Maritime Intent Estimation and the Detection of Unknown Obstacles

by

Edward Hsiang Lung Fong

B.S. Computer Science
United States Naval Academy, 2002

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN AERONAUTICS AND ASTRONAUTICS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2004

Author ______________________________
Department of Aeronautics and Astronautics
May 14, 2004

Certified by ______________________________
Michael E. Cleary, Ph.D.
Charles Stark Draper Laboratory
Technical Supervisor

Certified by ______________________________
Professor Leslie Pack Kaelbling
Professor of Computer Science and Engineering
Thesis Advisor

Accepted by ______________________________
Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

[This page intentionally left blank]

Maritime Intent Estimation and the Detection of Unknown Obstacles

by

Edward Hsiang Lung Fong

Submitted to the Department of Aeronautics and Astronautics on
May 14, 2004, in partial fulfillment of the requirements for the
Degree of Master of Science in Aeronautics and Astronautics

Abstract

The benefits of using Unmanned Undersea Vehicles (UUVs) in maritime operations are numerous. However, before these benefits can be realized, UUV capabilities must be expanded. This thesis focuses on improving certain aspects of the Maritime Reconnaissance and Undersea Search and Survey capabilities of a UUV. An algorithm is first presented which provides the UUV with the ability to estimate the intent of the contacts it is observing (*intent estimation*). This was accomplished by developing a probabilistic model of the contact's possible intents and then using those models to estimate the contact's actual intent. The results from that algorithm are used to analyze the contact's observed path to determine a probabilistic belief of the potential location of obstacles in the environment (*obstacle detection*) that the contact is avoiding. These values are recorded in an obstacle inference map which is capable of incorporating the results from the analysis of any number of observed paths from multiple contacts. The laws of probability were used to develop the algorithms in this thesis—with an emphasis on Bayes' rule. Various scenarios are presented to demonstrate the capabilities and limitations of the *intent estimation* and *obstacle detection* algorithms.

Technical Supervisor:	Michael E. Cleary, Ph.D.
Title:	Principal Member of Technical Staff The Charles Stark Draper Laboratory, Inc.
Thesis Advisor:	Professor Leslie Pack Kaelbling, Ph.D.
Title:	Professor of Computer Science and Engineering

[This page intentionally left blank]

ACKNOWLEDGMENT

This thesis would not have been possible without the inspiration and support of many individuals whom I have had the pleasure of knowing and who have helped me get to where I am today.

I owe a debt of gratitude to my technical supervisor at Draper Laboratory, Michael Cleary, for helping me through this thesis. The advice and guidance he gave me during the last two years proved to be indispensable. His patience with and trust in me were remarkable, and the extra support he provided (especially towards the end) went well beyond what I deserved.

I would also like to extend my gratitude to Professor Leslie Kaelbling for her support and guidance on this thesis. Her expertise was invaluable and even the smallest suggestion she made greatly contributed to this thesis. The time that Prof. Kaelbling made for me in her busy schedule is greatly appreciated.

Thank you to my friends and family for helping me through this challenge in my life and accommodating my unpredictable schedule with flexibility. To my newfound friends in Boston, you have all made my time here a memorable and rewarding experience. I would also like to thank my MIT, youth, and men's Bible study groups for being a pivotal part of my development and for all their prayers (especially during the writing of this thesis).

Finally, the greatest thanks are due to Him who gives me strength in everything I do (Philippians 4:13). May we all learn and grow through the challenges and opportunities He has given us.

This thesis was prepared at The Charles Stark Draper Laboratory, Inc., under Contract N0014-02-C-0191, sponsored by the Office of Naval Research.

Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings or conclusions contained herein. It is published for the exchange and stimulations of ideas.

Edward H.L. Fong
May 14, 2004

[This page intentionally left blank]

Contents

[This page intentionally left blank]

List of Figures

List of Tables

[This page intentionally left blank]

Chapter 1

Introduction

The objective of this thesis is to develop a method that will provide an Unmanned Undersea Vehicle (UUV) with the ability to estimate the intent of a vessel it is observing in a maritime environment and then use that information to infer the presence of unknown obstacles. Estimating the intent of a maritime vessel will be referred to as *maritime intent estimation* (or *intent estimation* for short) and the inference of unknown obstacles in the environment will be known as *obstacle detection.*

1.1 Problem Motivation

The rapid advances in computational power within the last decade have stimulated the interest, and made possible the development, of numerous automated systems. One area of study with a particular interest in automated capabilities is the field of robotics. Such an automated system would be able to provide numerous benefits by performing tasks which would otherwise have to be carried out by a human operator. The benefits would include freeing the human operator from performing undesirable, menial, or dangerous tasks. These activities have been referred to by many roboticists as jobs that are dirty, dull, or dangerous (*the 3 D's*) [13]. As a result, the United States military has a vested interest in automated robotic systems since they offer an enormous potential to "greatly reduce the risk to manned platforms while freeing them to perform other high priority missions." [6]

The US military has already successfully incorporated the use of numerous automated systems in many of its operations. These systems have been used "as both weapons (cruise missiles) and reconnaissance platforms (Predator UAV [Unmanned Aerial Vehicle])" [6]. The level of maturity in the automatic capabilities of these systems can be seen by the recent use of the Predator to deliver a weapon on a target [17]. However, the Predator is a human-operated, remotely controlled platform. Another UAV program currently being pursued is the X-45A which is an unmanned combat aircraft designed for "high-risk missions like the suppression of enemy air defense and precision strike." This system has recently demonstrated the ability to autonomously deliver a precision weapon on a target, requiring only an authorization from a human operator to release its weapon [1].

Alongside the military's UAV program is the US Navy's Unmanned Undersea Vehicle (UUV) program. The benefits and desired uses of UUVs in US naval operations were recognized and set forth in the US Navy's UUV Master Plan [6]. The Master Plan defined a UUV to be:

> *... a self-propelled submersible whose operation is either fully autonomous (pre-programmed or real-time adaptive mission control) or under minimal supervisory control and is untethered except for data links such as a fiber optic cable.*

An analysis of the potential benefits of incorporating UUVs in some of the Navy's high priority missions led to the creation of four desired signature capabilities. These capabilities listed in priority order are as follows:

1. Maritime Reconnaissance
2. Undersea Search and Survey
3. Communication/Navigation Aids
4. Submarine Track and Trail

The most important signature capability, maritime reconnaissance, envisioned for a UUV is the ability to "collect multidisciplinary intelligence data across the entire electromagnetic spectrum while remaining undetected by the enemy." This capability can then be used to fulfill the mission of Intelligence/Surveillance/Reconnaissance (ISR) which is critical in military operations [6].

UUVs can provide a number of advantages in performing ISR missions. For example, a UUV can be used in conjunction with manned platforms to extend the range of operations into high-risk areas or areas where the water is too shallow for conventional platforms to operate. UUVs offer a low cost, low maintenance, and expendable means for performing certain military operations which would reduce the burden and risk to US forces.

There has been a number of projects undertaken to develop and improve the signature capabilities of UUVs so they can be incorporated into the naval fleet. An example is Autonomous Minehunting and Mapping Technologies (AMMT) which "was designed to survey undersea areas and to map the terrain and any potential mine-like objects" [16]. This project was aimed at improving the undersea search and survey capability for UUVs. There are also a number of UUV projects developed by the Space and Naval Warfare Systems Command (SPAWAR) in San Diego, CA. Some of these projects include the Advanced Unmanned Search System (AUSS), Mine Neutralization Vehicle (MNV), and Remote Unmanned Work System (RUWS). However, despite these efforts, the US Navy currently has few operational UUV-based capabilities [6].

1.2 Problem Statement

This thesis contributes to the development of maritime reconnaissance and undersea search and survey signature capabilities for a UUV. The problem is that autonomous systems cannot interpret the motions of maritime vessels, nor figure out where obstacles might be by observing the motions of surface craft.

1.2.1 Maritime Reconnaissance

The primary purpose of maritime reconnaissance for UUVs is to collect data from an assigned observation area. This data can then be used autonomously by the UUV to direct its course of action or provided to human analysts. If the UUV is to use the data it collects, it must first be able to interpret it without human assistance. The development of this capability would greatly increase the potential usefulness of a UUV by expanding the range of missions it can perform. Since there can be a number of types of useful information that the UUV may desire to extract from the data it collects, this thesis will focus on the ability to estimate the intent of a surface ship (called a *contact*) from the observations the UUV makes of its motions in the environment. With this capability, the UUV can be tasked with observing an area and responding to certain cues from the contacts in the environment. For example, the UUV may be assigned to patrol a harbor and return upon observing any suspicious activity (e.g., upon seeing a contact actively pursuing a number of other vessels) or to follow those suspicious contacts. The UUV can also use this ability to identify and maintain a safe distance from such a vessel. However, before the UUV can respond to these cues, it must be able to estimate the intent of the contacts it is observing (*intent estimation*).

1.2.2 Undersea Search and Survey

The signature capability of undersea search and survey "is an access enabler and force multiplier for the fleet, preparing the littoral undersea battlespace for entry and occupation" [6]. This desired capability includes identifying the location of hazards to navigation (obstacles) which may be unknown to US forces. These obstacles could include shipwrecks, shallow water areas, minefields, etc. Identifying the location of these obstacles may be valuable for both a follow-on force entering the area (for which the UUV was sent ahead as a scout) and the UUV itself (to determine areas it should avoid). However, since the area of interest may be fairly large and beyond the ability of a single UUV (or a small number of UUVs) to survey in a reasonable amount of time, a method to rapidly approximate the general area where obstacles may be located would be beneficial. Since it is reasonable to assume that most vessels frequenting an area would have some knowledge as to the location of obstacles and would maneuver to avoid them, a method to detect unknown obstacles in the environment by using the tracks of observed contacts would be beneficial (*obstacle detection*).

1.3 Related Work and Problem Approach

This section will present the related work and approach that will be used to develop the intent estimation and obstacle detection capabilities for a UUV.

1.3.1 Intent Estimation

Since the emphasis that has been placed on autonomous capabilities for UUVs is relatively recent, there has only been a limited amount of research performed in this area. The research that has been performed was focused on path planning considerations for a UUV [10, 16]. There has been an extensive amount of research performed in the area of activity recognition. The majority of that work has been in the field of computer vision. For example, Bobick & Davis [3], Johnson & Hogg [9], and Brand, et al. [4] present methods to probabilistically model certain activities through a learning process which requires a sizable set of data. The learned models have then been used successfully to recognize the activities of interest. However, such an extensive data set is unavailable, and difficult to obtain, for learning models of different maritime vessel intents.

A method for estimating a maritime vessel's intent was presented in Mierisch [11]. This work employed the use of Bayes' rule to successfully determine if an observed vessel had one of two intents related to the UUV. That approach will be expanded upon to explore the potential of using Bayes' rule in estimating the intent of a vessel from a domain of multiple possible intents related to various objects in its environment. Using this approach, a large data set of vessel movements will not be necessary but a reliance is made on hand-built models of how vessels with different intents would behave.

1.3.2 Obstacle Detection

Determining the location of hazards (obstacles) in an environment has been an object of research since autonomous mobile robots first became available. To detect these obstacles, the robots were provided with sensors. The data collected from these sensors were then incorporated into a map structure to represent the location of obstacles in the environment. Numerous map structures have been developed to store such information along with methods to incorporate sensor data into those maps [13]. One of the most capable and widely used of these map structures is the certainty grid, and a sound method used to incorporate sensor readings into this grid was presented by Moravec using Bayes' rule [12]. The certainty grid is a fixed rectilinear grid where each cell represents a location in the environment and indicates the likelihood that the cell is occupied. This structure will be used to represent the likelihood that a given area in the observation environment contains an obstacle. The observed path of the contact will then be used to provide the information for updating this structure using Bayes' rule.

1.4 Contributions

This thesis makes the following contributions to the development of the signature capabilities for a UUV.

1. A method for estimating the intent of the contacts in a maritime environment by analyzing their movements relative to other observed objects in the environment.

2. A method for using a contact's observed path, and its estimated intents, to infer a probabilistic obstacle map for identifying the possible locations of unknown

obstacles in the environment.

These two methods can provide the UUV with the ability to extract certain useful information from the data it collects without the assistance of human operators. That information can then be used by the UUV in determining its next course of action.

1.5 Organization of the Thesis

This chapter provided a brief introduction to the current uses of unmanned vehicles in the military operational environment. The capabilities required for such vehicles operating in an undersea environment were presented and the motivation for developing those capabilities were discussed. Chapter 2 provides a basic intent estimation model which illustrates the basic concept behind the intent estimation algorithm. Chapter 3 expands on the intent estimation model from Chapter 2 to provide a more capable model that can be used in a wider range of environments. Chapter 4 presents an obstacle detection algorithm which uses the intent estimation algorithm developed in the earlier chapters. Chapters 5 and 6 provide several simulations to illustrate the capabilities and limitations of the intent estimation and obstacle detection algorithms respectively. Chapter 7 summarizes the work presented in this thesis and provides several suggestions for future research.

[This page intentionally left blank]

Chapter 2

Basic Intent Estimation Model

When an Unmanned Undersea Vehicle (UUV) is deployed to monitor a remote region of water, it will collect various information about the environment. If equipped with the appropriate sensors (such as sonar), a UUV can detect and track certain vessels operating in the area. It would be particularly useful if the UUV could use the observed movements of the vessels to determine the larger intent of their actions.

The objective of *intent estimation* is to be able to determine the intent of an observed contact from observing its behavior. For example, is the contact trying to pull into a harbor or actively pursue another vessel in the area? Though this characteristic (which will be considered the *intent variable*) cannot be observed directly, a probability distribution over all possible contact intents can be maintained and updated with observations made of the contact's movements. The techniques of *Bayesian updating* [2] and the *forward-backward procedure* [15] are used. The *state space* [18] for the intent variable is the set of all possible intents that a contact is capable of performing. The contact is assumed to have one intent at any given time but has the ability to change its intent. In other words, the intent of the contact is only represented by one state in the variable state space but can be changed to another state at any given time. The distribution of the probability of a contact being in each state is known as the *belief state* [18] of the intent variable. The belief state thus provides a probabilistic belief of what a contact is trying to do.

This is useful because it will give the UUV the ability to develop a certain level of situational awareness on its own regarding the events unfolding in the environment it is observing. The UUV may then use that information to determine its next course of action. For example, if the UUV was assigned to monitor an area for any contacts performing a defined suspicious activity and detected such an activity, its next course of action might be to follow that contact. Even though a UUV is used to motivate the development of the intent estimator, it does not appear in any of the figures or calculations in this thesis. The contact is the actor of interest and the targets are its possible targets. The UUV would be the observer and would create and maintain the models. If the UUV thought that contact could detect it, then the UUV would include itself in the list of possible targets. In addition, the use of the intent estimator is not limited to a UUV. The estimator can be used by any vessel (or air or ground station) desiring to determine the intent of a contact and which is able to maintain a model of the environment.

This chapter will define and describe the state space for the intent variable and demonstrate how the belief state for the intent variable is updated using a basic intent estimation model. This model will then be improved in Chapter 3.

2.1 Definitions

In the context of this thesis the following definitions are used:

Contact - The vessel that the UUV is focused on (i.e., the UUV is trying to determine what the contact is doing).

Behavior - An action that the contact can undertake in relation to another vessel, or object, of interest (e.g., the behavior of the contact is to *intercept* some vessel). The modeled behaviors are described in Section 2.3.2.

Target - An object or another vessel that the contact could have an interest in.

Intent - A behavior/target combination describing what the contact could be trying to accomplish (e.g., the intent of the contact is to *follow target B*).

2.2 Assumptions

To focus this research on the development of the intent estimator, the following major assumptions are made. Other assumptions are mentioned in the appropriate sections.

1. The UUV can sense what the intent estimator needs as inputs and is capable of providing the intent estimator with detailed and accurate information concerning the contacts in the environment. Such information includes the (x, y) position of the contact along with the contact's heading. The location of the contact can be determined by a number of methods such as repositioning the UUV to obtain a positional fix from two passive sonar readings, incorporating the sensor readings from multiple UUVs working in collaboration with each other, or the employment of an active sonar. Passive sonar operates by detecting an acoustic signal from a vessel to provide its bearing in relation to the position of the sensor. However, since it is unable to provide a range measurement to the vessel, two or more readings are required for a fix. Active sonar is able to determine the distance and bearing of an object. This is accomplished by first emitting a sound pulse and then measuring the time it takes the pulse to bounce off the object and return to the sonar (the elapsed time). The distance to the object can then be calculated using the elapsed time and the known speed at which the pulse travels through the water. The bearing of the object is determined by the direction where the pulse was emitted and received.

   In addition, any information regarding vessels in the environment that are not detected by the UUV is considered unnecessary to the objective of the intent estimator. For example, if the UUV only uses a passive sonar to detect contacts in the environment, it will almost certainly not be able to sense ("acquire") and track a sailboat that is operating without a motor. In such a scenario, the UUV would not be concerned with sailboats in the environment.

2. The intent estimator can accurately predict the movements a contact would demonstrate if it had a given intent. In other words, if the UUV believes that the contact's intent is to intercept a certain vessel, the UUV can predict the resulting future position and heading of the contact. A path planner is incorporated into the intent estimator to provide the UUV with this capability. It is assumed that this path planner and its usage accurately models the behavior of the contact. The path planner is described in Appendix A.

3. The state space of the intent variable is limited to the combinations of behaviors modeled and the targets that the UUV can identify, and is comprehensive. This assumes that the contact does not perform an intent that is not in the state space (i.e., does not have an intent that cannot be modeled).

2.3 Contact Intents

Before the intent of any contact can be determined, the intent state space must be defined. Since an intent is defined as a behavior/target pair, the behavior and target domains must be defined. This section presents the target and behavior domains and then describes how the intent state space is derived.

2.3.1 Targets

Combining the definition and assumptions related to the target, which are made in Sections 2.1 and 2.2 respectively, the domain is comprised of known environmental objects (e.g., buoys, which will be represented by a point) and known vessels. Therefore, the target domain is

$$\mathfrak{T} = O \bigcup V, \quad \text{where} \quad \begin{aligned} O &= \{\text{objects}\} \\ V &= \{\text{vessels}\} \end{aligned}$$

and the individual target is

$$\boldsymbol{\mathfrak{t}}_i \in \boldsymbol{\mathfrak{T}}, \quad 1 \leq i \leq |O| + |V|.$$

2.3.2 Behaviors

The behavior domain is comprised of actions that the contact can perform with regards to the available target types (an object (O) or vessel (V) of interest in the environment).

At the grossest level, almost all the behaviors of sea-faring vessels can be described as a series of instances of a single action—moving from point A to point B. For the contact's immediate path, point A is its current position but point B can be less apparent depending on the contact's intent. For example, if a contact was tasked with entering a certain harbor, point B is easily identified as being the location of the targeted harbor. However, if the contact was assigned to patrol a harbor, it could move in a circuitous path consisting of a number of waypoints—each of which is difficult for an observer to determine ahead of time and could change from one circuit to the next. Therefore, to simplify the complexity of the behavior models and reduce the size of the domain, the only behaviors that are modeled are the ones in which a destination is easily identifiable. These behaviors (one for stationary targets and two for moving targets) are identified and described as follows:

Approach - The approach behavior is only valid for stationary targets (a pier, the beginning of a channel, etc.) with the simple goal of just reaching the target location. For this behavior the destination would be a designated (x, y) position representing the target.

Intercept - The intercept behavior is only valid for moving targets (other vessels). In this situation, the goal of the contact is to move along a path that will bring it next to the target vessel. The destination would be the closest point where the two vessels would meet if the target vessel continued on a straight-line path with constant speed and the contact was traveling at a predetermined speed.

Follow - The follow behavior is only valid for moving targets (other vessels). In this situation, the contact is modeled to position itself to be somewhere near (most likely behind) the contact without necessarily being right next to it. To keep the behavior simple, this was modeled by setting the destination to be the target's current position—which at the next time step will be behind the target.[1]

2.3.3 Intent State Space

Now that the target and behavior domains are defined, the intent state space ($\mathfrak{I}$) can be defined as the set of all valid behavior/target combinations:

$$\mathfrak{I} = \{behaviors_S \times O\} \cup \{behavior_M \times V\},$$

where $behaviors_S$ are behaviors valid for stationary targets, objects (O), and $behavior_M$ are behaviors valid for moving targets, vessels (V). The size of the state space is

$$|\mathfrak{I}| = |behaviors_S||O| + |behavior_M||V|.$$

Each intent will be represented by its behavior and target elements as follows: *Behavior(Target)*. For example, an intent to "Approach Object A" will be represented as *Approach(A)*. It is then the task of the intent estimator to calculate the belief state for this state space.

2.4 Intent Belief State

As mentioned earlier, the intent variable of a contact (which will also be referred to as simply the "intent of the contact" or the "contact's state") cannot be directly determined from what is observed. However, a belief state can be maintained by

[1] Mierisch describes a more detailed hostile follow behavior [11].

observing the contact and comparing those observations to what is expected from each intent model. Let a system be defined with N distinct intent states

$$\mathcal{I} \in \mathfrak{I} = \{\mathbf{i}_1, \mathbf{i}_2, \ldots, \mathbf{i}_N\},$$

with $\mathcal{I}$ being a random variable used to represent the intent state of the contact. For example $\mathcal{I} = \mathbf{i}_i$ would indicate that the contact has intent $\mathbf{i}_i$. The observation made of the contact will be represented by the variable

$$\mathcal{O} = (x, y, \theta),$$

and will be referred to as the observation variable. The observation variable consists of three variables used to represent the contact's *pose*. The contact's pose is defined as the contact's (x, y) location and heading (θ). The domain for x and y is $[-\infty, \infty]$, and the domain for θ is [0°, 360°). The observation state space is

$$\mathfrak{O} = \{(x, y, \theta) : x, y \in \mathbf{R} \text{ and } \theta \in \mathbf{A}\},$$

where **R** denotes the set of real numbers and **A** denotes the set of real numbers in the domain [0°, 360°). The heading used in this thesis is an absolute degree (deg) measurement where 0° is North and the values increase in a clockwise direction as in a normal compass rose. North is defined along the positive y-axis and East is along the positive x-axis. Although the specific intent ($\mathcal{I}$) of the contact cannot be directly determined from the observation made ($\mathcal{O}$), the probability that the contact has a certain intent can be calculated if the probability that the observation $\mathcal{O}$ will be made is known for each intent. Bayes' rule [2] provides the relationship between these two probabilities and in the most general case can be written as

$$P(\mathcal{I} = \mathbf{i}_i | \mathcal{O}) = \frac{P(\mathcal{I} = \mathbf{i}_i) P(\mathcal{O} | \mathcal{I} = \mathbf{i}_i)}{\sum_{j=1}^{N} P(\mathcal{I} = \mathbf{i}_j) P(\mathcal{O} | \mathcal{I} = \mathbf{i}_j)} \tag{2.1}$$

1. Make an observation of the contact ($\mathcal{O}_{u-1}$).
2. Predict the contact's path for each intent using (1).
3. Wait until the next contact observation ($\mathcal{O}_u$) is made.
4. Determine the elapsed time between the two observations (τ_u).
5. Calculate the probability of $\mathcal{O}_u$ using (2) and (4).
6. Determine the belief state ($P(\mathcal{I}_u)$) using (5) and the previous belief state ($P(\mathcal{I}_{u-1})$).

