater vehicle navigation," M.S. thesis, Dept. of Elec. Eng., Naval Postgraduate School, Monterey, CA, USA, 2009.

[59] *Numurus 3DX Product Catalogue*, Numurus: Smart System Solutions.

[60] "Autopilots," class notes for state space derivation of an AUV steering model, Department of Mechanical Engineering, Naval Post Graduate School, Monterey, CA 93940, 2021.

[61] P. Tydingco, "The use of epi-splines to model empirical semivariograms for optimal spatial estimation," M.S. thesis, Dept. of Mech. Eng., Naval Postgraduate School, Monterey, CA, USA, 2016.

[62] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, 1948. [Online]. Available: https://doi.org/10.1002/j.1538-7305.1948.tb01338.x.

[63] V. Britanak, P. Yip, and K. R. Rao, *Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms and Integer Approximations*. Academic Press, 2007.

[64] S. M. Ali and S. D. Silvey, "A general class of coefficients of divergence of one distribution from another," *Journal of the Royal Statistical Society*, pp. 131–142, 1993.

[65] K. P. Murphy, *Machine learning: A Probabilistic Perspective*. Cambridge, MA: MIT Press, 2013.

[66] A. Gelb, J. F. Kasper, R. A. Nash, C. F. Prince, and A. A. Sutherland. *Applied Optimal Estimation*. Massachusetts Institute of Technology Kirtley, Cambridge, MA, USA.

[67] Y. Kim and H. Bang. *Introduction to Kalman Filter and Its Applications*. IntechOpen 2018 [Online]. Available: https://www.intechopen.com/chapters/63164

[68] C. Montella, "The Kalman Filter and Related Algorithms: A Literature Review," *ResearchGate*, 2011.

[69] B. A. McElhoe, "An assessment of the navigation and course corrections for a manned flyby of mars or venus," *IEEE Transactions on Aerospace and Electronic Systems*, 1996. [Online]. Available: https://doi.org/10.1109/TAES.1966.4501892.

[70] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman Filter to nonlinear systems," *The Robotics Research Group, Department of Engineering Science, The University of Oxford*, 1997. [Online]. Available: https://doi.org/10.1117/12.280797.

[71] "Unscented Kalman Filter Tutorial," class notes for Unscented Kalman Filtering, Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY 14260, [Online]. Available: https://cse.sc.edu/ terejanu/files/tutorialUKF.pdf.

[72] Y. Shi, H. Li, X. Li, and L. Wang, "Application of federated Particle Filters to SINS-GPS/BDS integrated navigation system," *Computer modeling and New technologies*, vol. 18, pp. 188–191, 2014. [Online]. Available: https://doi.org/10.1007/s11771-010-0556-7.

[73] R. C. Rao, "Rao-Blackwell theorem," *Scholarpedia*, vol. 3, no. 8, p. 7039, 2008. [Online]. Available: https://doi.org/10.4249/scholarpedia.7039

[74] D. Blackwell, "Conditional expectation and unbiased sequential estimation," *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 105–110, 1947 [Online]. Available: https://doi.org/10.1214/aoms/1177730497.

[75] A. Doucet, N. de Freita, K. Murphy, and S. Russell, "Rao-Blackwellised Particle Filtering for dynamic Bayesian networks," *Uncertainty in Artificial Intelligence Proceedings*, pp. 176–183, 2000. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1301/1301.3853.pdf.

[76] H. Cramér. *Mathematical Methods of Statistics*. Princeton, NJ: Princeton Unv. Press 1946. [Online]. Available: https://archive.org/details/in.ernet.dli.2015.223699.

[77] N. Bergman, "Recursive Bayesian estimation: navigation and tracking applications," in *Proc. of the Natl. Acad. of Sci. of the USA*, 1999. [Online]. Available: https://www.semanticscholar.org/paper/Recursive-Bayesian-Estimation-3A-Navigation-and-Bergman/93a1955bc281164a6e3d71d4ead67f5215aad910

[78] T. Kloek and H. K. van Dijk, "Bayesian estimates of equation system parameters: an application of integration by Monte Carlo," *Journal of the Economic Society*, vol. 46, no. 1, pp. 1–19, Jan 1978. [Online]. Available: https://www.jstor.org/stable/1913641.