Figure 2-1: General update algorithm for update u.

where $P(\mathcal{I} = \mathbf{i}_i|\mathcal{O})$ is the probability that the contact has intent $\mathbf{i}_i$ given the observation $\mathcal{O}$, $P(\mathcal{I} = \mathbf{i}_i)$ is the prior belief that the intent of the contact was $\mathbf{i}_i$, and $P(\mathcal{O}|\mathcal{I} = \mathbf{i}_i)$ is the probability that the observation $\mathcal{O}$ would be made given intent $\mathbf{i}_i$. By calculating the probability value for each intent in the state space, the belief state for the intent variable ($\mathcal{I}$) is obtained. The belief state can be updated iteratively with each additional observation made of the contact. This is accomplished by using values from the prior belief state to calculate the current belief state. This calculation will be referred to as an *update* of the intent belief state.

Each update will be represented by the variable u, with the first update being $u = 1$. When necessary, a variable will have a subscript with the update when the value was obtained. For example, the observation from update u would be represented by $\mathcal{O}_u$ and the state of the contact at update $u - 1$ would be $\mathcal{I}_{u-1}$. The elapsed time between updates will be represented by the variable τ. This value may be fixed or variable and depends on whether the UUV will make observations of the environment at fixed intervals. As a result, τ may be different for each update and will be uniquely identified as τ_u.

The general algorithm used to perform an update (at u) is provided in Figure 2-1. Steps 1–4 encompass the required actions needed to perform step 5—the calculation of the probability that the observation ($\mathcal{O}_u$) will be observed using each of the contact's intents. This calculation is based on the predicted path of the contact generated by the intent estimator using the previous observation ($\mathcal{O}_{u-1}$). The elapsed time (τ_u) between the two observations is also required for the calculation. The resulting probability was represented by $P(\mathcal{O}|\mathcal{I} = \mathbf{i}_i)$ in Equation 2.1 and will be rewritten as $P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})$ to reflect all the variables used in calculating the probability value. This term will be referred to as the *observed intent probability distribution function* (OI) since it provides a probability distribution over the *observation space* ($\mathbb{O}$) of how likely $\mathcal{O}_u$ will be made for a specific intent. The previous belief state values used in step 6 are represented by the $P(\mathcal{I} = \mathbf{i}_i)$ term in Equation 2.1 and will be rewritten as $P(\mathcal{I}_{u-1} = \mathbf{i}_i)$. This notation is used to indicate that the value provided is the probability of the *previous* intent state, $P(\mathcal{I}_{u-1})$. The rewritten terms can be used to replace their original representations in Equation 2.1 to produce $P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_i|\mathcal{O}_u, \mathcal{O}_{u-1})$. This is the probability that the state of the contact $\mathcal{I}_u$ is $\mathbf{i}_i$ given the current observation made of the contact $(\mathcal{O}_u)$, the previous observation $(\mathcal{O}_{u-1})$, and the elapsed time between updates $u-1$ and u (τ_u). The complete equation can be written as

$$P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_i|\mathcal{O}_u, \mathcal{O}_{u-1}) = \frac{P(\mathcal{I}_{u-1} = \mathbf{i}_i)P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})}{\sum_{j=1}^{N} P(\mathcal{I}_{u-1} = \mathbf{i}_j)P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_j, \mathcal{O}_{u-1})}, \quad 1 \le i \le N. \quad (2.2)$$

The denominator on the right hand side of Equation 2.2 is known an a *normalizing factor* which is used to ensure that the distribution for $P_{\tau_u}(\mathcal{I}_u|\mathcal{O}_u, \mathcal{O}_{u-1})$ adds up to 1 [18]. This factor will be represented by η and used to indicate that the calculated results need to be normalized.

At the first update $(u = 1)$, the prior probability $P(\mathcal{I}_{u-1} = \mathbf{i}_i)$ is set to the initial probability of the contact being in intent state $\mathbf{i}_i$. The initial probability will be represented by π_i, where $1 \le i \le N$ and

$$\sum_{i=1}^{N} \pi_i = 1.$$

The initial probabilities are assumed to incorporate any prior information available concerning how likely the contact is to be in one state over another. If no such information is available, then all π_i terms are equal. The initial probabilities represent the initial belief state of the contact's intent and are provided externally. The set of these probabilities will be represented by Π. In addition, an initial contact observation ($\mathcal{O}_0$) needs to be made before the first update to provide the $\mathcal{O}_{u-1}$ value.

For subsequent updates ($u > 1$), the $P_{\tau_u}(\mathcal{I}_u = \mathfrak{i}_i | \mathcal{O}_u, \mathcal{O}_{u-1})$ value calculated from the previous update ($u - 1$) is used for $P(\mathcal{I}_{u-1} = \mathfrak{i}_i)$. By using each previous belief that the contact was in state $\mathfrak{i}_i$ to calculate the new value, a running belief can be maintained of what the contact is trying to do.

An initial assumption is made that the contact does not change states (i.e., the contact's intent remains the same) during the observed transit. This assumption will be removed in Section 3.2.3 where the intent estimator will have the capability to update the belief state in an environment where the contact's intent can change from one update to another.

2.5 Preliminary Algorithm Overview

A more detailed description of the general intent estimation algorithm provided in Figure 2-1 is shown in Figure 2-2. This algorithm provides the process used for updating the intent belief state from the contact observations.

The algorithm receives as inputs the intent state space ($\mathfrak{I}$) and the set of initial probabilities that the contact is in each intent state (Π). An initial observation ($\mathcal{O}_o$) is made and used to create an *intent model* ($M_i[0]$) of how the contact would perform a specific intent, $\mathfrak{i}_i$ (lines 2–8). This was performed using CREATE-INTENT-MODEL, which is described in Section 2.6.1. Since the intent estimator can only update the intent belief state using another contact observation, the process is paused until the observation of the contact is made (line 11). Once an observation is made, the time elapsed (τ) since the previous observation is calculated (lines 14–15). The elapsed time (τ) is therefore determined by the rate at which the UUV makes observations of the contact—which is determined externally. The elapsed time is then used in conjunction with each intent model to determine the expected observation value for each intent

```
Intent-Estimation($\mathfrak{I}$, $\Pi$)
$u \leftarrow 0$
$\mathcal{O}_u \leftarrow$ obtain contact's initial position and heading
$\mathcal{T}_u \leftarrow$ initial simulation time
for each $i$ in $\mathfrak{I}$
    do
        $P(\mathcal{I}_u = \mathfrak{i}_i) \leftarrow \pi_i$
        $M_i[u] \leftarrow$ Create-Intent-Model($\mathcal{O}_u, \mathfrak{i}_i$)
    done
while 1 // intent estimator runs
    do
        pause // wait until the next contact observation is made
        $u \leftarrow u + 1$
        $\mathcal{O}_u \leftarrow$ obtain contact's current position and heading
        $\mathcal{T}_u \leftarrow$ current simulation time
        $\tau_u = \mathcal{T}_u - \mathcal{T}_{u-1}$
        for each $i$ in $\mathfrak{I}$
            do
                $\mu_i[u] \leftarrow$ Extract-Ideal-XY$\theta$($M_i[u-1], \tau_u$)
                $P(\mathcal{O}_u|\mathfrak{i}_i, \mathcal{O}_{u-1}) \leftarrow$ Observed-Intent-PDF($\mathcal{O}_u, \mu_i[u]$)
            done
        for each $i$ in $\mathfrak{I}$
            do
                $P(\mathcal{I}_u = \mathfrak{i}_i) \leftarrow$ Calc-Belief-State($P(\mathcal{I}_{u-1}), P(\mathcal{O}_u|\mathcal{I}_u, \mathcal{O}_{u-1}), \mathfrak{i}_i$)
                $M_i[u] \leftarrow$ Create-Intent-Model($\mathcal{O}_u, \mathfrak{i}_i$)
            done
    done
```

Figure 2-2: Algorithm for a basic intent estimation model.

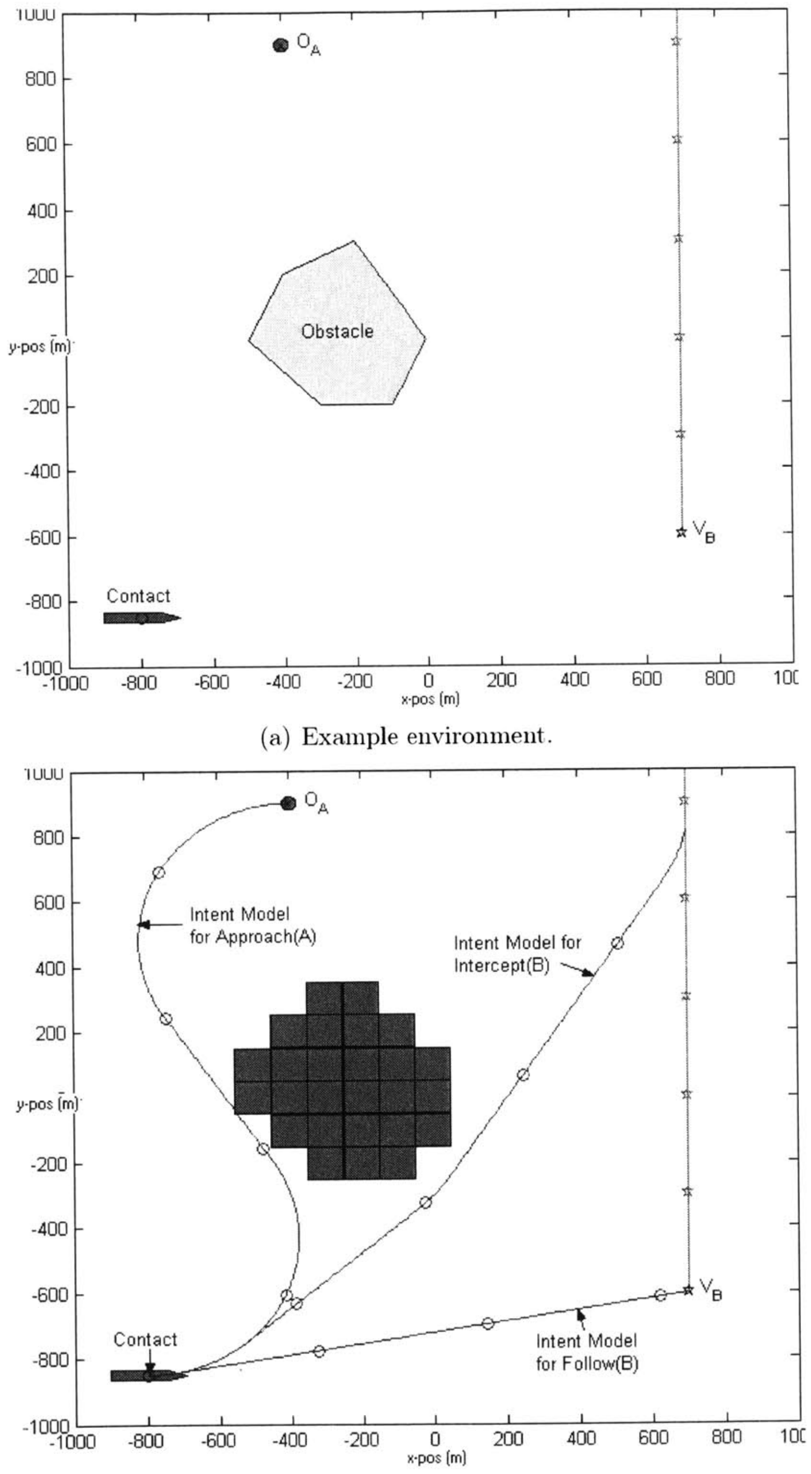
using EXTRACT-IDEAL-XYθ (line 18), described in Section 2.6.2. The expected observation value is what the intent model predicted that $\mathcal{O}_u$ would be if the contact had that intent given the elapsed time. The expected value for each observation is then used with the actual observation made to determine the probability that $\mathcal{O}_u$ would be observed for each intent model. This is calculated using OBSERVED-INTENT-PDF (line 19) which is described in Section 2.6.2. These values are then used to update the intent belief state using CALC-BELIEF-STATE (line 23), described in Section 2.4. A new intent model is then generated for each intent based on the current observation (line 24). The process then waits for the next observation to be made and lines 12–25 are repeated.

2.6 Observed Intent PDF (OI)

The Observed Intent probability distribution function (OI) in Equation 2.2 provides the capability to indirectly determine the intent belief state of a contact from observing the contact's pose at various times. As defined in Section 2.4, the OI is the probability of how well the actual observation of the contact matched what was expected from each of the intent models. This section will describe how the intent models are created, how they are used to determine the expected observation values for each intent, and finally how to compare the expected and actual observation values to obtain the probability of a match. Section 2.6.2 describes the OI and an example is provided in Section 2.6.3. A simulation run is also provided in Section 2.6.4.

2.6.1 Creating the Intent Model

Each intent model represents the contact's expected action to accomplish that intent. For example, in Figure 2-3(a), a contact is observed with a certain pose. There are two targets in the environment (O_A and V_B). The target V_B is represented by a '$\star$' symbol. The line extending up from V_B is the projected path of the vessel assuming a constant heading North. The additional $\star$ symbols along the path are the projected



(a) Example environment.

(b) Example intent models.

Figure 2-3: Path planner example.

future positions of the target at 60 second intervals (assuming a constant vessel speed). From Section 2.3, it is determined that there are three possible intents: $Approach(A)$, $Follow(B)$, and $Intercept(B)$.

The expected path (intent model) is generated using the path planner from the Autonomous Minehunting and Mapping Technologies (AMMT) program [16]. This path planner implements the A* search algorithm [14, 10]. Each path is created using the same starting contact pose and its vessel properties (speed and turn radius). However, the path planner is used slightly differently to generate each path since the desired final pose for each intent may be different. In addition, the path planner was run multiple times to generate a reasonable intent model for intercepting moving targets. The actual implementation of the path planner is provided in Appendix A. The resulting path is the intent model that represents the expected path for a contact with that intent (CREATE-INTENT-MODEL). Each generated path is represented by a list of $<x_t, y_t>$ positional pairs. Each positional pair will be identify by the subscript t and represents the predicted position of the contact at a certain time in the future according to its index (t) in the list. For example, the first positional pair ($t = 0$) in the list represents the contact's current location and the last pair represents the intent destination. The adjacent pairs in the list are separated by 1 second which allows the index to also be used for identifying the time, from when the path was generated, that the contact is predicted to be at the location represented by a given positional pair. Each path will be assigned to its respective intent model variable as follows

$$M_i[u].pos[t] = <x_t, y_t>, \quad 0 \leq t \leq \textsc{Length}(\text{Path}) - 1,$$

which represents the generated path for intent $\mathbf{i}_i$, and created using $\mathcal{O}_u$. The initial intent models will each be defined by a single path. Therefore, the terms "intent model" and "path" can be used interchangeably. Each intent model for the example illustrated by Figure 2-3(a) is shown in Figure 2-3(b).

As illustrated by the intent estimation algorithm presented in Figure 2-2, each intent model is recreated at every update using the most recent observation. This

```
EXTRACT-IDEAL-XYθ($M_i[u-1], \tau_u$)
while LENGTH($M_i[u-1]$) $< (\tau + 2)$
    do $M_i[u-1] \leftarrow$ REPEAT-LAST-POSITION($M_i[u-1]$)
$\mu_x \leftarrow$ GET-X-POSITION($M_i[u-1].pos[\tau_u]$)
$\mu_y \leftarrow$ GET-Y-POSITION($M_i[u-1].pos[\tau_u]$)
$\mu_\theta \leftarrow$ CALC-TRUE-BEARING($M_i[u-1].pos[\tau_u - 1], M_i[u-1].pos[\tau_u + 1]$)
return $\mu = (\mu_x, \mu_y, \mu_\theta)$
```

Figure 2-4: Algorithm for extracting the ideal position and heading.

is necessary since the ideal path to perform an intent will change as the situation evolves, for two reasons. The first is when the contact does not follow the intent model exactly (as can happen due to drift or steering errors) and is observed at a location off the ideal path. The second is when the contact changes its intent. In this situation, the intent model created at some previous time will not represent what the contact would do now. To allow for this variation, each intent model is redetermined using the most recent information available (at each update).

2.6.2 Using the Intent Model

Each intent model provides the ideal path for a contact performing a given intent. Given an observation, a method for determining how likely it is that the contact is on a given path (following an intent model) is required. The likelihood that the contact is following each intent model is used to calculate the intent belief state.

To estimate the likelihood, the observed pose of the contact at update u ($\mathcal{O}_u$) is compared to the pose expected from each intent model. As mentioned in Section 2.4, each observation consists of the contact's (x, y) position and heading (θ). Therefore, the contact's ideal position and heading at the time of update u is extracted from each intent model using EXTRACT-IDEAL-XYθ (Figure 2-4). This function is called by INTENT-ESTIMATION (Figure 2-2, line 18) and receives as inputs an intent model ($M_i[u-1]$) and the elapsed time since the last update (τ_u) in integer seconds. Using the path format from Section 2.6.1, the ideal position (μ_x, μ_y) using the elapsed time since $M_i[u-1]$ was created is selected from index τ_u of $M_i[u-1].pos$. Since the generated path only provides a positional pair, the ideal heading (μ_θ) is obtained by calculating the true bearing of the position from index $\tau_u + 1$ relative to the position from index $\tau_u - 1$. Since the path planner only generates a path for reaching a given destination, the length of the path may be shorter than τ_u if it contact is able to reach its target in less time than the time between the two updates (τ_u). In this situation, it is assumed that the contact will stop once it reaches its destination. Since the adjacent position pairs of the generated path are separated by 1 second, the length of the path list represents the duration of the path (-1 for the list starting at index 0). To extend the path to reflect that the contact remained at its last position upon reaching its destination, the list is extended by repeating the last positional pair of the path as necessary (Figure 2-4, lines 1–2). This would provide a positional pair representing the contact's predicted location for $\tau_u + 1$ seconds. The expected observation values are then returned as μ (Figure 2-4, line 6). Since the ideal pose at update u (μ_u) depends on $M_i[u-1]$ and τ_u, μ is fully identified as $\mu_{M_i[u-1],\tau_u}$ but will usually be referred to simply as μ.

The expected observation values (μ) from an intent model ($M_i[u-1]$) anticipates the ideal observation ($\mathcal{O}_u$) at update u that would be made of the contact if it had intent, $\dot{\mathbf{\imath}}_i$. In other words, if a contact had intent $\dot{\mathbf{\imath}}_i$, its expected position and heading at the next update will be the μ values extracted from $M_i[u-1]$. Given that intent, the probability that the contact will be observed in some other pose will decrease as the pose diverges from the expected observation. Such a probability distribution is modeled by a combination of distributions described in the following paragraph.

By making the assumption that these three parameters are independent of each other, the probability of a match for each part of the observation ($\mathcal{O} = (x, y, \theta)$) can be calculated separately and then multiplied together. Representing those parameters as independent variables is a reasonable assumption since some unmodeled environmental factors (such as the wind and ocean currents) can move the contact with respect to one parameter without affecting the others.

The x and y probabilities can be calculated using a normal distribution with means

Variable	Value
σ_x^2 (m)	200^2
σ_y^2 (m)	200^2
k	5

Table 2.1: Values for σ_x^2, σ_y^2 and k.

of μ_x and μ_y respectively and variance σ^2. For this thesis, the variances are arbitrarily defined in Table 2.1 These values have been set to reflect a standard deviation (σ) of 200 m. This indicates that the contact's position (along each axis) will be within ±200 m of the predicted position about 70% of the time. Because the range for θ does not extend from $-\infty$ to ∞ (as is the case for x and y), the normal distribution cannot be used for it. Fortunately, there is a distribution similar to the normal distribution for a "circular random variable" known as the von Mises distribution [19]. The probability distribution function for this distribution is as follows:

$$P_k(\theta) = \frac{e^{k\cos(\theta-\mu_\theta)}}{2\pi I_0(k)}, \quad \text{where } 0 \leq rad(\theta) \leq 2\pi, \text{ and } k > 0, \tag{2.3}$$

with the mean μ_θ at the ideal heading (θ). $I_0(k)$ is a modified Bessel function of the first kind and order 0:

$$I_0(k) = \sum_{r=0}^{\infty} \frac{1}{r!^2}(\frac{1}{2}k)^{2r}.$$

As k increases, the distribution narrows about the mean, μ_θ (Figure 2-5). The effect of increasing k for the von Mises distribution is similar to decreasing σ^2 for the normal distribution. In this thesis, the value of k is arbitrarily set at 5. This value was selected to reflect a belief the contact's observed heading will fall within 30° of the predicted heading about 70% of the time.

The three individual probabilities (two from the normal distribution and one from the von Mises distribution) are then multiplied together to obtain the actual probability that the observation matches the expected state, given the intent and elapsed

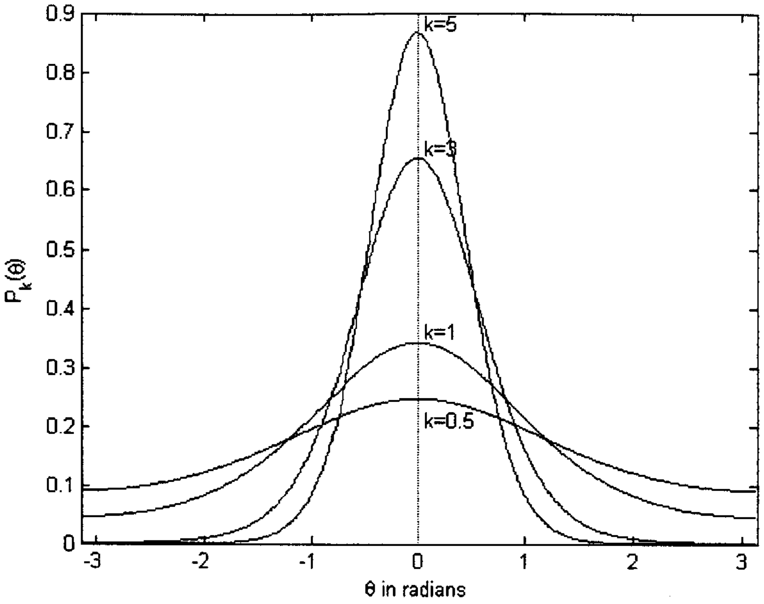


Figure 2-5: The von Mises distribution with $\mu_\theta = 0$ and various k values.

time. The OI (OBSERVED-INTENT-PDF) can be written as:

$$P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}) = P(\mathcal{O}_u.x)P(\mathcal{O}_u.y)P(\mathcal{O}_u.\theta), \quad \text{where} \tag{2.4}$$

$$P(\mathcal{O}_u.x) = \frac{e^{-(\mathcal{O}_u.x-\mu_x)^2/(2\sigma_x^2)}}{\sqrt{2\pi}\sigma_x}$$

$$P(\mathcal{O}_u.y) = \frac{e^{-(\mathcal{O}_u.y-\mu_y)^2/(2\sigma_y^2)}}{\sqrt{2\pi}\sigma_y}$$

$$P(\mathcal{O}_u.\theta) = \frac{e^{k\cos(\mathcal{O}_u.\theta-\mu_\theta)}}{2\pi I_0(k)}.$$

As discussed previously, the expected values are actually $\mu_{x|M_i[u-1],\tau_u}$, $\mu_{y|M_i[u-1],\tau_u}$, and $\mu_{\theta|M_i[u-1],\tau_u}$ but simplified to μ_x, μ_y, and μ_θ respectively for clarity. The probability that the observation made ($\mathcal{O}_u$) can be determined for each OI. These values are then used to calculate the belief state of the intent variable using Equation 2.2 (CALC-BELIEF-STATE). An example OI and simulation run will now be provided.

2.6.3 Example OI

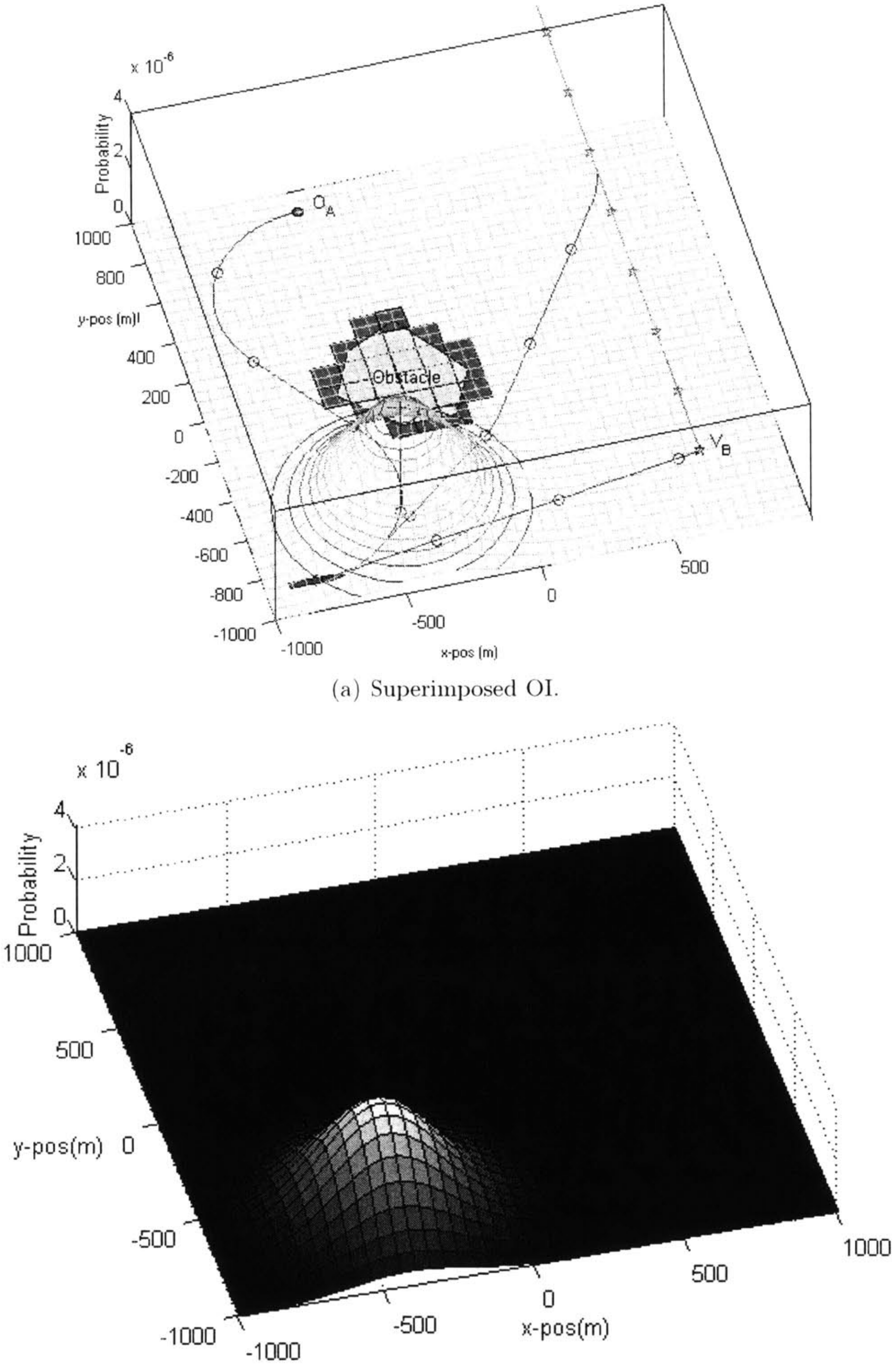
A visual illustration of a marginalized OI for a given intent is shown in Figure 2-6. Since OI contains the probabilistic value for the combinations of three random variables (x, y, and θ), presenting the distribution in terms of all three variables can be difficult. Therefore, the displayed OI is marginalized against the heading parameter to make it easier to present. A marginalized probability distribution allows OI to be calculated in terms of only some of its variables by integrating out the other variables. For example, the distribution presented in Figure 2-6 was calculated by integrating out the θ from the observation variable ($\mathcal{O}_u$)

$$P_{\tau_u}((x,y)_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}) = \int_\theta P_{\tau_u}((x,y,\theta)_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}).$$

This particular OI (Figure 2-6) is for the *Approach*(A) intent (represented hereafter as $\mathbf{i}_1$) across all xy observations using a model generated for the first update ($u = 1$) with an update time (τ_1) of 60 seconds from the initial observation ($\mathcal{O}_o$). Figure 2-6(a) represents the same configuration from Figure 2-3(b) with the marginalized OI superimposed on it. Each circle along each path (intent model) denotes where the contact is projected to be at 60 second time intervals. Even though the figures only display a range of [-1000, 1000] along both axes, the OI extends out to $\pm\infty$. As expected, the maximum probability is located directly over the 60 second mark along the anticipated path. Figure 2-6(b) presents a better view of the marginalized OI where black represents a near-0 probability. To obtain the complete OI (including the heading), the marginalized OI is multiplied by the probability value obtained from Equation 2.3.

2.6.4 Example Simulation

To help illustrate and integrate the modeling elements discussed in this section, an example simulation will now be presented. This simulation will use the scenario provided by Figure 2-3. The task is to determine what the contact is trying to



(a) Superimposed OI.

(b) OI only.

Figure 2-6: Example Marginalized OI for the *Approach*(A) intent as previously shown in Figure 2-3(b). The maximum probability value is at $(-418, -604)$

	$\mathfrak{i}_1$	$\mathfrak{i}_2$	$\mathfrak{i}_3$
$\mu_x\ (m)$	-417.8744	-327.5816	-389.0770
$\mu_y\ (m)$	-604.2986	-775.9550	-633.4657
$\mu_\theta\ (den)$	24.5191	80.3852	50

Table 2.2: Ideal pose for $\mathfrak{i}_i$ at $\tau_1 = 60$.

accomplish (what its intent is). Using the intent definition from Section 2.3 and the available targets in the environment, the list of possible intents that the contact could have are as follows:

$$\mathfrak{I} = \{Approach(A), Follow(B), Intercept(B)\}.$$

This is the state space for the contact's intent and the states will be referred to as $\mathfrak{i}_1$, $\mathfrak{i}_2$, and $\mathfrak{i}_3$ respectively. Since there is no prior knowledge concerning the contact's intent, the prior probabilities are equal ($\pi_i = 1/3, \forall_i$). This provides the initial intent belief state as

$$\begin{aligned} P(\mathcal{I}_0 = \mathfrak{i}_1) &= 1/3, \\ P(\mathcal{I}_0 = \mathfrak{i}_2) &= 1/3, \\ P(\mathcal{I}_0 = \mathfrak{i}_3) &= 1/3. \end{aligned} \tag{2.5}$$

2.6.4.1 Initial Update

The intent models ($M_i[0]$) are determined using the initial observation ($\mathcal{O}_0$) of the environment and are displayed in Figure 2-7 (CREATE-INTENT-MODEL). The process then waits until the next observation ($\mathcal{O}_1$) is supplied to the intent estimator by the UUV. For this simulation, the contact is next observed at the position shown in Figure 2-8, 60 seconds later ($\tau_1 = 60$). The observed pose of the contact is $x = -334.9467m$, $y = -747.3993m$ and $\theta = 65.1173°$. For the first update ($u = 1$), using the intent models ($M_i[0]$) and elapsed time (τ_u), the ideal poses are determined (EXTRACT-IDEAL-XYθ) and provided in Table 2.2.

Using Equation 2.4 (OBSERVED-INTENT-PDF), the calculated probability values

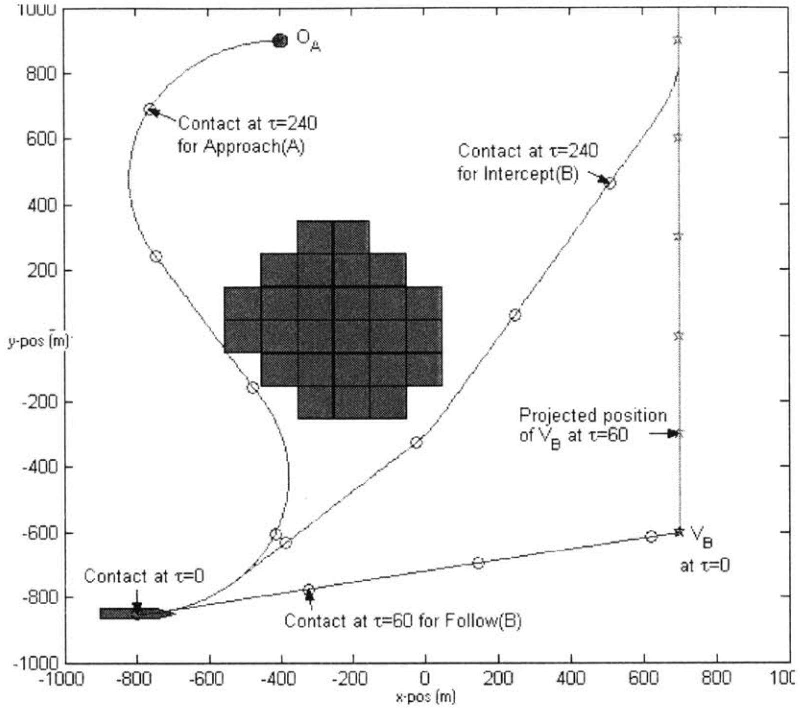


Figure 2-7: Example intent models.

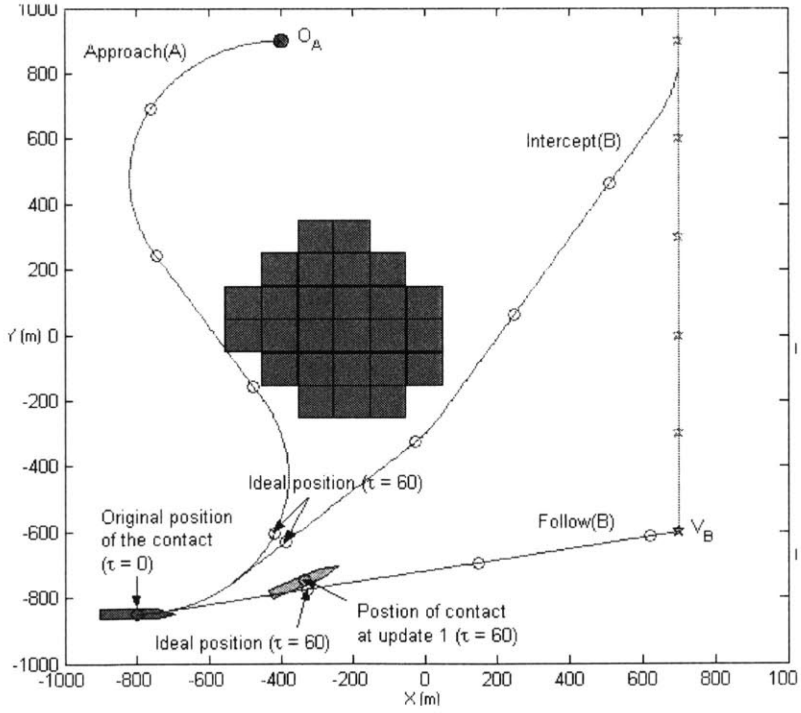


Figure 2-8: Update 1 (at $\tau = 60$).

from each OI for the first update are as follows

$$\begin{aligned}
P_{\tau_1}(\mathcal{O}_1|\mathcal{I}_1 = \mathbf{i}_1, \mathcal{O}_0) &= 7.3563 \times 10^{-7},\\
P_{\tau_1}(\mathcal{O}_1|\mathcal{I}_1 = \mathbf{i}_2, \mathcal{O}_0) &= 2.8608 \times 10^{-6},\\
P_{\tau_1}(\mathcal{O}_1|\mathcal{I}_1 = \mathbf{i}_3, \mathcal{O}_0) &= 2.3786 \times 10^{-6}.
\end{aligned}$$

Using Equations 2.5 and the above $P_{\tau_1}(\mathcal{O}_1|\mathcal{I}_1 = \mathbf{i}_i, \mathcal{O}_0)$ values in Equation 2.2 (CALC-BELIEF-STATE), the updated intent estimates are calculated as

$$\begin{aligned}
P_{\tau_1}(\mathcal{I}_1 = \mathbf{i}_1|\mathcal{O}_1, \mathcal{O}_0) &= 0.1231,\\
P_{\tau_1}(\mathcal{I}_1 = \mathbf{i}_2|\mathcal{O}_1, \mathcal{O}_0) &= 0.4788,\\
P_{\tau_1}(\mathcal{I}_1 = \mathbf{i}_3|\mathcal{O}_1, \mathcal{O}_0) &= 0.3981.
\end{aligned}$$

The highest probability belongs to $\mathbf{i}_2$ (*Follow*(B)), so the intent estimator considers that the contact's current intent.

2.6.4.2 Subsequent Updates

After the $u = 1$ update, the ideal path for each intent is recalculated from the contact's currently observed pose using the path planner, resulting in the paths shown in Figure 2-9. Each path is an intent model $M_i[u-1]$ for update $u = 2$. As for the first update, the process waits for the UUV to make the next observation $(\mathcal{O}_u)$. In Figure 2-9, the contact's position at the next observation is indicated. This observation is made 120 seconds from the previous update $(\tau_u = 120)$. This illustrates that the time between updates is not required to be constant. Using the intent models $(M_i[u-1])$, created using the last observation $(\mathcal{O}_{u-1})$, and τ_u, the ideal contact poses are shown in Table 2.3. The OI $P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})$ is calculated using Equation 2.4. The intent belief state is then updated (using Equation 2.2) to obtain:

$$\begin{aligned}
P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_1|\mathcal{O}_u, \mathcal{O}_{u-1}) &= 0.1223,\\
P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_2|\mathcal{O}_u, \mathcal{O}_{u-1}) &= 0.0127,\\
P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_3|\mathcal{O}_u, \mathcal{O}_{u-1}) &= 0.8650.
\end{aligned}$$

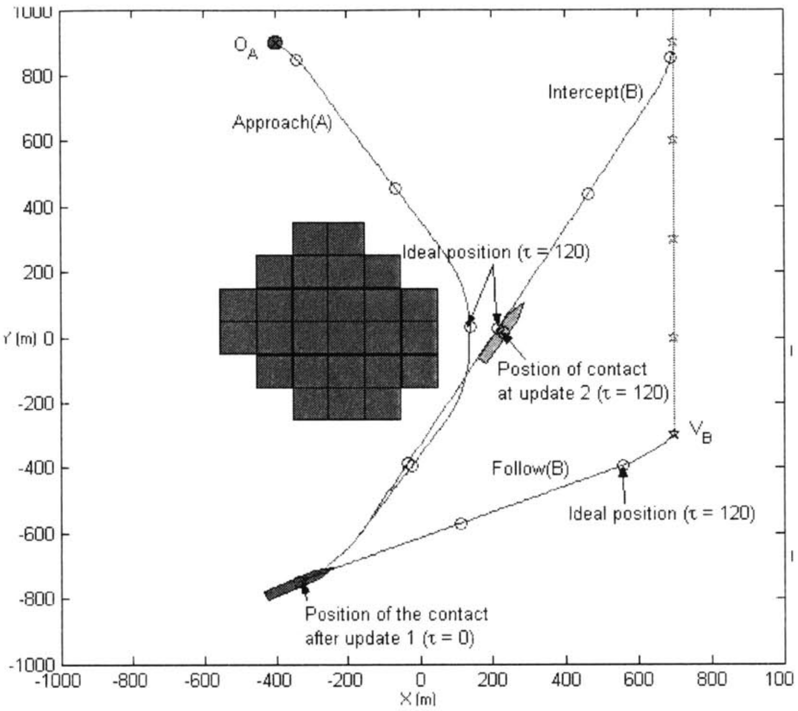


Figure 2-9: Update 2 (after $\tau = 120$).

	$\dot{\boldsymbol{\iota}}_1$	$\dot{\boldsymbol{\iota}}_2$	$\dot{\boldsymbol{\iota}}_3$
$\mu_x\ (m)$	138.1059	557.5062	214.8984
$\mu_y\ (m)$	33.5123	-393.5693	26.0778
$\mu_\theta\ (den)$	359.2074	68.1267	31.0914

Table 2.3: Ideal poses for $\dot{\boldsymbol{\iota}}_i$ at $\tau_2 = 120$.

The intent estimator has now determined (with an 86.50% confidence level) that the contact has intent $\mathbf{i}_3$ (*Intercept*(B)).

Once the intent belief state is updated, u is incremented and lines 12–25 from INTENT-ESTIMATION (Figure 2-2) are repeated.

2.7 Summary

This chapter introduced and developed a method for maintaining a belief state over the possible intents of a maritime contact by observing its actions. This method is referred to as the intent estimation model. At the very fundamental level, the intent estimator uses Bayes' rule to indirectly determine the belief state of the intent variable using the previous and current observations made of the contact along with the intent models provided.

[This page intentionally left blank]

Chapter 3

Advanced Intent Estimation Model

The basic intent estimation model presented in Chapter 2 has been relatively simple and straightforward. Multiple assumptions were made concerning the observed contact's capabilities, its maneuvering behaviors, and the targets in the environment. As a result, the capabilities of the intent estimator is limited to scenarios where those assumptions are accurate. This chapter will present several methods to improve the basic intent estimation model to make it more robust. A structure called the *Bayesian network* will first be used to represent the basic intent estimation model. The network will then be expanded to improve the intent estimator. Finally, a forward-backward procedure for improving the belief state calculations is presented.

3.1 Bayesian Network

A Bayesian network is a structure that can be used to incorporate the variables in a probabilistic model and their interdependencies in a clear manner [18, 8]. This section will provide a basic overview of how a Bayesian network is structured, create a Bayesian network for the intent estimation model presented earlier, illustrate how the belief state for $\mathcal{I}_u$ can be determined using the network, and then compare the intent belief state calculations using the network to the method provided in Section 2.4 to show that they are the same.

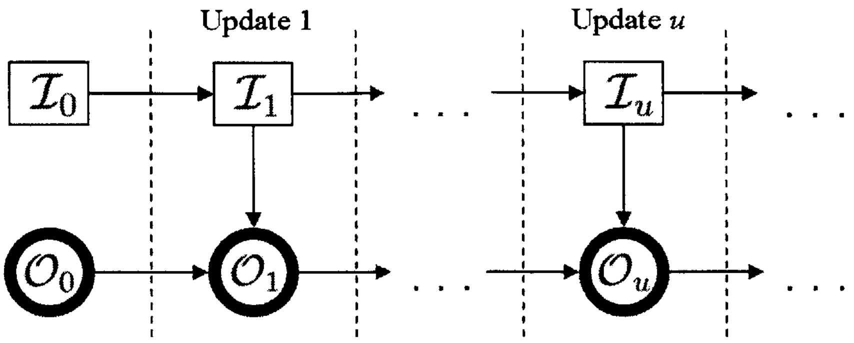


Figure 3-1: Initial Bayesian network for the intent estimator.

A Bayesian network consists of a set of nodes, each representing a variable in the model. Each variable can be either discrete or continuous. Any *direct* dependencies between the variables in the model are represented by an arrow connecting the variables together. An arrow from node A (the *source* or *parent*) to node B (the *child*) indicates that B is directly affected by the value of A. The effects of the parent nodes on a child node, X, can be represented by a Conditional Probability Distribution (CPD) $P(X|Parents(X))$. If a node has no parents, then $P(X|Parents(X))$ is the same as $P(X)$, which is the belief state for that variable. Since each node in a Bayesian network is "conditionally independent of its predecessors in the node ordering, given its parents", the full joint probability distribution between all the nodes in the model can be determined using just the CPDs by "computing sums of products of conditional probabilities from the network" [18].