[79] "Importance Sampling and Particle Filtering," class notes for Particle Filtering, Department of Electrical Engineering, Iowa State University, Ames, IA, spring 2008.

[80] T. Li, M. Bolić, and P. Djurić, "Resampling methods for Particle Filtering: classification, implementation, and strategies," *IEEE Signal Processing Magazine*, 2015.

[81] L. Hou, X. Xu, Y. Yao, and D. Wang, "An M-estimation-based Improved Interacting Multiple Model for INS/DVL Navigation Method," *IEEE Sensors Journal*, 2015.

[82] S. P. Won, W. Melek, and F. Golnaraghi, "Fastening tool tracking system using a kalman Filter and Particle Filter combination," *Measurement Science and Technology*, vol. 22, 2011, [Online]. Available: https://doi.org/10.1002/j.1538-7305.1948.tb01338.x.

[83] E. N. Chatzi and A. W. Smyth, "The Unscented Kalman Filter and Particle Filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing," *Structural Control and Health Monitoring*, 2009, [Online]. Available: https://doi.org/10.1002/stc.290.

[84] Y. Xu, K. Xu, J. Wan, Z. Xiong, and Y. Li, "Research on Particle Filter tracking method based on Kalman Filter," *IEEE Advanced Information Management, Communities, Electronic and Automation Control Conference (IMCEC)*, vol. 2, pp. 1564–1568, 2018, [Online]. Available: https://doi.org/10.1109/IMCEC.2018.8469578.

[85] L. Li, H. Ji, and J. Lou, "The iterated Extended Kalman Particle Filter," *IEEE International Symposium on Communications and Information Technology (ISCIT)*, pp. 1213–1216, 2005, [Online]. Available: https://doi.org/10.1109/ISCIT.2005.1567087.

[86] I. Ullah, Y. Shen, X. Su, and C. Esposito, "A localization based on Unscented Kalman Filter and Particle Filter localization algorithms," *IEEE Access Green Communications on Wireless Networks*, 2019, [Online]. Available: https://doi.org/10.1109/ISCIT.2005.1567087.

[87] I. Ullah, X. Su, J. Zhu, X. Zhang, D. Choi, and Z. Hou, "Evaluation of localization by Extended Kalman Filter, Unscented Kalman Filter, and Particle Filter-based techniques," *Hindawi Wireless Communications and Mobile Computing*, vol. 2020, 2020, [Online]. Available: https://doi.org/10.1155/2020/8898672.

[88] S. P. Won, W. Melek, and F. Golnaraghi, "Position and orientation estimation using Kalman Filtering and Particle Filtering with one IMU and one position sensor," *IEEE Trans. Ind.*, pp. 3006–3010, 2008.

[89] H. Shariati, H. Moosavi, and M. Danesh, "Application of Particle Filter combined with extended Kalman Filter in model identification of an autonomous underwater vehicle based on experimental data," *Applied Ocean Research*, vol. 82, pp. 32–40, 2019, [Online]. Available: https://doi.org/10.1016/j.apor.2018.10.015.

[90] SNAME, "Nomenclature for treating the motion of a submerged body through a fluid," *The society of Naval architects and Marine engineers, Technical and Research Bulletin*, 1950. [Online]. Available: https://ci.nii.ac.jp/naid/10026864886/en/.

[91] H. Fischer. *A History of the Central Limit Theorem: From Classical to Modern Probability Theory*. Springer, New York Dordrecht Heidelberg London, 2010 [Online]. Available: http://www.medicine.mcgill.ca/epidemiology/hanley/bios601/GaussianModel/HistoryCentralLimitTheorem.pdf

[92] F. M. Dekking, C. Kraaikamp, H. P. Lopuhaä, and L. E. Meester. *A Modern Introduction to Probability and Statistics*. Springer Texts in Statistics. London: Springer London [Online]. Available: https://doi.org/10.1007/1-84628-168-7.

THIS PAGE INTENTIONALLY LEFT BLANK

# Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California