The Bayesian network representation of the intent estimation model presented so far is shown in Figure 3-1. Square nodes are used to represent discrete variables and circular nodes represent continuous variables. Nodes with bold borders are observable (variables for which values are directly observable) and all others nodes are hidden. The values for $\mathcal{I}_u$ and $\mathcal{O}_u$ before any updates are made, at $u = 0$, are the initial belief state and observation of the contact respectively used to start the intent estimation process (Section 2.4). Using the example from Section 2.6.4, the initial belief state

$P(\mathcal{I}_0 = \mathbf{i}_1)$	$P(\mathcal{I}_0 = \mathbf{i}_2)$	$P(\mathcal{I}_0 = \mathbf{i}_3)$
1/3	1/3	1/3

Table 3.1: Example initial probability distribution for $\mathcal{I}_0$.

	$P(\mathbf{i}_1 \vert \mathcal{I}_{u-1})$	$P(\mathbf{i}_2 \vert \mathcal{I}_{u-1})$	$P(\mathbf{i}_3 \vert \mathcal{I}_{u-1})$
$\mathcal{I}_{u-1} = \mathbf{i}_1$	1	0	0
$\mathcal{I}_{u-1} = \mathbf{i}_2$	0	1	0
$\mathcal{I}_{u-1} = \mathbf{i}_3$	0	0	1

Table 3.2: $P(\mathcal{I}_u|\mathcal{I}_{u-1})$, where $u > 1$.

for $\mathcal{I}_0$ is shown in Table 3.1. For each update, where $u \geq 1$, the CPD for the $\mathcal{O}_u$ node is $P_{T_u}(\mathcal{O}_u|\mathcal{I}_u, \mathcal{O}_{u-1})$ and the CPD for the $\mathcal{I}_u$ node is $P(\mathcal{I}_u|\mathcal{I}_{u-1})$.

Using this model and the semantics of the Bayesian network, the probability of the contact having intent $\mathbf{i}_i$ given the current and previous observations of it is

$$P_{T_u}(\mathcal{I}_u = \mathbf{i}_i|\mathcal{O}_u, \mathcal{O}_{u-1}) = \eta \sum_{\mathcal{I}_{u-1}} P_{T_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})P(\mathcal{I}_u = \mathbf{i}_i|\mathcal{I}_{u-1})P(\mathcal{I}_{u-1})P(\mathcal{O}_{u-1}), \tag{3.1}$$

where η is the normalizing factor. Continuing the assumption that the contact's intent does not change,

$$P(\mathcal{I}_u = \mathbf{i}_i|\mathcal{I}_{u-1} = \mathbf{i}_j) = 0, \quad \forall\, i \neq j,$$

the CPD for node $\mathcal{I}_u$ is shown in Table 3.2.

Equation 3.1 represents the intent estimation method presented in Section 2.4 and produces the same result as Equation 2.2. The latter equation is a condensed version of the former. They can be shown to be the same by the following steps. Rearranging Equation 3.1 produces

$$P_{T_u}(\mathcal{I}_u = \mathbf{i}_i|\mathcal{O}_u, \mathcal{O}_{u-1}) = \eta P_{T_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})P(\mathcal{O}_{u-1}) \sum_{\mathcal{I}_{u-1}} P(\mathcal{I}_u = \mathbf{i}_i|\mathcal{I}_{u-1})P(\mathcal{I}_{u-1}). \tag{3.2}$$

Using the CPD for $\mathcal{I}_u$ (Table 3.2), the summation in Equation 3.2 can be represented as

$$\sum_{j=1}^{N} P(\mathcal{I}_u = \mathbf{i}_i|\mathcal{I}_{u-1} = \mathbf{i}_j)P(\mathcal{I}_{u-1} = \mathbf{i}_j) = P(\mathcal{I}_{u-1} = \mathbf{i}_j), \text{ where } i = j,$$

allowing Equation 3.2 to be rewritten as

$$P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_i|\mathcal{O}_u, \mathcal{O}_{u-1}) = \eta P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})P(\mathcal{O}_{u-1})P(\mathcal{I}_{u-1} = \mathbf{i}_i).$$

Since the value for $\mathcal{O}_{u-1}$ is observable,

$$P(\mathcal{O}_{u-1}) = 1,$$

which produces

$$P_{\tau_u}(\mathcal{I}_u = \mathbf{i}_i|\mathcal{O}_u, \mathcal{O}_{u-1}) = \eta P(\mathcal{I}_{u-1} = \mathbf{i}_i)P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}),$$

which is Equation 2.2, where the denominator (normalizing factor) is represented by η.

As mentioned earlier in this section, the intent estimation model can be made more robust and representative of reality by incorporating additional variables. As more variables are incorporated, the model becomes more complex. The Bayesian network model provides the ability to clearly illustrate the dependencies among the variables in the model and to provide a method for calculating the intent belief state ($P(\mathcal{I})$) without having to generate a full joint probability distribution. This model will now be expanded with additional variables to make the intent estimator more realistic.

3.2 Intent Estimator Improvements

In Chapter 2, the contact's intent had been modeled as not changing and using a fixed speed and turn radius. This section will discuss how the intent models can be improved by allowing a contact to move with various speeds and turn radii, and to change intents. The first improvement allows modeling different vessel types. The second allows the contact to respond to vessels entering or leaving the observation environment.

3.2.1 Multiple Contact Path Models

A contact of a certain vessel type (e.g., tanker, speedboat) has the ability to travel using a wide range of speeds and turn radii as dictated by the design of the vessel. The current intent model only considers the use of one combination of speed and turn radius to create an unique path for each intent. This model can be improved by accounting for the different possible paths a contact could take (using different speeds and turn radii) to accomplish a certain intent.

To incorporate a variety of speed/turn radius combinations into the intent model for a specific vessel type, certain parameters of the vessel need to be known. These parameters include the speed range over which the vessel can operate and its minimum turn radius. The unclassified speed range for a given vessel type is usually published and easily accessible. The minimum maneuvering speed (the minimum speed at which the vessel can move and still maintain rudder effectiveness) and the maximum speed of the vessel will be represented by s_S (slow) and s_F (fast) respectively. Obtaining the turning radius of the vessel is not as straightforward. The turn path of a large vessel does not follow a perfect circle with a specific radius. The shape of an actual turn is provided by an *advance and transfer* table which is specific to a type of vessel and depends on its speed, extent of rudder deflection ("rudder used"), and the desired amount of change in heading [7]. To simplify the model, it is assumed that the deviation between the actual turn of a vessel and a circular model of that turn is negligible for the purposes of this thesis. Therefore, the turn capability of a vessel (W) will be represented by a turn radius. The minimum turn radius, W_M (the turn radius produced when full rudder is used), is assumed to be two times the vessel's length (W_L) [5]. This is written as

$$W_M = 2W_L.$$

The domain for both the speed and turn radius variables are continuous, creating an infinite number of possible speed/turn radius combinations that can be considered. To reduce the combinatorial complexity of the problem, only three values for each variable will be used to produce a representative set of possible paths for each intent. The domain for the speed variable ($\mathcal{S}$) is

$$\mathcal{S} \in \mathfrak{S} = \{s_S, s_M, s_F\},$$

where s_S and s_F are the minimum and maximum speeds and s_M is the estimated normal maneuvering speed of the vessel (somewhere in the middle). The contact's turn radius (r) will depend on the amount of rudder used ($\mathcal{U}$) and the relationship between the two. The domain for $\mathcal{U}$ will be described as

$$\mathcal{U} \in \mathfrak{U} = \{100\%, 66\%, 33\%\},$$

where a value of 100% would represent full rudder deflection, and 0% would indicate no rudder usage. Since the value for $\mathcal{U}$ represent the amount of rudder used to make a turn, a value of 0 would signify that the contact does not turn (only moves in a straight line) and therefore is not included in $\mathfrak{U}$. The equation created to represent the relationship between the turn radius and the amount of rudder used is:

$$r_{\mathcal{U}} = \frac{100}{\mathcal{U}} W_M, \; \mathcal{U} > 0. \tag{3.3}$$

This equation was developed to provide a simplified model for estimating the relationship between the extent of the rudder used and the resulting turn radius. The modeled relationship illustrates that a full rudder deflection would produce the vessel's minimum turn radius, a half deflection would double the turning radius, etc.

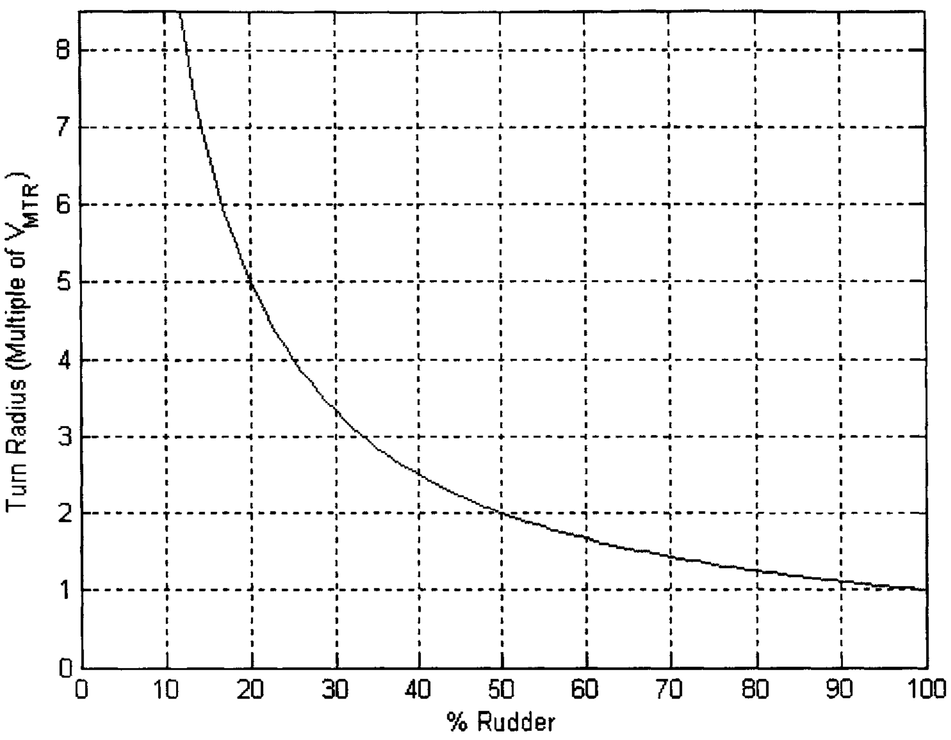


Figure 3-2: Relationship between the turn radius and % of rudder used.

The graph of this relationship is displayed in Figure 3-2. Using Equation 3.3 and the values for $\mathcal{U}$, the domain for the turn radius variable ($\mathcal{R}$) is

$$\mathcal{R} \in \mathfrak{R} = \{r_S, r_M, r_L\},$$

where r_S, r_M, and r_L represents a short (using 100% rudder), medium (using 66% rudder), and long (using 33% rudder) turning radius respectively. The different speed/turn radius combinations that will be incorporated into each intent model is the cross product: $\mathcal{SR} \in \mathfrak{S} \times \mathfrak{R}$.

Using the path planner to generate a path for every speed/turn radius combinations produces nine unique paths. Each path specifies the (x, y) positions that a contact will move through and the time at which the contact will be at those positions. The time dimension associated with each path produces another level of distinction among the paths that is just as important, but not as visible, as the x and y values. For example, two paths created to approach a target using the same value for $\mathcal{R}$ but different values for $\mathcal{S}$ will have the same track over the ground but would get to the target at different times. Therefore, the two paths might look exactly the same but are actually different. The example intent models illustrated in Figure 2-7 shows the modeled paths of the contact for the different intents with the times that the contact is expected to be at certain positions along those paths.

The domain of the paths generated by the intent estimator using each intent ($\mathcal{I}$), speed ($\mathcal{S}$), and turn radius ($\mathcal{R}$) combination is

$$\mathcal{P} \in \mathfrak{P} = \mathfrak{I} \times \mathfrak{S} \times \mathfrak{R}. \tag{3.4}$$

Each path (p) will be uniquely identified with an "$\mathcal{I}, \mathcal{SR}$" subscript indicating the *"intent,speed/turn"* combination with which it was created. For example, a path generated for $\mathcal{I} = \mathfrak{i}_i$ and $\mathcal{SR} = s_M r_M$ will be represented as $p_{i,s_M r_M}$. The subscript i is used to represent the specific intent instead of $\mathfrak{i}_i$ for clarity. To combine these paths to create an intent model for each intent, a probability distribution over $\mathfrak{P}$ is needed to indicate the likelihood that the contact would implement a particular path given its intent. This distribution can be represented as

$$P(\mathcal{P}|\mathcal{I}_u = \mathfrak{i}_i, \mathcal{O}_{u-1}). \tag{3.5}$$

Equation 3.5 can be used to calculate the OI for $\mathfrak{i}_i$ at update u as follows

$$P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathfrak{i}_i, \mathcal{O}_{u-1}) = \sum_{\mathcal{P}} P_{\tau_u}(\mathcal{O}_u|\mathcal{P}) P(\mathcal{P}|\mathcal{I}_u = \mathfrak{i}_i, \mathcal{O}_{u-1}), \tag{3.6}$$

where $P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathfrak{i}_i, \mathcal{O}_{u-1})$ represents the $\mathfrak{i}_i$ OI from Section 2.6, $P_{\tau_u}(\mathcal{O}_u|\mathcal{P})$ is the probability that the observation value ($\mathcal{O}_u$) will be made of the contact given that it is has travelled τ_u seconds along path indicated by $\mathcal{P}$, and $P(\mathcal{P}|\mathfrak{i}_i, \mathcal{O}_{u-1})$ is the probability that the contact would choose to follow the path indicated by $\mathcal{P}$, given its intent and the observation made at the previous update. The previous observation, $\mathcal{O}_{u-1}$, is used to generate the paths.

For a given intent, $\mathbf{i}_i$, only the nine paths generated for that intent are used to create the intent model. These paths are identified as $p_{i,\mathcal{SR}}$. To preclude the other paths from being incorporated into the intent model for $\mathbf{i}_i$, the probability distribution value for paths which were not created for $\mathbf{i}_i$ is 0,

$$P(p_{j,\mathcal{SR}}|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}) = 0, \quad \forall\, i \neq j. \tag{3.7}$$

To combine the remaining paths together to create one model, a probability distribution over the nine paths (each produced using a different $\mathcal{SR}$) is needed to determine how likely a contact will implement one path over another. It is reasonable to assume that the speed/turn radius combination used by a contact to perform a certain intent will depend on the geometric relationship between the contact and the intent target (this will be referred to as the *contact-target* relationship). The probability distribution over the nine paths for a given intent ($\mathbf{i}_i$) can be represented as

$$P(\mathcal{P}_{i,\mathcal{SR}}|\mathbf{i}_i, \text{contact-target relationship}). \tag{3.8}$$

This contact-target relationship is the distance and *relative bearing* of the target from the contact. The relative bearing will also be referred to as just the *bearing* and will have a range of [-180°, 180°]. To simplify the correlation between the contact-target relationship and the different speed/turn radius combinations, the domain for the contact-target relationship will be discretized into nine zones centered around the contact. The domain for the distance variable ($\mathcal{D}$) between the target and the contact is:

$$\mathcal{D} \in \mathfrak{D} = \{d_C, d_I, d_F\},$$

where d_C, d_I, and d_F indicates that the target is *close*, *intermediate*, or *far* from the contact respectively. The distance zones are arbitrarily defined as follows:

Close	$0 \leq \quad d_C \quad \leq 3W_M,$
Intermediate	$3W_M < \quad d_I \quad \leq 9W_M,$
Far	$d_F \quad > 9W_M.$

By making the distance zone classifications dependent on the minimum turning radius of the vessel (W_M), the zones will scale in proportion to the size of the vessel. This assumes that the classification (close, intermediate, or far) of the distance to a target is directly affected by the size of the vessel. For this thesis, a target is considered to be close to the contact if it is within three W_M of the contact. If the target is within nine W_M but further than three W_M, then it is in an intermediate zone. If the distance from the contact to the target is greater than nine W_M, then the target is considered to be far from the contact. The domain of the bearing variable ($\mathcal{B}$) is also divided into three groups:

$$\mathcal{B} \in \mathfrak{B} = \{b_F, b_S, b_A\},$$

where b_F, b_S, and b_A represents a target that is *forward*, to the *side* (abeam), or *aft* of (behind) the contact respectively. The bearing zones are arbitrarily divided as follows:

Forward	$0 \leq \quad \lvert b_F \rvert \quad \leq 45°,$
Side	$45° < \quad \lvert b_S \rvert \quad \leq 135°,$
Aft	$135° < \quad \lvert b_A \rvert \quad \leq 180°.$

The domain for the target zone variable ($\mathcal{DB}$) is

$$\mathcal{DB} \in \mathfrak{DB} = \mathfrak{D} \times \mathfrak{B}$$

and is illustrated in Figure 3-3.

Since the value for $\mathcal{DB}$ is determined by the previously observed pose of the contact ($\mathcal{O}_{u-1}$) and the position of the target (provided by $\mathfrak{i}_i$), the zone that the target is

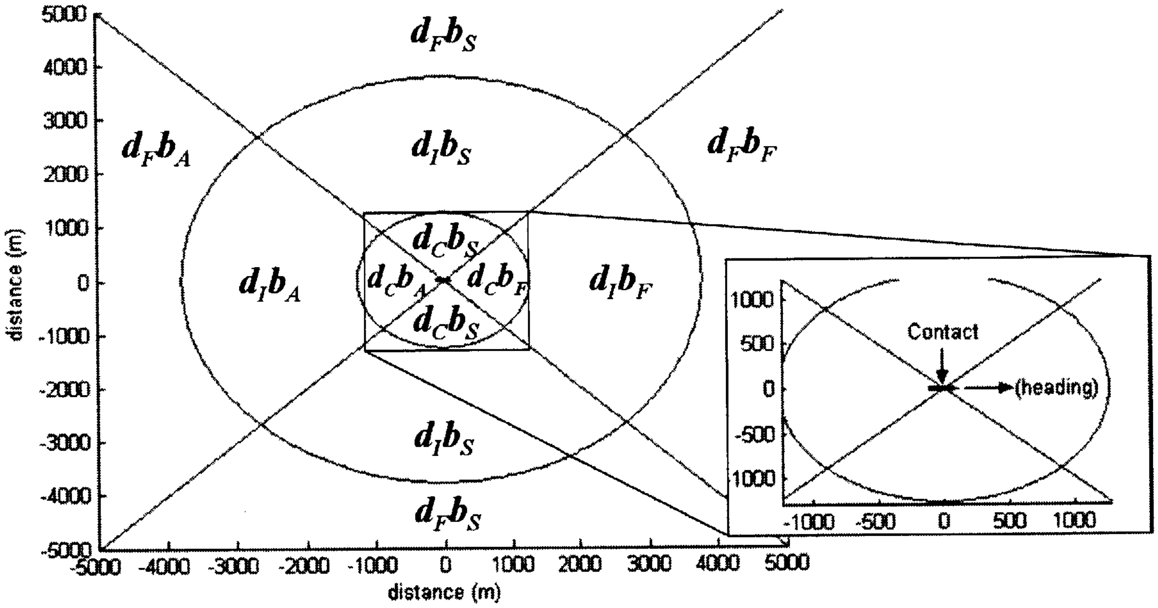


Figure 3-3: Distance and bearing zones ($\mathcal{DB}$) for a contact whose $W_M = 420m$.

in can be determined by the information currently available. Equation 3.8 can be rewritten with these two terms as follows

$$P(\mathcal{P}_{i,\mathcal{SR}}|\mathbf{i}_i, \mathcal{O}_{u-1}).$$

The probability distribution over the nine paths, which were created using different $\mathcal{SR}$ combinations, are arbitrarily defined using the intent *behavior* and zone where the intent target appears as a guide. The probability distribution over the nine paths is therefore different for each intent behavior and zone that the target is in. The values were chosen to produce what seem to be reasonable behaviors. An example distribution for the *approach* behavior is provided in Table 3.3 where each column is a probability distribution for a set of paths that the contact would consider implementing given the contact-target relationship. For example, given that the behavior of the contact's intent is to *approach* a target which is close to and aft (behind) the contact,

$$\mathcal{DB} = d_C b_A,$$

	$d_C b_F$	$d_C b_S$	$d_C b_A$	$d_I b_F$	$d_I b_S$	$d_I b_A$	$d_F b_F$	$d_F b_S$	$d_F b_A$
$P(\mathcal{P}_{i,s_F r_S} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.01	0.19	0.19	0.01	0.18	0.19
$P(\mathcal{P}_{i,s_M r_S} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.01	0.19	0.19	0.01	0.33	0.33	0.01	0.33	0.33
$P(\mathcal{P}_{i,s_S r_S} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.01	0.33	0.33	0.01	0.01	0.01	0.01	0.01	0.01
$P(\mathcal{P}_{i,s_F r_M} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.25	0.18	0.18	0.19	0.25	0.25
$P(\mathcal{P}_{i,s_M r_M} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.18	0.18	0.18	0.33	0.25	0.25	0.18	0.19	0.18
$P(\mathcal{P}_{i,s_S r_M} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.19	0.25	0.25	0.01	0.01	0.01	0.01	0.01	0.01
$P(\mathcal{P}_{i,s_F r_L} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.19	0.01	0.01	0.25	0.01	0.01
$P(\mathcal{P}_{i,s_M r_L} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.25	0.01	0.01	0.18	0.01	0.01	0.33	0.01	0.01
$P(\mathcal{P}_{i,s_S r_L} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$	0.33	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Table 3.3: Probability distribution table for $P(\mathcal{P}_{i,\mathcal{SR}} \mid \mathbf{i}_i, \mathcal{O}_{u-1})$ where $\mathbf{i}_i$ reflects an *approach* behavior.

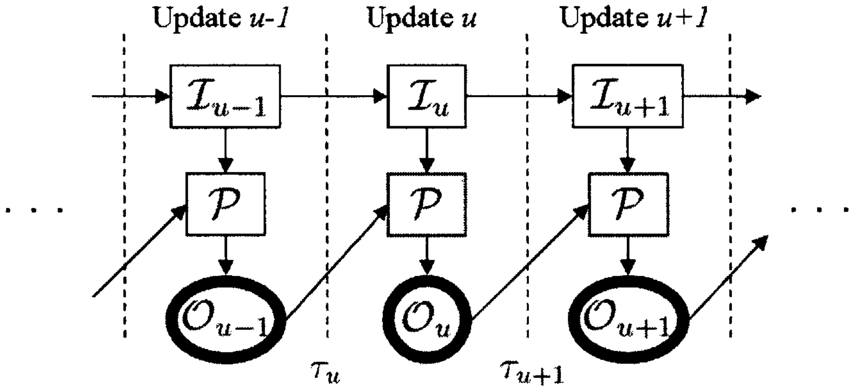


Figure 3-4: Updated Bayesian network to include multiple path models.

the values in the table indicated that there is a 0.01 probability that the contact will use its maximum speed and shortest turning radius to reach the target, a 0.19 probability that the contact will use its medium speed and shortest turning radius to reach the target, etc. Similar tables for the intersect and follow behaviors are provided in Appendix B.

The complete CPD for $\mathcal{P}$ ($P(\mathcal{P} \mid \mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1})$ from Equation 3.5) can be created using Equation 3.7, Table 3.3, and the tables in Appendix B. The CPD is shown in Table 3.4. The graphical view of the Bayesian network of Equation 3.6 is shown in Figure 3-4.

The original intent estimation model is a degenerate case of the model just provided—

$\mathcal{I}_u$	$\mathcal{DB}$	$\mathcal{P}_{1,s_F r_S}$	$\dots$	$\mathcal{P}_{1,s_S r_L}$	$\mathcal{P}_{2,s_F r_S}$	$\dots$	$\mathcal{P}_{N,s_S r_L}$
$\mathbf{i}_1$	$\mathcal{O}_{u-1}$	$P(\mathcal{P}_{1,s_F r_S} \mid \mathbf{i}_1, \mathcal{O}_{u-1})$	$\dots$	$P(\mathcal{P}_{1,s_S r_L} \mid \mathbf{i}_1, \mathcal{O}_{u-1})$	0	$\dots$	0
$\mathbf{i}_2$	$\mathcal{O}_{u-1}$	0	$\dots$	0	$P(\mathcal{P}_{2,s_F r_S} \mid \mathbf{i}_2, \mathcal{O}_{u-1})$	$\dots$	0
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\mathbf{i}_N$	$\mathcal{O}_{u-1}$	0	$\dots$	0	0	$\dots$	$P(\mathcal{P}_{N,s_S r_L} \mid \mathbf{i}_N, \mathcal{O}_{u-1})$

Table 3.4: CPD for $\mathcal{P}$, $P(\mathcal{P} \mid \mathcal{I}_u, \mathcal{O}_{u-1})$.

$\mathcal{I}$	$\mathcal{DB}$	$p_{1,\mathcal{SR}}$	$p_{2,\mathcal{SR}}$	$\cdots$	$p_{N,\mathcal{SR}}$
$\mathbf{i}_1$	$\mathcal{O}_{u-1}$	1	0	$\ldots$	0
$\mathbf{i}_2$	$\mathcal{O}_{u-1}$	0	1	$\ldots$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\mathbf{i}_N$	$\mathcal{O}_{u-1}$	0	0	$\ldots$	1

Table 3.5: $P(p|\mathbf{i}_i, \mathcal{O}_{u-1})$ for the degenerate case.

where one path is used to determine an intent model instead of nine. The CPD for the original intent model using the current Bayesian network is provided in Table 3.5, and Equation 3.6 simplifies for that case to

$$P_{\tau_u}(\mathcal{O}_u|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}) = P_{\tau_u}(\mathcal{O}_u|\mathcal{P}_{i,\mathcal{SR}}).$$

In certain situations, a path cannot be generated using the $\mathbf{i}_i$, $\mathcal{O}_{u-1}$, $\mathcal{S}$, and $\mathcal{R}$ parameters provided. For example, if a contact is navigating in a restricted environment (i.e., an environment in which the contact would not be able to make wide turns without running into an obstacle), using r_L to generate a path to accomplish its intent may be impossible. However, using r_S would generate a usable path. Another example would be if the contact was trying to intercept another vessel. In most situations, such an intent cannot be accomplished (i.e., a path cannot be generated) if the contact moves at its minimum maneuvering speed, s_S. However, the intent can be accomplished if the maximum speed (s_F) was used.

In both of those scenarios, some of the paths cannot be generated (and will be referred to as being unavailable) while others can be created. The method used to handle this situation is to remove these path options from being considered in any calculations by replacing each column in the $\mathcal{P}$ CPD table with zeros for an unavailable path with zeros. Since each path was created for a specific intent, as illustrated by Table 3.4, only one row is affected by each unavailable path. The affected rows are then normalized to provide a probability distribution over the resulting (available) paths options. If all the path options for a given intent are unavailable, then the intent is considered invalid and will be removed from the intent state space. This scenario is addressed in Section 3.2.4.

3.2.2 Multiple Contact Types

The extension of using multiple paths to create the intent model in Section 3.2.1 only takes into consideration the capabilities of a single vessel type. If the vessel type of the contact is unknown, then the intent estimator may fail if the actual capabilities of the contact are significantly different from those assumed by the model. To improve the robustness of the model, a variable can be incorporated to account for the uncertainty of the capabilities of the contact.

To provide a range of vessel types to represent different contact capabilities, the variable $\mathcal{V}$ is used to represent the vessel type of the contact. The domain for $\mathcal{V}$ is

$$\mathcal{V} \in \mathfrak{V} = \{v_1, v_2, \ldots, v_M\},$$

where each v_i represents one vessel type with a certain set of capabilities, and M is the number of different types of vessels modeled. Since the actual type of the contact is not observed, a set of paths (using different speed/turn radius combinations) is generated for each v_i. As a result, the domain for the generated paths from Equation 3.4 is extended to incorporate vessel type as follows

$$\mathcal{P} \in \mathfrak{P} = \mathfrak{I} \times \mathfrak{S} \times \mathfrak{R} \times \mathfrak{V}.$$

The probability that a contact will undertake a certain path now depends on its intent, the contact/target relationship (to determine the likelihood of each speed/turn radius combination), and the capability of the vessel (to achieve the actual values for each speed/turn radius combination). As before, the intent of the contact at update u is represented by the $\mathcal{I}_u$ node. The contact/target relationship depends on the observation from the previous update ($\mathcal{O}_{u-1}$) and on the properties of the contact, $\mathcal{V}$, to create the distance zones. In addition, the capability (i.e., speed range and minimum turn radius) of the vessel also depends on the value of the $\mathcal{V}$ variable.

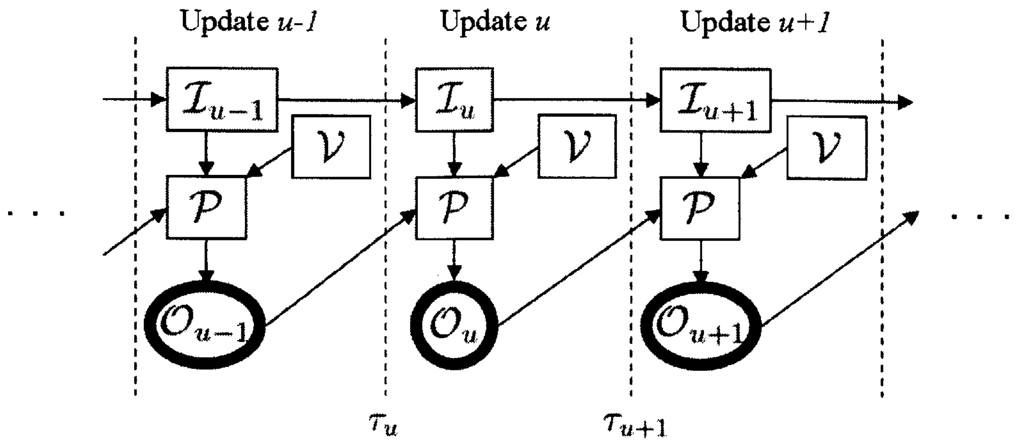


Figure 3-5: Modified Bayesian network to include multiple vessel types.

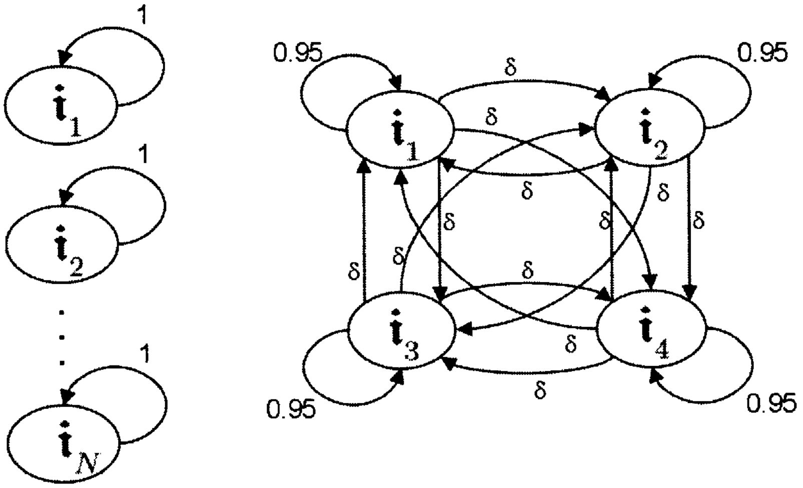
The difference between the model presented in this section and the model from Section 3.2.1 is the addition of the vessel type variable, $\mathcal{V}$. By conditioning on $\mathcal{V}$, the probability that a contact will implement a certain path is calculated as before. The only difference is that the probability value obtained is multiplied by the belief that the contact is of type $\mathcal{V}$ and then summed over $\mathcal{V}$ as in Equation 3.9.

$$P(\mathcal{P}_{i,\mathcal{SR}}|\mathcal{I}_u = \mathbf{i}_i, \mathcal{O}_{u-1}) = \sum_{\mathcal{V}} P(\mathcal{P}_{i,\mathcal{SR}}|\mathcal{I}_u = \mathbf{i}_i, \mathcal{V}, \mathcal{O}_{u-1})P(\mathcal{V}). \tag{3.9}$$

The prior belief values for $P(\mathcal{V})$ (that the contact is of a given type) are provided externally. The resulting Bayesian network is shown in Figure 3-5.

3.2.3 Dynamic Contact Intent

Another improvement that can be made to the intent estimator is to allow the contact to change its intent during the time it is observed. In reality, such a change may occur at any time and would require the intent estimator to consider the possibility of an intent change at each time step. This approach was used by Mierisch [11] where an intent change by a vessel (with constant parameters) may occur at any time step between updates. However, to simplify the model used in this thesis, the intent estimation model is created to only account for an intent change at each update.



(a) Original transition model.

(b) Example using the improved transition model.

Figure 3-6: Transition probabilities.

This is a reasonable model if the contact does not change its target and behavior (intent) frequently and the time between updates is reasonably small. Therefore, if the contact changes its intent between two updates, the discrepancy between its observed pose and the intent model's expectation would not be that large (since insufficient time would have elapsed). In addition, the effects from that update would eventually diminish as subsequent updates are made of the new intent (if it stays constant).

The probability that the contact will change its intent between updates is represented by the transition probabilities between the states of the intent variable, $\mathcal{I}$. For the original model (Table 3.2), these transition probabilities were 1 for staying in a state, and 0 for changing intents. This is illustrated in Figure 3-6(a). To model the possibility of a change in the contact's intent, the transition probabilities between the states are assigned to reflect the assumption that there is a 0.95 probability that the contact will remain in its current state with the probability of changing to another state evenly distributed among the remaining intents. The transition probability for

	$P(\mathbf{i}_1\|\mathcal{I}_{u-1})$	$P(\mathbf{i}_2\|\mathcal{I}_{u-1})$	$\ldots$	$P(\mathbf{i}_N\|\mathcal{I}_{u-1})$
$\mathcal{I}_{u-1} = \mathbf{i}_1$	0.95	δ	$\ldots$	δ
$\mathcal{I}_{u-1} = \mathbf{i}_2$	δ	0.95	$\ldots$	δ
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\mathcal{I}_{u-1} = \mathbf{i}_N$	δ	δ	$\ldots$	0.95

Table 3.6: Modified $P(\mathcal{I}_u|\mathcal{I}_{u-1})$, where $u > 1$.

changing to another particular intent (δ) can be determined as follows:

$$\delta = \frac{1 - 0.95}{N - 1}. \tag{3.10}$$

The numerator is the total probability for changing intents and the denominator is the number of intents to distribute it over. A transition diagram for an intent with four states is provided in Figure 3-6(b). The general transition probabilities are represent by the CPD for node $\mathcal{I}$ in the Bayesian network and illustrated by Table 3.6.

3.2.4 Dynamic Intent State Space

The intent estimator will now be extended to allow for changes in the contact's intent. Such changes can be influenced by its goals and its environment. The intent estimation model provided in Section 3.2.3 relies on the assumption that the number of intents in the state space remains constant. This assumes that no new targets are considered and that none disappear. It also assumes that a path to perform an intent can always be generated and that the contact might always be interested in pursuing any intent. For example, it assumes that a contact would always be able to intercept another vessel, and it might still be interested in intercepting a vessel even after the contact has moved right next to the vessel. To make the intent estimator more robust by being able to account for these situations, a dynamic intent state space is required. A method for adding and removing intents from the state space is provided.

	$P(\mathbf{i}_1 \mid \mathcal{I}_{u-1})$	$P(\mathbf{i}_2 \mid \mathcal{I}_{u-1})$	$P(\mathbf{i}_3 \mid \mathcal{I}_{u-1})$	$P(\mathbf{i}_4 \mid \mathcal{I}_{u-1})$	$P(\mathbf{i}_5 \mid \mathcal{I}_{u-1})$
$\mathcal{I}_{u-1} = \mathbf{i}_1$	0.95	δ	δ	δ	δ
$\mathcal{I}_{u-1} = \mathbf{i}_2$	δ	0.95	δ	δ	δ
$\mathcal{I}_{u-1} = \mathbf{i}_3$	δ	δ	0.95	δ	δ
$\mathcal{I}_{u-1} = \mathbf{i}_4$	δ	δ	δ	0.95	δ
$\mathcal{I}_{u-1} = \mathbf{i}_5$	δ	δ	δ	δ	0.95

Table 3.7: CPD for $P(\mathcal{I}_u \mid \mathcal{I}_{u-1})$ after adding $\mathbf{i}_4$ and $\mathbf{i}_5$ to the intent state space.

3.2.4.1 Adding Intents

Additional intents can be created when new targets are detected in the environment. This can occur when a new vessel transits into the observation area or a previously undetected target is detected. Each new target is combined with the pertinent behaviors (as describe in Section 2.3) to generate a set of new intents. These intents are added to the intent state space and the CPD for the $\mathcal{I}_u$ node is created (Section 3.2.3). The prior probabilities for new intents is 0 since the contact could not have had that intent if it did not know about the target. For example, if a new vessel (X) enters the observation area at update $u-1$, two new intents are created using the method described in Section 2.3: $Intercept(X)$ and $Follow(X)$. Since these intents were not previously in the state space,

$$\begin{aligned} P(\mathcal{I}_d = \mathbf{i}_{Intercept(X)}) &= 0, \quad \text{and} \\ P(\mathcal{I}_d = \mathbf{i}_{Follow(X)}) &= 0, \quad \text{where } 1 \leq d \leq u-1. \end{aligned}$$

If there were three possible intents before the detection of the new vessel, then the intents $Intercept(X)$ and $Follow(X)$ would be added as $\mathbf{i}_4$ and $\mathbf{i}_5$ respectively. The CPD for the $\mathcal{I}_u$ node at the next update, u, is provided by Table 3.7. The values in rows four and five represent the transition probabilities of the contact's intent changing from that state to another. Regardless of what these values are, any calculations using these rows for the first update since the intents were added, at u, will be zero. This is because $P(\mathcal{I}_u = \mathbf{i}_i \mid \mathcal{I}_{u-1})$ is conditioned on the prior belief that the contact was in that state ($\mathbf{i}_i$). For $\mathbf{i}_4$ and $\mathbf{i}_5$ this value is 0. The result is justified since the probability that a contact's intent changes from a state it was not in is zero. Transitions from any of the previously known states to the new ones are possible and calculated in the same way as a transition between any two previous known states. Using this method, any number of new intents can be added to the intent state space for the intent estimator to consider.

3.2.4.2 Removing Intents

Three conditions require removing an intent from the model: environmental changes, the contact's inability to accomplish an intent, and the behavior definition. The first condition occurs if a target moves out of the observation area or is no longer detected by the UUV (the assumption was made in Section 2.2 that the contact would not pursue a target that could not be detected by the UUV). In this case the related intents should be removed from the state space since they will no longer represent valid states. For the second scenario, if the contact is unable to perform a certain intent, then the intent should also be removed. For example, if a contact is incapable of intercepting a target vessel because target is moving away from it faster than the contact's maximum speed, then the intent is invalid for the model and should be removed from the intent state space. In this illustration, the validity of the intent is dependent on the capabilities of the vessel, which is determined by the vessel type. As a result, the intent variable ($\mathcal{I}$) becomes dependent on the vessel type variable ($\mathcal{V}$). This dependency is incorporated into the Bayesian network as shown in Figure 3-7. The third condition occurs when a behavior may only be valid if certain conditions are met. For example, the intercept behavior may only be valid if the contact is at least a certain distance away from the targeted vessel. Otherwise, the intent to intercept that vessel would not be reasonable since the contact is already close enough to the target. For this thesis, it is arbitrarily assumed that the valid condition for all the behaviors modeled is when the contact is more than one minimum turning radius from the target but less than 8 nautical miles from it. An invalid behavior/target combination would require that the associated intent be removed from the state space.

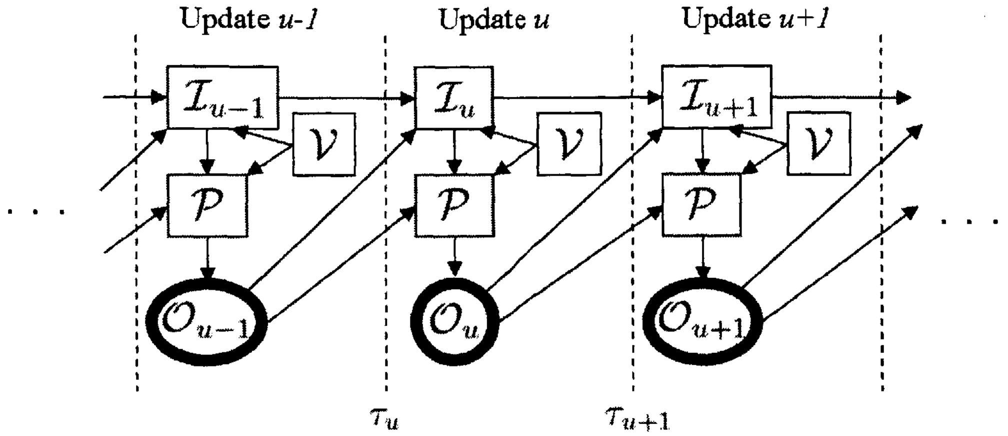


Figure 3-7: Modified Bayesian network to support a dynamic intent state space.

In this scenario, the validity of the intent depends on the contact-target relationship (obtained from $\mathcal{O}_{u-1}$), and the vessel type ($\mathcal{V}$). These two dependencies are also included in the Bayesian network as extended for dynamic intents (Figure 3-7).

The method for removing an intent from the state space is the same regardless of the reason for its removal. The removal of an intent state essentially prevents any transitions into that state. Since there is a possibility that the contact was in that state before it became invalid, transitions out of the invalid state is allowed. This is accomplished by making all the transition probabilities into the invalid node 0 and then normalizing the remaining transition probabilities for each node.

To illustrate the removal of an intent from the intent state space, a system with five intents is defined. The original transition probabilities from $\mathcal{I}_{u-1}$ to $\mathcal{I}_u$ is the same as the CPD provided by Table 3.7. If at update $u-1$, it is determined that $\mathbf{i}_4$ is an invalid intent, the transition probabilities into state $\mathbf{i}_4$ are set to 0 and all remaining transition probabilities are normalized. The result is shown in Table 3.8. The normalizing factor, η_i, is necessary to make sure that the sum of the transition probabilities out of each state is 1,

$$\sum_{j=1}^{N} P(\mathcal{I}_u = \mathbf{i}_j | \mathcal{I}_{u-1}) = 1, \quad 1 \le j \le N.$$

	$P(\mathbf{i}_1\|\mathcal{I}_{u-1})$	$P(\mathbf{i}_2\|\mathcal{I}_{u-1})$	$P(\mathbf{i}_3\|\mathcal{I}_{u-1})$	$P(\mathbf{i}_4\|\mathcal{I}_{u-1})$	$P(\mathbf{i}_5\|\mathcal{I}_{u-1})$
$\mathcal{I}_{u-1} = \mathbf{i}_1$	$\eta_1 0.95$	$\eta_1 \delta$	$\eta_1 \delta$	0	$\eta_1 \delta$
$\mathcal{I}_{u-1} = \mathbf{i}_2$	$\eta_2 \delta$	$\eta_2 0.95$	$\eta_2 \delta$	0	$\eta_2 \delta$
$\mathcal{I}_{u-1} = \mathbf{i}_3$	$\eta_3 \delta$	$\eta_3 \delta$	$\eta_3 0.95$	0	$\eta_3 \delta$
$\mathcal{I}_{u-1} = \mathbf{i}_4$	$\eta_4 \delta$	$\eta_4 \delta$	$\eta_4 \delta$	0	$\eta_4 \delta$
$\mathcal{I}_{u-1} = \mathbf{i}_5$	$\eta_5 \delta$	$\eta_5 \delta$	$\eta_5 \delta$	0	$\eta_5 0.95$

Table 3.8: CPD for $P(\mathcal{I}_u|\mathcal{I}_{u-1})$ after removing $\mathbf{i}_4$ from the intent state space given $\mathcal{V}$ and $\mathcal{O}_{u-1}$.

Each η in Table 3.8 is subscribed with the intent number (i) to indicate that the normalizing factor for each $\mathbf{i}_i$ is specific to that intent. From this table, it can be seen that a state transition from any previous state ($\mathcal{I}_{u-1}$) to the removed state ($\mathbf{i}_4$) is not allowed. However, transitions out of the removed state are allowed, as illustrated by the $\mathbf{i}_4$ row from the table. The constant values for the transition probabilities from state $\mathbf{i}_4$ to the valid intents indicates that if the contact was in state $\mathbf{i}_4$, then it is equally likely to change to any of the other intents. This CPD for $\mathcal{I}_u$ will then be used at update u and all subsequent updates until the next modification to the intent state space is made. After the first update using Table 3.8, the calculated probability belief value of the contact being in state $\mathbf{i}_4$ becomes zero. Leaving the $\mathbf{i}_4$ row and column in the CPD would not have a dramatic effect on the calculations of the intent estimator. Any calculations using the $\mathbf{i}_4$ row are conditioned on the probability of the contact being in that state (which is zero), and the $\mathbf{i}_4$ column just indicates that transitions into the $\mathbf{i}_4$ state are not allowed. The only consequence of leaving the intent in the state space is that it is included in calculating the transition probability between two different states (δ from Equation 3.10) and will therefore require the use of the normalization factor (η). Since the transition probability represented by δ is small, the effects are assumed to be negligible. In addition, if the removed intent became valid again, the CPD would return to the distribution represented by Table 3.7 and updated as if it was a new intent. The only difference is that it would retain its original intent number but that has no semantic effect. Using the method described, any number of intents can be removed from (with the ability to be reincorporated into) the intent state space.

The Bayesian network in Figure 3-7 represents the final intent estimation model implementing all the improvements presented in this section. Using this model, the equation used to calculate the intent belief state can be written as follows:

$$P_{\mathcal{T}_u}(\mathcal{I}_u = \mathbf{i}_i | \mathcal{O}_u, \mathcal{O}_{u-1}) = \eta \sum_{\mathcal{I}_{u-1}} \sum_{\mathcal{V}} \sum_{\mathcal{P}} \; P_{\mathcal{T}_u}(\mathcal{O}_u | \mathcal{P}) P(\mathcal{P} | \mathbf{i}_i, \mathcal{V}, \mathcal{O}_{u-1})$$
$$P(\mathcal{I}_u = \mathbf{i}_i | \mathcal{I}_{u-1}, \mathcal{O}_{u-1}) P(\mathcal{I}_{u-1}) P(\mathcal{V}),$$

where $1 \leq i \leq N$.

3.3 Forward-Backward Procedure

One characteristic of the method presented for determining the intent belief state has been that each update intent is only calculated using the current observation ($\mathcal{O}_u$), the previous belief state ($P(\mathcal{I}_{u-1})$), and the expected intent model created from the previous update (using $\mathcal{O}_{u-1}$). Although this method provides an estimate of the contact's intent belief state, the quality can be improved by calculating the belief state in the context of the entire observation sequence,

$$\boldsymbol{\mathfrak{O}} = \mathcal{O}_1 \mathcal{O}_2 \ldots \mathcal{O}_U,$$

where $\mathcal{O}_u$ is the observation made for update u, and U is the total number of observations made. This can be performed by implementing the *forward-backward procedure* to calculate a forward and backward variable on the observation sequence ($\boldsymbol{\mathfrak{O}}$), using the intent model. The forward variable represents the probability that the observation sequence will be made up to u given the resulting state of the contact (at u) and the intent estimation model. The backward variable is the probability of observing the rest of the observation sequence (after u) given the state of the contact at u and the model. The formulation in this section follows Rabiner [15] with slight notation modifications.

3.3.1 The Forward-Backward Intent Model

The advance intent estimation model presented in this chapter can be represented by three "probability measures" [15]. They are 1) the transition probabilities for $\mathcal{I}$ at all updates (which will be represented by A), 2) the observation probability at each update for each intent (represented by B), and 3) the initial probability distribution over all intents (represented by Π).

3.3.1.1 State Transition Probabilities

The state transition probability distributions are represented as
$A = \{a_u(i,j)\}$, where

$$a_u(i,j) = P(\mathcal{I}_{u+1} = \mathbf{i}_j | \mathcal{I}_u = \mathbf{i}_i), \quad \begin{array}{l} 1 \leq i,j \leq N \\ 1 \leq u \leq U, \end{array}$$

which is the probability that the contact will change its intent to $\mathbf{i}_j$ at update $u+1$, given that its original intent was $\mathbf{i}_i$ at u. This value depends on $\mathcal{I}_u$, $\mathcal{I}_{u+1}$, $\mathcal{V}$ and $\mathcal{O}_u$ (which is observable), as illustrated in Figure 3-7, and can be calculated as

$$a_u(i,j) = \eta \sum_{\mathcal{V}} P(\mathcal{I}_{u+1} = \mathbf{i}_j | \mathbf{i}_i, \mathcal{V}, \mathcal{O}_u) P(\mathcal{V}) P(\mathcal{O}_u).$$

Since the transition probabilities $(a(i,j))$ are allowed to change from one update to another (to support the dynamic state space incorporated in Section 3.2.4), the distributions used for each update are identified separately (by the u subscript) and recorded as they are created. In addition, the size of the intent state space (N) may also change. This only occurs when new intents are added to the state space since the removal of an intent is achieved by setting the transition probabilities into that state to zero. As a result, the value of N only increases and its maximum value can be represented by the size of the intent state space at the last update,

	$P(\mathbf{i}_1\vert\mathcal{I}_{u-1})$	$P(\mathbf{i}_2\vert\mathcal{I}_{u-1})$	$P(\mathbf{i}_3\vert\mathcal{I}_{u-1})$	$P(\mathbf{i}_4\vert\mathcal{I}_{u-1})$	$P(\mathbf{i}_5\vert\mathcal{I}_{u-1})$
$\mathcal{I}_{u-1} = \mathbf{i}_1$	0.95	δ	δ	0	0
$\mathcal{I}_{u-1} = \mathbf{i}_2$	δ	0.95	δ	0	0
$\mathcal{I}_{u-1} = \mathbf{i}_3$	δ	δ	0.95	0	0
$\mathcal{I}_{u-1} = \mathbf{i}_4$	0	0	0	0	0
$\mathcal{I}_{u-1} = \mathbf{i}_5$	0	0	0	0	0

Table 3.9: Expanded transition probabilities example, $P(\mathcal{I}_u|\mathcal{I}_{u-1})$.

U. To provide a constant state-space size for the forward-backward calculations, the transition probabilities ($a(i,j)$) for each update (u) will be expanded (if needed) to accommodate a size N state space, where N will be used to represent the maximum size that the state space reaches. The size of the intent state space at a particular update (u) will be represented by N_u. A probability of zero is used to fill in the expanded transitions:

$$\begin{aligned} a_u(i,j) = 0, \quad & \text{if } i \text{ or } j > |N_u| \\ & 1 \leq i,j \leq N \\ & 1 \leq u \leq U. \end{aligned}$$

For example, if the size of the intent state space at the first update ($u = 1$) is three, and the final size of the state space ($u = U$) is five, the transition probabilities for $a_1(i,j)$ will be expanded as shown in Table 3.9. The zeros in columns $\mathbf{i}_4$ and $\mathbf{i}_5$ prevents any transitions into those states (since they are not valid yet). Any calculations using the values in rows $\mathbf{i}_4$ and $\mathbf{i}_5$ will be zero regardless of the values in those rows since it will be conditioned on the probability of the contact originally having that intent, which is zero. Since the values in rows $\mathbf{i}_4$ and $\mathbf{i}_5$ have no affect on the intent estimation at that update, they are assigned a value of zero.

3.3.1.2 Observation Probability Distribution

The observation probability distribution for all updates made is represented as $B = \{b_{j,u}(\mathcal{O}_u)\}$, where

$$b_{j,u}(\mathcal{O}_u) = P(\mathcal{O}_u | \mathcal{I}_u = \mathbf{i}_j) \quad \begin{array}{l} 1 \le j \le N \\ 1 \le u \le U, \end{array}$$

which is the probability that the observation $(\mathcal{O}_u)$ will be made, given that the contact's intent was $\mathbf{i}_j$. This value depends on the variables $\mathcal{I}_u$, $\mathcal{V}$, $\mathcal{P}$, and $\mathcal{O}_u$ and $\mathcal{O}_{u-1}$ (which are observable), as illustrated in Figure 3-7, and can be calculated as

$$b_{j,u}(\mathcal{O}_u) = \sum_{\mathcal{V}} \sum_{\mathcal{P}} P_{\mathcal{T}_u}(\mathcal{O}_u | \mathcal{P}) P(\mathcal{P} | \mathbf{i}_j, \mathcal{V}, \mathcal{O}_{u-1}) P(\mathcal{V}) P(\mathcal{O}_{u-1}).$$

The probability distribution (b) over the observation space for each intent $(\mathbf{i}_j)$ is unique for each update and identified by the subscript u, representing the update it was created for. Similar to the state transition probabilities, the function $b_{j,u}(\mathcal{O}_u)$ is created at each update using the size of the intent state space at that time. This function is also modified to provide a constant state-space size for the forward-backward calculations. Using the previous notation, the modification to $b_{j,u}(\mathcal{O}_u)$ is as follows

$$b_{j,u}(\mathcal{O}_u) = 0, \text{ if } j > |N_u|, \quad \begin{array}{l} 1 \le j \le N \\ 1 \le u \le U. \end{array}$$

This modification states that the probability of observing the contact at $\mathcal{O}_u$ if it had an intent that did not exist is zero.

3.3.1.3 Initial Probability Distribution

The initial probability distribution over the intent state space is represented as $\Pi = \{\pi_i\}$, where

$$\pi_i = P(\mathcal{I}_u = \underset{i}{\mathbf{i}}), \quad 1 \leq i \leq N_u$$
$$u = 0,$$

which represents the initial belief state of the intent variable before any updates are made ($u = 0$). The distribution is over the number of intents at the start of the intent estimation process (N_0). Since π is also used in the forward-backward calculations, it is expanded to accommodate the final state-space size (N) as follows

$$\pi_i = 0, \text{ if } i > |N_u|, \quad 1 \leq i \leq N$$
$$u = 0.$$

Combining the three probability measures, the forward-backward intent model (λ) is

$$\lambda = (A, B, \Pi).$$

3.3.2 Forward Variable

From [15], the forward variable ($\alpha_u(i)$) is defined as

$$\alpha_u(i) = P(\mathcal{O}_1\mathcal{O}_2 \ldots \mathcal{O}_u, \mathcal{I}_u = \underset{i}{\mathbf{i}}|\lambda),$$

representing the probability of the partial observation sequence, $\mathcal{O}_1\mathcal{O}_2 \ldots \mathcal{O}_u$, (until update u) and being in state $\underset{i}{\mathbf{i}}$ at update u ($\mathcal{I}_u = \underset{i}{\mathbf{i}}$), given the model λ. The forward variable can be calculated inductively as follows:

1. Initialization:

$$\alpha_1(i) = \pi_i b_{i,1}(\mathcal{O}_1), \quad 1 \leq i \leq N.$$

2. Induction:

$$\alpha_{u+1}(j) = \left[\sum_{i=1}^{N} \alpha_u(i) a_u(i,j)\right] b_{j,u+1}(\mathcal{O}_{u+1}), \quad 1 \leq u \leq U-1$$
$$1 \leq j \leq N.$$

3.3.3 Backward Variable

From [15], the backward variable ($\beta_u(i)$) is defined as

$$\beta_u(i) = P(\mathcal{O}_{u+1}\mathcal{O}_{u+2}\ldots\mathcal{O}_U|\mathcal{I}_u = \mathbf{i}_i, \lambda),$$

representing the probability of the partial observation sequence from $u+1$ to the end, given $\mathbf{i}_i$ at update u and the model λ. The backward variable can also be calculated inductively as follows:

1. Initialization:

$$\beta_U(i) = 1, \quad 1 \leq i \leq N.$$

2. Induction:

$$\beta_u(i) = \sum_{j=1}^{N} a_u(i,j) b_{j,u+1}(\mathcal{O}_{u+1}) \beta_{u+1}(j), \quad 1 \leq i \leq N$$
$$u = U-1, U-2, \ldots, 1.$$

3.3.4 Retrospective Belief State

The forward and backward variables can now be used to determine the intent belief state (γ) for each update u, in the context of the entire observation sequence $\mathfrak{Q}$. This can be computed as follows [15]:

$$\gamma_u(i) = \eta \alpha_u(i) \beta_u(i), \quad 1 \leq i \leq N,$$

where γ_u is the retrospective belief state for update u and η is a normalizing factor. This is an improved calculation of the belief state since it calculates the belief state using the entire observation sequence.

3.4 Summary

An advance intent estimation model was developed in this chapter by expanding the model introduced in Chapter 2. The advanced model was developed to account for multiple uncertain variables concerning the contact (such as its speed and turn capabilities and movement behaviors). Further, the intent estimator was also extended to be capable of operating in a dynamic surrounding by allowing the intent belief state to change in response to the contact's decisions and changes in its environment. Finally, the belief state calculation was improved by incorporating a forward-backward procedure to include the entire observation sequence for calculating a retrospective intent belief state.

[This page intentionally left blank]

Chapter 4

Obstacle Detection

At times, a UUV may be deployed to a region where certain information concerning the environment is not available. Such information may include the location of obstacles in the environment that could present a hazard to navigation. Examples of such obstacles would include shipwrecks, shallow water areas, and minefields. Some of these obstacles, such as shipwrecks and shallow water zones, may not be known if the UUV is sent into an area which has not been surveyed yet or the regional chart is unavailable. Other obstacles may be intentionally hidden, such as minefields, in which case there would be no prior way of determining where they may be. However, the ships that normally transit the area would most likely know where the obstacles are in the environment and would maneuver to avoid them. It would therefore be beneficial if the UUV could analyze the movements of the contacts it observes to infer the location of any obstacles in the environment. This is the objective of *obstacle detection* by intent estimation.

4.1 Overview

The presence and location of unknown obstacles in an environment can be hypothesized by observing a contact's path through an area and speculating that any indirect movements of the contact to accomplish its intent resulted from avoiding obstacles.

```
OBSTACLE-DETECTION(contact's path ($G$), obstacle inference map ($M$))
  Separate the contact's path into intent blocks ($b$).
  For each intent block ($b_j$):
      Calculate a single intent belief state for the block ($P(\mathcal{I}_{b_j})$).
      For each intent ($\mathfrak{i}_i$) in the intent state space ($\mathfrak{I}$):
          Determine the intent destination.
          Analyze the path segment for indirect movements.
          Create a hypothetical obstacle map ($H_{b_j,i}$) to explain the path segment.
      Combine each $H_{b_j,i}$ to produce a combined hypothetical obstacle map ($H_{b_j}$).
      Incorporate $H_{b_j}$ into the obstacle inference map ($M$).
      Decrease the probability of obstacles along the contact's path in $M$.
```

Figure 4-1: Obstacle detection overview.

The process used in this thesis to translate the observed path of the contact (from the observation sequence) into the location of obstacles is presented in Figure 4-1. The obstacle detection process begins with the observed contact's path (G) and a probability map to record the possible locations of inferred obstacles, M (the obstacle inference map). The path is first divided into intent blocks,

$$\{b_1, b_2, \ldots, b_J\},$$

using the calculations from the intent estimator (Section 4.4). Each block corresponds to a unique segment of the contact's path (G) and will be represented as g_j where

$$G = g_1 + g_2 + \ldots + g_J. \tag{4.1}$$

A given b_j represents a g_j which was believed to have been made by the contact while it pursued a single intent. Since the exact intent of the contact is not known for each g_j, a probability distribution over the contact's possible intents (intent belief state) is calculated for the corresponding b_j. The belief state for an intent block, b_j, will be represented as $P(\mathcal{I}_{b_j})$.

For a given g_j, each intent identifies the destination of the contact during that b_j if it had that intent ($\mathfrak{i}_i$). The g_j is then analyzed for any indirect movements made by the contact to reach each destination. An indirect movement is considered to be a deviation from a straight-line path from the contact's observed position to the destination. A hypothetical obstacle map ($H_{b_j,i}$) is next generated for each intent ($\mathbf{i}_i$) which would explain any indirect movements found in the path segment for b_j. The hypothetical obstacle map for each intent is then weighted with its respective intent belief value and combined to create a combined hypothetical obstacle map (H_{b_j}) as follows

$$H_{b_j} = \sum_{i=1}^{N} P(\mathcal{I}_{b_j} = \mathbf{i}_i) H_{b_j,i}. \tag{4.2}$$

Finally, H_{b_j} is incorporated into the obstacle inference map (M) to record the probability of any obstacles derived from the path segment. The path segment (g_j) analysis, creation of a combined hypothetical obstacle map (H_{b_j}), and incorporation of that map to into the obstacle inference map (M) is performed for each intent block (b_j). The path of the contact is also incorporated into M to decrease the belief probability of obstacles in the areas that the contact travelled through. The resulting M can be updated with additional paths from the observations of other contacts to provide a combined probability map of obstacles in the environment.

4.2 Assumptions

The following major assumptions are made.

1. The Closest Point of Approach (CPA) that a contact would make in relation to an obstacle is known. The CPA is defined as the closest distance that the contact would move next to an obstacle. This distance can vary from one vessel type to another and even among vessels of the same type (as the CPA can be set by the ship's captain). To simplify this issue, the CPA has been arbitrarily set in this thesis to equal the minimal turning radius of the contact. This distance would hypothetically provide the contact with the ability to maneuver without running into an obstacle as long as the maneuver was initiated at least one turning radius away from any obstacles. This assumption simplifies the

development of the obstacle map and provides the size of the area around the contact's path which can be assumed to be clear by the passage of the contact. To further simplify the calculations, only one CPA will be considered. As a result of this simplification, and the assumed relationship between the contact's CPA and its capabilities, only one vessel type is considered.

2. All targets and obstacles in the map are stationary objects in the environment. This assumption simplifies the calculation of the destination of the contact (which is the position of the object for stationary targets).

3. The contact pursues the most direct path possible to its intended target. All deviations from a direct approach to the projected target result from obstacles in the environment.

4.3 Obstacle Inference Map

The obstacle inference map (M) used to maintain the probability about the location of obstacles in the environment uses a fixed rectilinear grid imposed on the environment. Each cell in the grid indicates the likelihood in [0,1] that the cell contains an obstacle (is occupied). A value of 0 corresponds to the belief that the area is cleared of an obstacle (empty) while a value of 1 means that the area has an obstacle. A value of 0.5 represents complete uncertainty (i.e., such a value indicates that there is no information available concerning an obstacle, or the absence of an obstacle, in that area). For this thesis, it is assumed that there is no prior information concerning the presence and location of any obstacles in the environment. Therefore, each cell in M is initially assigned the value of 0.5. The values in the cells that correspond to the location of the stationary targets in the environment, which are known, are set to 1 and will remain unchanged. The expression "a cell is occupied" is used interchangeably with "an area which the cell represents is occupied." This map of probabilities of obstacle locations will be updated with the hypothetical obstacle maps created from each contact path observed.

4.4 Intent Blocks

Since an observed contact has the ability to change its intent as often as it desires and at any given time, its observed path (G) may be an aggregate of smaller path segments (g_j) with different intents. To prevent the entire path from inadvertently being analyzed for one intent, the path is divided into segments for which the contact was believed to have had a consistent intent. These segments will be referred to as intent blocks (b_j) and represents an appropriate path segment, g_j (Equation 4.1).

To divide the path into intent blocks, the intent estimator developed in Chapters 2 and 3 is used to determine the various intent belief states of the contact along the observed path. Since the observed path of the contact is defined by a series of observations made of the contact at various times, the path provides the intent estimator with all the information required to perform its calculations (an observation of the contact and the elapsed time since the last update). An intent belief state can be generated for each observation point used to define the observed path. The only exception is the first observation, which initializes the intent estimation process. From the assumptions made in Section 4.2, only one vessel type is considered. Therefore, the vessel type variable ($\mathcal{V}$—from Section 3.2.2) used in the calculations by the intent estimator is known. The results from the intent estimation process are then used to segment the contact's observed path into intent blocks. This was done in this thesis by grouping together consecutive updates which have the same predominant intent.

Segmenting the path is necessary to determine which continuous blocks of the observed path were committed to pursuing a certain intent and separating them from the parts of the path that were pursuing a different intent. Each g_j (from the corresponding b_j) is then analyzed separately to determine how that segment deviated from a direct approach to its intended target.

Before a g_j can be analyzed for deviations from the direct path to a certain destination, the target that the contact is trying to reach needs to be identified. Since the intent of the contact cannot be directly observed, the deviations of the path segment for *each* intent are considered. The results are then combined using a probability distribution over the intent state space for the corresponding b_j. The distribution will be referred to as the intent block's belief state and represented as $P(\mathcal{I}_{b_j})$. Since $P(\mathcal{I}_{b_j})$ represents the individual intent belief states ($P(\mathcal{I}_u)$) within the intent block (b_j), a straightforward method for determining $P(\mathcal{I}_{b_j})$ is to sum the belief value for each $\mathbf{i}_i$ in b_j and then normalize it as follows

$$P(\mathcal{I}_{b_j} = \mathbf{i}_i) = \frac{\sum_u P(\mathcal{I}_u = \mathbf{i}_i)}{\sum_u \sum_{j=1}^{N} P(\mathcal{I}_u = \mathbf{i}_j)}, \quad 1 \leq i \leq N \text{ and } u_s \leq u \leq u_f, \qquad (4.3)$$

where $P(\mathcal{I}_{b_j} = \mathbf{i}_i)$ is the probability of the contact having intent $\mathbf{i}_i$ during the intent block defined as starting with update s (u_s) and ending with update f (u_f). The intent belief value for a given intent ($\mathbf{i}_i$) at a update (u) in the block is represented by $P(\mathcal{I}_u = \mathbf{i}_i)$. Equation 4.3 computes the intent belief state for the intent block by taking the sum of each intent belief value (from each update) during the provided block and then dividing that value by the sum of all the belief values in that block to normalize $\sum_u P(\mathcal{I}_u = \mathbf{i}_i)$ into a probability value.

The sum of all the values in the intent belief state for each update is 1,

$$\sum_{j=1}^{N} P(\mathcal{I}_u = \mathbf{i}_j) = 1,$$

which allows the denominator from Equation 4.3 to be rewritten as

$$\sum_u \sum_{j=1}^{N} P(\mathcal{I}_u = \mathbf{i}_j) = \sum_u 1 = u_f - u_s + 1, \quad u_s \leq u \leq u_f. \qquad (4.4)$$

Combining Equations 4.3 and 4.4 produces

$$P(\mathcal{I}_{b_j} = \mathbf{i}_i) = \frac{\sum_u P(\mathcal{I}_u = \mathbf{i}_i)}{u_f - u_s + 1}, \quad 1 \leq i \leq N \text{ and } u_s \leq u \leq u_f. \qquad (4.5)$$

Equation 4.5 indicates that each intent belief value for the intent block is actually the average of that intent's belief value from the individual updates in the block. Therefore, the $P(\mathcal{I}_{b_j})$ is the average of the individual intent belief states provided by the updates in that block.

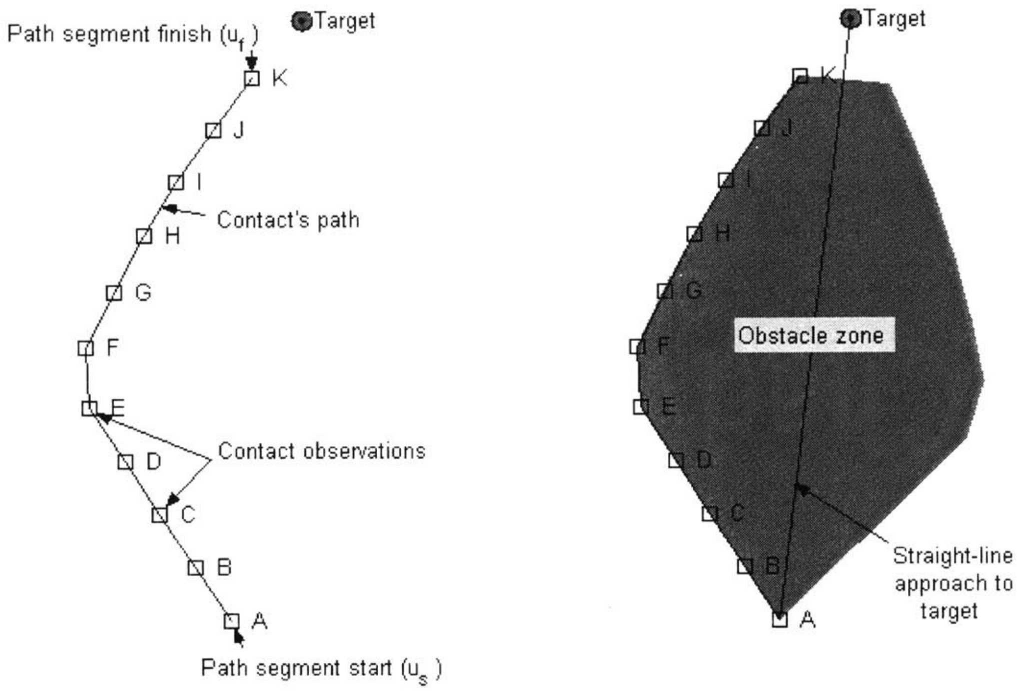
Finally, the original observed path is divided into path segments with each segment having a single intent belief state of the contact's intent.

4.5 Path Segment Analysis

Once the path segments are defined, each path segment (g_j) is analyzed using the assumption that the most direct path from a contact's location to its destination is a straight line path (SLP) connecting the two points. Any deviation from this line is assumed to be the result of the contact maneuvering around obstacles.

4.5.1 Initial Analysis Method

Even though the exact position, size, and shape of the hypothesized obstacles cannot be determined by a contact's deviation from a SLP, the general area that the obstacles occupy can be approximated. This is done by classifying the space between the SLP and the contact's actual path as an obstacle zone. The shape of the obstacle zone is dictated by the path of the contact. This is a reasonable approximation of the basic shape of the obstacle zone since it is assumed that the location of the obstacles generally influence the shape of a contact's path. A mirror image of this zone is also created on the opposite side of the SLP. This second zone is needed to explain why the contact did not simply travel to the opposite side of the SLP to avoid the first obstacle zone created. The two half zones combined represents the obstacle zone used to explain the indirect movements of the contact for that path segment. An example is provided in Figure 4-2. Figure 4-2(a) displays an observed path segment with a destination (identified as the target). The path segment is represented by the series of contact observations (which are indicated by the boxes in the figure).

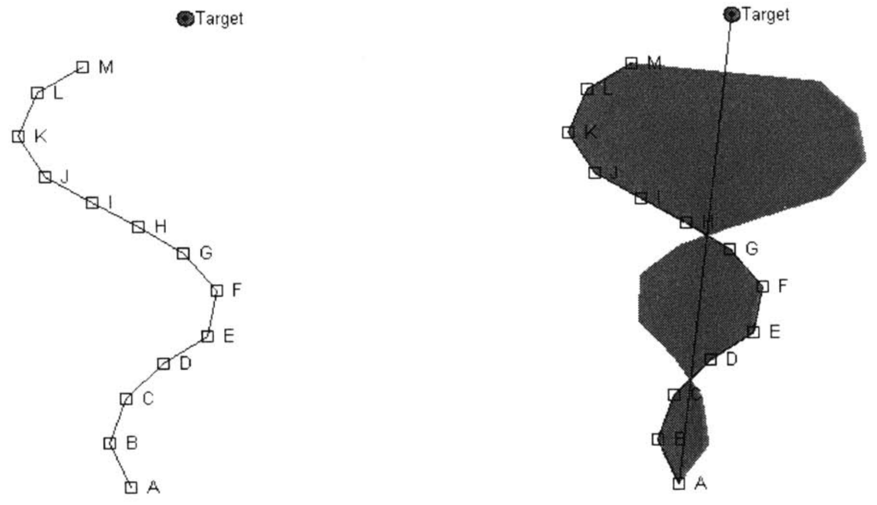


(a) Path segment.

(b) Derived hypothetical obstacle zone (right-hand side is the mirror image).

Figure 4-2: General path segment analysis (boxes indicate observed contact positions).

In this example, the path segment provided was created from 11 observations made of the contact labelled sequentially from A to K. Point A is the start of the path segment for the given intent (as determined by u_s from the intent block) and K is the last observation made of the contact for the same intent (u_f). Analyzing this path segment using the method just described produces the obstacle zone illustrated in Figure 4-2(b). From Figure 4-2(b), it can be seen that the obstacle zone extends toward the target only to the end of the intent block (it stops at point K). The reason the obstacle zone was cut off at the end of the path segment is because no other information is available to further define the obstacle zone. The zone cannot be shaped once the path segment terminates (which may occur when the contact changes intent). In addition, the obstacle zone does not stop at point F (at which point the contact seemed to have taken a relatively straight path to the target). This



(a) Complex path segment.

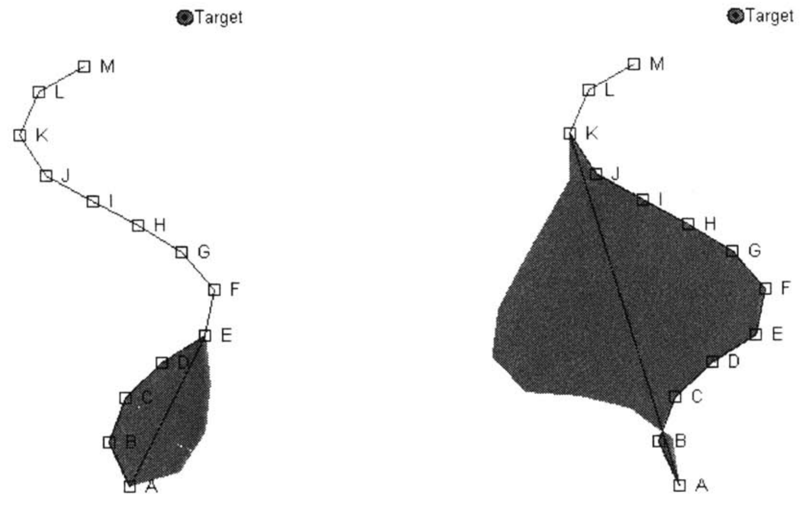
(b) Derived obstacle zones using the original analysis method.

Figure 4-3: General path segment analysis for a complex path.

is because it is plausible that an obstacle may exist in the obstacle zone area past F, even though the contact travelled in a straight line past that area. For example, from the Figure 4-2(b), an obstacle could exist in the area of the obstacle zone near the target (past F) which would cause, along with other obstacles in the area, the contact to move out to point F first. From F, the contact would then have enough clearance to move straight to the target and avoid the obstacles.

4.5.2 Improved Analysis Method

Determining the obstacle zone for the path segment in the example just provided can be relatively straightforward. For a more complicated path segment, such as the one illustrated in Figure 4-3(a), intermediate destinations need to be considered in order to properly explain the shape of the contact's path. For example, using the starting position of the path segment (A), the original method used to determine the obstacle zones (presented in Section 4.5.1) is illustrated in Figure 4-3(b). The purpose of the obstacle zones is to identify locations in the environment where obstacles may exist to help explain why the contact maneuver in the manner it was observed to reach its



(a) Obstacle Zone for Reaching Point E. (b) Obstacle Zones for Reaching Point K.

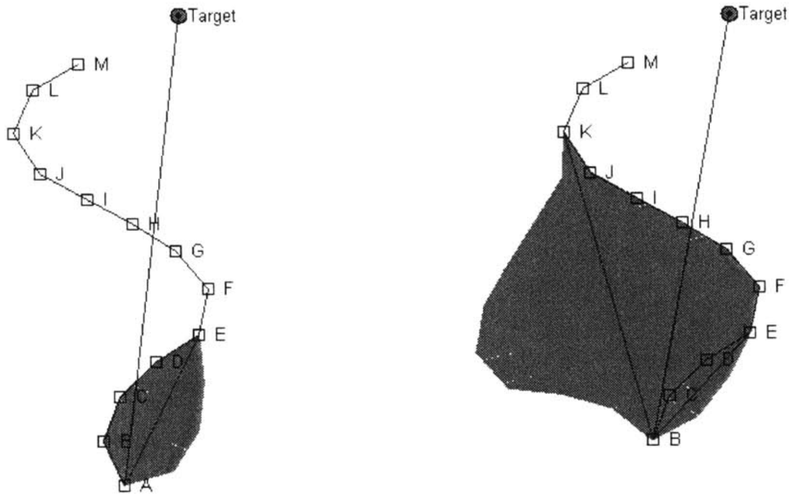
Figure 4-4: Intermediate Obstacle Zones.

destination. The observed path is assumed to be the most direct path available to the contact for reaching the destination. Using the contact's path and obstacle zones from Figure 4-3(b), it can been seen that a more direct path for reaching the target is available by traveling from point A directly to point K and then continuing the path from there. Therefore, the contact's path cannot be explained by the obstacle zones created using the original path analysis method.

To provide an improved method for determining the obstacle zones, the indirect movements along the path are analyzed. For example, in the path shown in Figure 4-3, the contact's movement towards points E and K were not direct. However, the contact's movements toward point E could be explained by the presence of obstacles in the zone illustrated in Figure 4-4(a). This zone was determined by applying the method for creating an obstacle zone (from Section 4.5.1) with point E as the destination. The same can be done for point K as illustrated in Figure 4-4(b). The obstacle zones illustrated in Figure 4-4 provides the explanations for the indirect movements that the contact made in reaching points E and K from point A.

For each position along the contact's path, an indirect movement made by the contact to either the destination or a future position along the observed path needs to be considered. Since the contact's path is defined by a series of observations made of the contact, any calculations using the contact's path only considers these positions. Therefore, the phrase "each position along the contact's path" is used interchangeably with "each observation made of the contact." To identify the indirect movements made by the contact while moving from one position to another along its path, every future position of the contact would have to be considered from every previous position. To simplify this calculation, only two future positions are considered for each previous position along the observed path—these positions are identified as an intermediate destination. In this thesis, an intermediate destination is chosen by determining the point along the path which would create a vector (with the starting position) that would produce the greatest angle from the SLP to the original target. The position that produces the largest angle would indicate the greatest deviation from the direct path. The deviation is only considered in terms of an angle away from the direct course to the target which may not always result in selecting the point furthest in distance from the SLP. Since the contact may deviate from the direct approach path by moving either to the left or the right, the point representing the largest deviation on each side of the SLP is selected, producing two intermediate destinations. Using the starting location at A in Figure 4-3, these points would be the locations indicated by B and E with the resulting obstacle zone shown in Figure 4-5(a). The process is then repeated to determine the obstacle zones for each observation used to define the path segment. These obstacle zones can then be combined to provide the comprehensive obstacle zones that can explain the indirect movements of the contact towards its destination and along the path.

For observation B, the two intermediate destinations are indicated by E and K with the obstacle zones illustrated in Figure 4-5(b). The portion of the path segment prior to the observation point being used is ignored since it does not provide any information concerning possible obstacle zones from observation B. At observation J, the intermediate destination for the left side of the remaining path segment portion



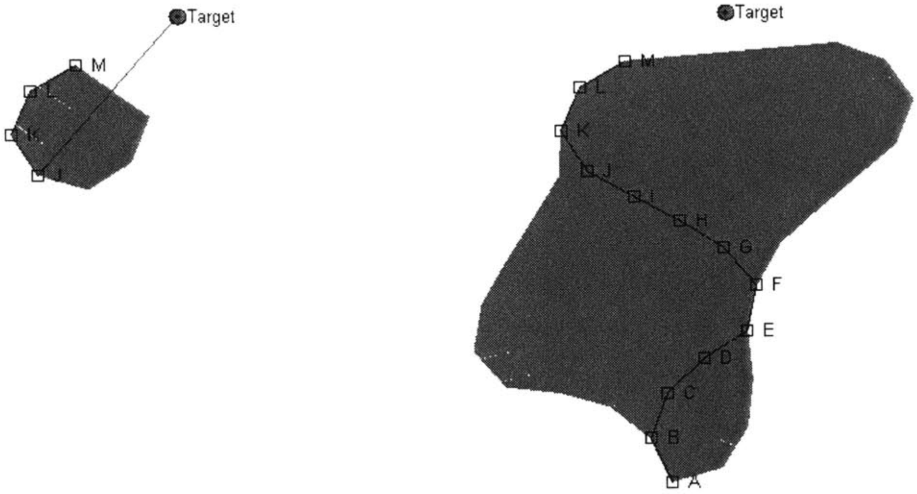
(a) Obstacle zones for contact at location A. (b) Obstacle zones for contact at location B.

Figure 4-5: Example obstacle zones for 2 observations.

illustrated by Figure 4-6(a) is K. However, there is no intermediate destination for the right side. In this situation, the intermediate destination is the primary destination and the obstacle zones are created as illustrated. The combination of all the obstacle zones created from each observation point along the provided path segment is illustrated in Figure 4-6(b). These zones are used to represent the areas where obstacles are located which would cause the contact to move along the path segment in the manner it was observed.

4.5.3 Destination Located Behind Contact's Path

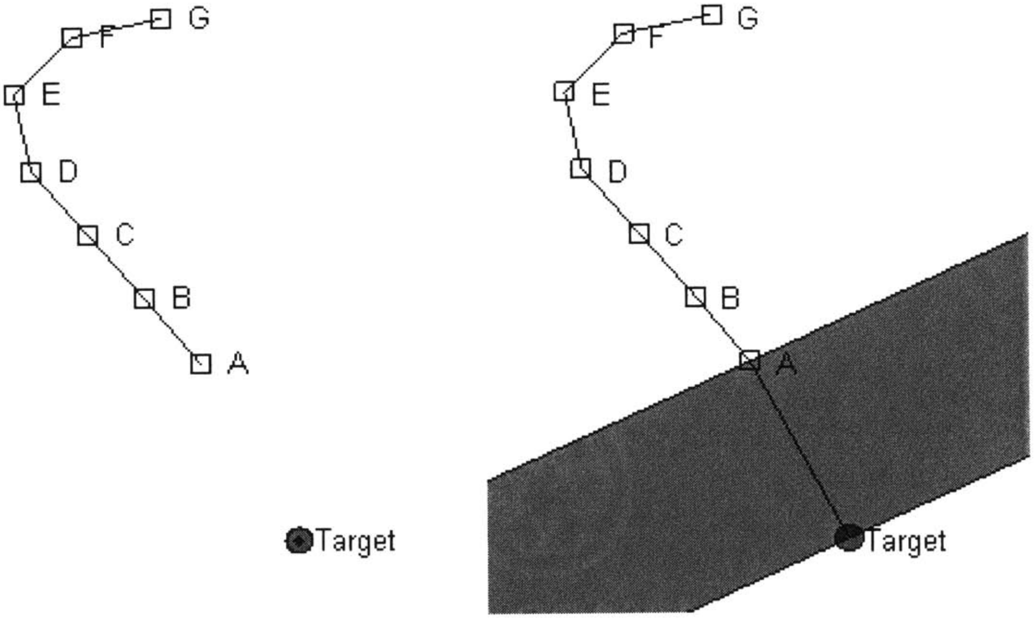
In the situation where the primary destination is located *behind* the contact's path, as illustrated in Figure 4-7(a), the original method for determining the intermediate destinations (described in Section 4.5.2) would still identify two positions along the observed path to be used for creating obstacle zones. The primary destination is considered to be behind the contact's path when no portion of the path segment is present in the area bounded by two lines perpendicular to the SLP and located at the start of the path segment and the primary destination. This area is shaded in



(a) Obstacle zones for contact at location J.

(b) Combined obstacle zones.

Figure 4-6: Additional obstacle zones examples.



(a) Target located behind the path segment.

(b) Required path zone.

Figure 4-7: Destination located behind contact's path.

Figure 4-7(b). The accuracy of the obstacle zones created is unknown since the actual obstacle zones for the scenario shown in Figure 4-7 is difficult to determine.

4.6 Hypothetical Obstacle Map

Now that a number of intent-based obstacle zones have been identified, a combined hypothesis map will be created. This is accomplished by creating a hypothetical obstacle map for each intent using its associated obstacle zones (if any) to transform these zones into an obstacle probability map which are then combined and incorporated into the primary obstacle inference map. The problem with an optimal solution will be briefly discussed and then two approximations are presented.

Each hypothetical obstacle map has the same structure as the obstacle inference map described in Section 4.3. Since the hypothetical obstacle map uses grid cells to represent a fixed area in the environment, the number of grid cells required to cover an obstacle zone may be fairly large—depending on the size of the obstacle zone and the area represented by a single cell. For instance, the obstacle zone illustrated in Figure 4-8, is represented by 307 cells. The target is also identified as occupied space (from Section 4.3) but since it is known to be present, its occupancy probability is fixed at 1. To determine the optimal probability for each cell in the zone of being occupied, all possible occupied cell combinations would need to be considered and determined if that combination of cells would cause the contact to move along the path that was observed. For the obstacle zone provided, this involves 2^{307} combinations. Such an approach is intractable. Therefore, in order to establish a probability of occupancy for the cells in an obstacle zone, an approximate method is needed to translate each cell into a probability value.

Two different approximations were implemented and tested to perform this task. The first is to assign a "blanket" probability to all the cells in the obstacle zone. This approach indicates that each cell within the obstacle zone is equally likely of being occupied and is illustrated in Figure 4-9(a) where each cell in the zone is arbitrarily

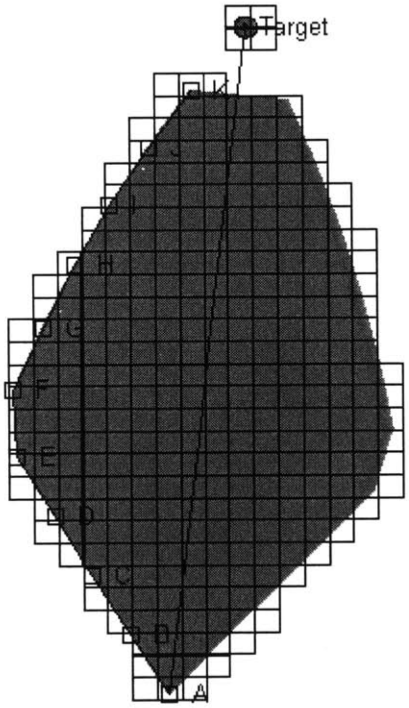
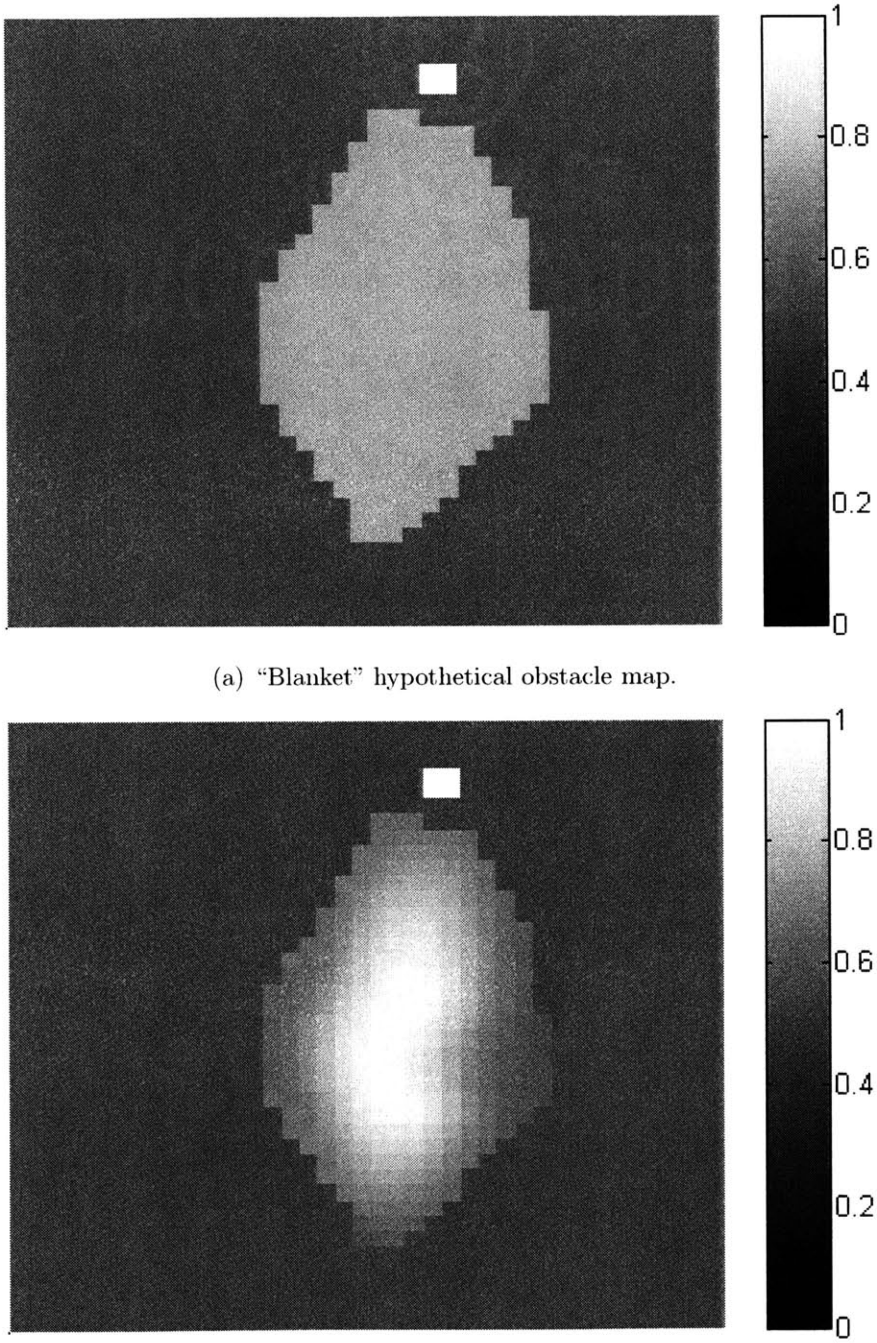


Figure 4-8: Gridded obstacle zone with 306 cells.



(a) "Blanket" hypothetical obstacle map.

(b) "Shaped" hypothetical obstacle map.

Figure 4-9: Hypothetical obstacle map examples. The black to white gradient represent the probability of occupancy from [0, 1].

assigned a probability value (say 0.75). This value is determined by the user and has been set in this scenario as the midpoint between the unknown (0.5) and occupied (1.0) probability values. The cells outside the obstacle zone are assigned a probability value of 0.5 since a zone provides no occupancy information for the cells outside its boundary. The figure displays the value in each cell using a gradient from black to white to represent the probability distribution between [0, 1].

The second approximation method used was to "shape" the obstacle field inside the obstacle zone by assigning a higher probability to areas inside the zone which are more likely to be occupied than others. The technique used to shape each zone is specified by the user's perception of how obstacles are usually situated in an obstacle zone. For example, there may be a stronger belief that obstacles are present in the area of the zone that is relatively close to the contact's path but not right next to it (since a contact is likely to stay some distance from an obstacle for safety reasons). As the distance from this area increases in the obstacle zone, the belief of an obstacle being in that area may not be as strong and is therefore assigned a lower probability value. This approach is shown in Figure 4-9(b). The specific process to generate this map is described in Appendix C. Similar to the first method, the cells representing the areas of the map outside the obstacle zone are assigned the value 0.5.

For each intent ($\mathbf{i}_i$), the obstacle zones can then be transformed into probability values using whichever method was selected to produce the hypothetical obstacle map ($H_{b_j,i}$) for the path segment being analyzed (g_j). The hypothetical obstacle map for each intent is weighted using the respective belief value from the identifying intent block (b_j) and combined to create the primary hypothetical obstacle map, H_{b_j} (Equation 4.2). This map provides an inferred probability belief for each cell in the map of being occupied using the observed path segment. The map is then incorporated into the obstacle inference map, M.

4.7 Map Integration

Once the hypothetical obstacle map (H_{b_j}) for a path segment (g_j) had been created, the information contained in that map is then incorporated into the obstacle inference map by combining the two together. This is accomplished by using a map-combining formula presented by Moravec in [12]. This formula uses Bayes' rule to combine "independent sources of information... into an estimate of a single quantity." In this situation, the two independent sources of information are the separate maps and the single quantity of concern is the probability that a given cell is occupied.

The probability of a given cell being occupied is represented as $P(o)$. The complement of $P(o)$ is the probability that the cell is not occupied and will be represented by $P(\neg o)$. By definition,

$$P(\neg o) = 1 - P(o) = \neg P(o).$$

The objective is to combine the $P(o)$ of a cell from the obstacle inference map with the $P(o)$ provided by the hypothetical obstacle map. Since the obstacle inference map incorporates each hypothetical obstacle map created from the previous path segments (old information), the value of a cell in the obstacle inference map can be represented as $P(o|A)$, where A represents the old information. Similarly, the value of a cell in the hypothetical obstacle map created using the new information (B) can be represented as $P(o|B)$. Combining these two maps would produce $P(o|A, B)$—the probability that a cell is occupied using both the old and new information. Moravec uses Bayes' rule to represent $P(o|A, B)$ and its complement as follows:

$$P(o|A,B) = \frac{P(B|o,A)P(o|A)}{P(B|o,A)P(o|A) + P(B|\neg o,A)P(\neg o|A)}, \tag{4.6}$$

$$P(\neg o|A,B) = \frac{P(B|\neg o,A)P(\neg o|A)}{P(B|o,A)P(o|A) + P(B|\neg o,A)P(\neg o|A)}. \tag{4.7}$$

Equations 4.6 and 4.7 can then be combined to produce

$$\frac{P(o|A,B)}{P(\neg o|A,B)} = \frac{P(B|o,A)}{P(B|\neg o,A)}\frac{P(o|A)}{P(\neg o|A)}. \tag{4.8}$$

Applying Equation 4.8 with information B and null A produces

$$\frac{P(o|B)}{P(\neg o|B)} = \frac{P(B|o)}{P(B|\neg o)}\frac{P(o)}{P(\neg o)}, \tag{4.9}$$

where $P(o)$ is the initial probability that the given cell is occupied before any information is considered and $P(\neg o)$ is its complement. Equation 4.9 can be rewritten as

$$\frac{P(B|o)}{P(B|\neg o)} = \frac{P(o|B)}{P(\neg o|B)}\frac{P(\neg o)}{P(o)} \tag{4.10}$$

to illustrate the relationship between $P(o|B)$ and $P(B|o)$. If the assumption is made that A is independent from B, the following equation can be used

$$\frac{P(B|o,A)}{P(B|\neg o,A)} = \frac{P(B|o)}{P(B|\neg o)}. \tag{4.11}$$

In the context of this thesis, the independence assumption between A and B would indicate that no additional information would be gained by using the old and new path segments together to determine the occupancy of a cell. In other words, calculating the occupancy of a cell from A and B separately and then combing the values would produce the same result as determining the cell occupancy using A and B together. In general, this assumption will produce an inferior estimation of a cell's occupancy probability when compared to a calculation made without such an assumption. This is true since two or more paths observed transiting though a common area can be used together to provide a more accurate obstacle map by considering all possible occupied cell combinations (using the pertinent cells) which would produce the observed deviations for all the paths observed instead of only one. However, such a problem becomes intractable (as mentioned in Section 4.6). As a result, the independence assumption of A from B allows the use of Equation 4.11. Using Equations 4.8, 4.10, and 4.11, the map-combining formulation is obtained

$$\frac{P(o|A,B)}{P(\neg o|A,B)} = \frac{P(o|B)}{P(\neg o|B)}\frac{P(o|A)}{P(\neg o|A)}\frac{P(\neg o)}{P(o)}. \tag{4.12}$$

The values for the terms $P(\neg o|A,B)$, $P(\neg o|B)$, $P(\neg o|A)$, and $P(\neg o)$ are the complement of the terms $P(o|A,B)$, $P(o|B)$, $P(o|A)$, and $P(o)$ respectively. The occupancy probability for the given cell calculated from the new information (B) is represented by $P(o|B)$, and $P(o|A)$ is the probability of the cell being occupied given the old information (A). Identifying $P(\neg o|A,B)$ as the complement of $P(o|A,B)$,

$$P(\neg o|A,B) = 1 - P(o|A,B).$$

Equation 4.12 can be solved for $P(o|A,B)$ producing

$$P(o|A,B) = \frac{Y}{1+Y}, \tag{4.13}$$

where Y represents the terms on the right hand side of Equation 4.12

$$Y = \frac{P(o|B)}{P(\neg o|B)}\frac{P(o|A)}{P(\neg o|A)}\frac{P(\neg o)}{P(o)}. \tag{4.14}$$

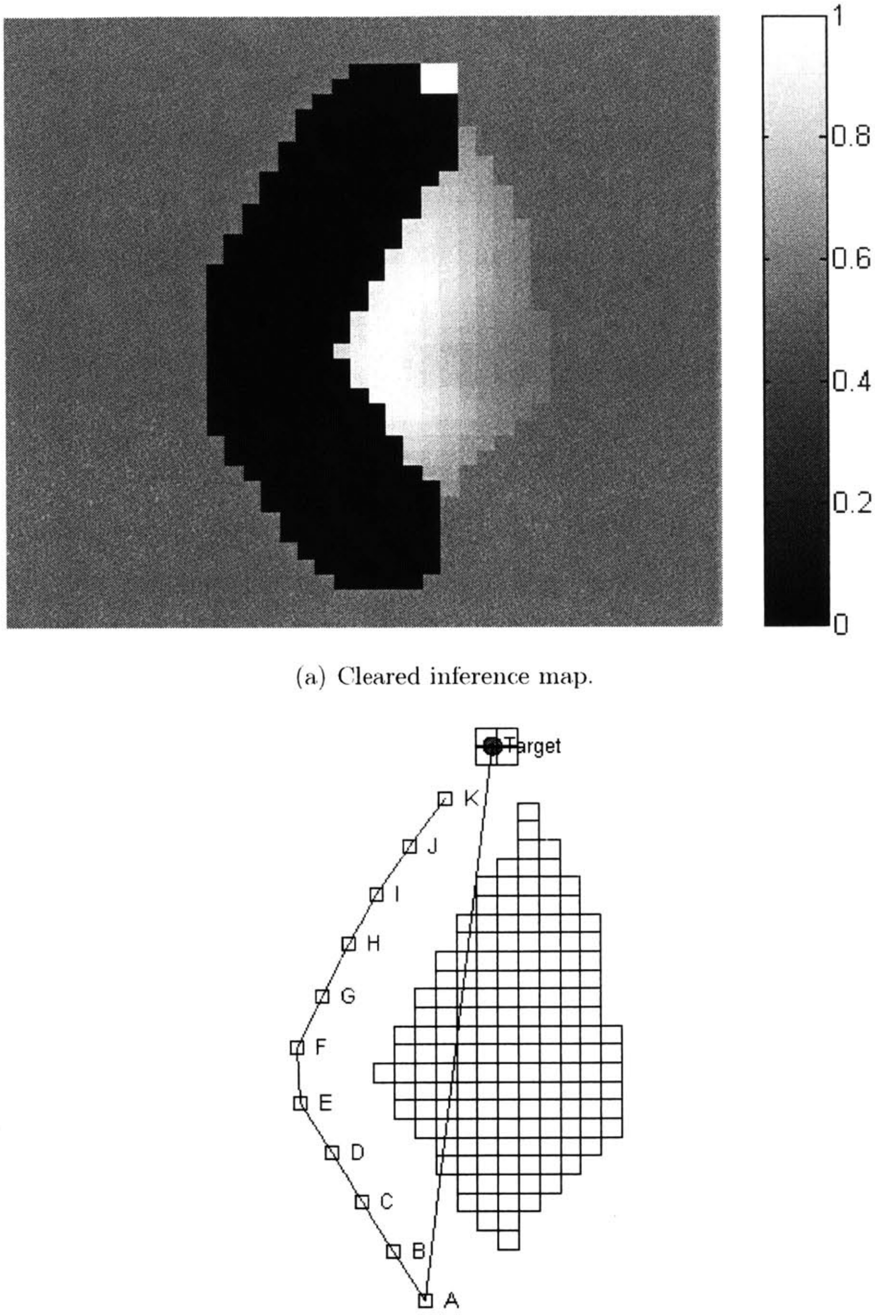
The obstacle inference map can now be updated with the hypothetical obstacle map by combing the two using Equation 4.13 and 4.14 and then using the combined map as the obstacle inference map for each remaining path segment of an observed path. In addition, subsequent observations made of additional contacts can also be incorporated into the obstacle inference map in the same manner. Using this method, the obstacle inference map can be updated for an observed contact path with multiple intent blocks.

4.8 Map Adjustment

After the obstacle inference map has been updated with the hypothetical obstacle maps for a given path, the probabilities in the inference map are then adjusted to reflect the belief that the immediate area surrounding the observed path of the contact does not contain any obstacles. This is a reasonable assumption since the contact would not be able to move directly through an area containing an obstacle nor would it desire to maneuver within a certain distance of an obstacle (in order to maintain a safety margin). From the assumptions made in Section 4.2, the area around the contact's path that will be considered free of any obstacles is defined by the minimum turning radius of the contact (which is assumed to be known). The cells in the obstacle inference map representing this area are then cleared of any hypothesized obstacles by decreasing the occupancy probability to zero. For example, using the map provided in Figure 4-9(b) as the obstacle inference map and the path of the contact illustrated in Figure 4-2(a), the adjusted map would contain the probability values illustrated in Figure 4-10(a). The width of the area cleared would be twice the minimum turning radius of the contact (to represent the safety margin on each side of the contact). Once a cell is cleared, its value will remain zero. The area representing the obstacle zone after the adjustment is indicated by the cell group illustrated in Figure 4-10(b).

4.9 Summary

This chapter presented a method for inferring the presence of obstacles in an environment by analyzing the movements of one or more contacts as they travelled through the observation area. Each path is analyzed to extract information concerning the contact's intent at various points along its path. The intents are used to segment the path and extract the possible obstacle zones from each path segment using its shape and intent. The identified obstacle zones are used to create a hypothetical obstacle map for each path segment. Two approximation methods were presented to accomplish this task since an exact approach is intractable. The hypothetical obstacle maps



(a) Cleared inference map.

(b) Adjusted obstacle zone.

Figure 4-10: Clearing the obstacle map of the contact's path.

are used to update the primary obstacle inference map by means of a map-combining formula presented by Moravec [12]. The obstacle inference map was also cleared of any obstacles in a reasonable area surrounding the path of the contact to reflect the safety margins with which ships generally navigate with. The process used to update the obstacle inference map with a single path can be used to incorporate any number of observed paths into the map. The result is a combined probability map of obstacle locations in the environment created from multiple sources of information.

[This page intentionally left blank]

Chapter 5

Intent Estimation Results

This chapter presents the results of testing the intent estimator developed in Chapter 2, and improved in Chapter 3. The first example simulation illustrates the general capability of the intent estimator. The demonstration of the estimator will then be tested through versions of the estimator and changes to the environment. Afterwards, the intent estimator will be evaluated using various user-defined run-time parameters.

5.1 Simulation Configuration

This section describes the elements used to configure the simulations presented in this chapter. These include the configuration of the observed contact and targets in the environment, the derived intent state space, and different vessel types used to model the contact.

5.1.1 Contact and Target Configurations

The configurations for the contact and targets in the environment are provided in Table 5.1 and illustrated in Figure 5-1. There are six targets in the environment, four objects (stationary targets) and two vessels (moving targets). A stationary target is labeled with its number and prefixed with the letter 'O' to identify the target as an object. A moving target is prefixed with the letter 'V'. Each target is identified

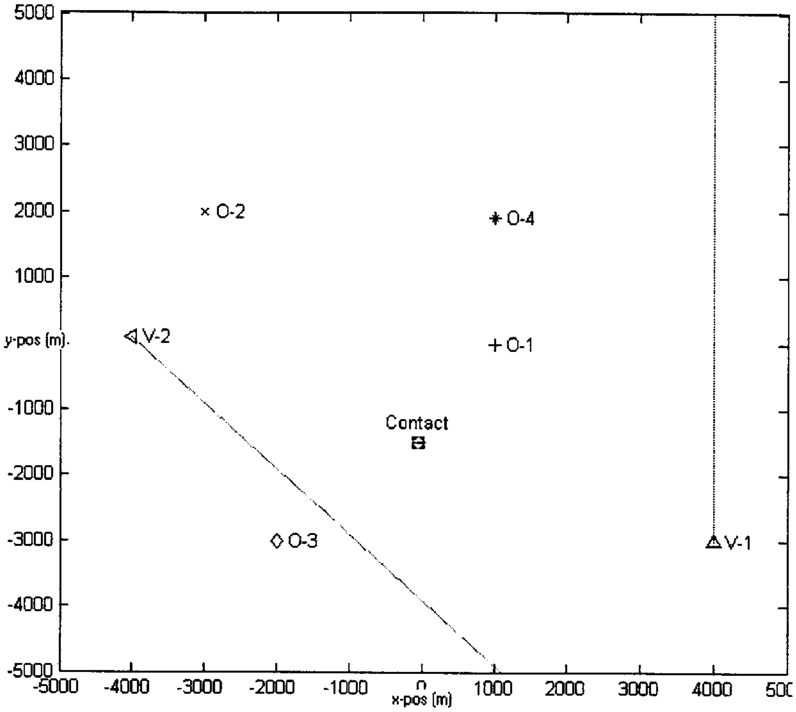


Figure 5-1: Simulation setup.

	Contact	Obj. 1	Obj. 2	Obj. 3	Obj. 4	Vessel 1	Vessel 2
Label	—	O-1	O-2	O-3	O-4	V-1	V-2
Map Symbol	□	+	×	◇	✱	△	◁
x-position (m)	−50	1000	−3000	−2000	1000	4000	−4000
y-position (m)	−1500	0	2000	−3000	1900	−3000	100
Speed (mps)	—	—	—	—	—	5	15
Heading (deg)	90	—	—	—	—	0	135

Table 5.1: Initial contact and target configurations.

$\mathbf{t}_i$	π_i	Belief Symbol	Target Map Symbol
Approach(O-1)	1/8	+	+
Approach(O-2)	1/8	×	×
Approach(O-3)	1/8	◇	◇
Approach(O-4)	1/8	✳	✳
Follow(V-1)	1/8	△	△
Intercept(V-1)	1/8	☆	△
Follow(V-2)	1/8	◁	◁
Intercept(V-2)	1/8	▷	◁

Table 5.2: Intent state space with initial probabilities and symbols used.

in Figure 5-1 by its label and symbol. The lines extending from the moving targets (V-1 and V-2) are the projected straight-line path of the targets. The initial position of the contact is identified in the figure. The observation area is a 10 km × 10 km environment.

5.1.2 Intent State Space

Using the intent definition process described in Section 2.3, eight intents are created to represent the contact's intent state space. These intents are listed in Table 5.2 along with the initial belief (π_i) that the contact has that intent and the symbol that will be used to represent each intent in the belief graphs. Most of the intent symbols correspond to their respective target symbols. This is the case when one intent is created for each target. In this thesis, such a situation occurs for stationary targets. For each moving target, two intents are generated. To distinguish between the two intents, a separate symbol is used for the second intent, which is provided in Table 5.2.

5.1.3 Contact Vessel Type

The contact's vessel type is represented by the vessel type variable ($\mathcal{V}$). The type of the contact reflects its maneuvering capabilities. The contact's actual maneuvers are analyzed by the intent estimator by comparing them to the predicted models of how the contact would have maneuvered given its modeled capabilities. The vessel type of

	Cargo Ship	Cruiser
Minimum Speed (mps)	2.6	2.6
Maximum Speed (mps)	10.3	15.4
Normal Speed (mps)	7.2	7.7
Minimum Turning Radius (m)	420	500
Probability	0.2	0.8

Table 5.3: Different possible contact vessel type modeled properties and values.

the contact in all simulations is the type identified as a cruiser. This information is not provided to the intent estimator. Since the contact's type is unknown (to the intent estimator), a number of vessel types are considered for providing an estimate of the contact's capabilities (as described in Section 3.2.2). In these simulations, the state space of $\mathcal{V}$ was created to reflect a strong belief that the contact being observed is a cruiser, while allowing for the possibility that it might be some other type of large ship with slightly different capabilities (a cargo ship). A cruiser is simply defined in this thesis as vessel with a faster speed capability and larger minimum turn radius than a cargo ship. The capabilities of each vessel type and a contact's *a priori* probability of being that type of vessel are provided in Table 5.3.

5.1.4 Run-Time Parameters

The user-defined parameters for the intent estimator include 1) the elapsed time between updates (τ), and 2) the parameters used in the von Mises and normal distributions for creating the Observed Intent probability distribution function, OI from Section 2.6.2, (k, σ_x^2, and σ_y^2). The values used for these parameters are provided in Table 5.4 and are held constant for most of the simulations in this chapter. The effects of running the intent estimator with different values for these run-time parameters are addressed Section 5.6. The value for τ was arbitrarily selected to reflect a reasonable amount of time that would pass between two observations of a contact. The elapsed time should be long enough so that the contact would have enough time to make a noticeable movement (from its last observation) towards a particular target. However, as τ increases, the intent observation models will become less and less

Parameter	value
τ (sec)	60
k	5
σ_x^2 (m)	200^2
σ_y^2 (m)	200^2

Table 5.4: Run-time parameters.

accurate since there will be more time for the contact to deviate from an ideal path. Such deviations may occur as a result of factors such as currents, winds, and navigation errors. A balance between these two concerns was achieved by setting τ at 60 seconds. This was perceived to be enough time for the contact to make a noticeable movement, but not so long that the contact's course would deviate dramatically.

The parameters used to create the OI (k, σ_x^2, and σ_y^2) reflect the expectation that the contact's path will match the predicted path modeled by the intent estimator. The values for σ_x^2 and σ_y^2 are used to compare the contact's x and y position to those predicted by the model. The values for σ_x^2 and σ_y^2 are the same since there is no bias that the contact is more likely to move along one axis versus the other. These values have been set to reflect a standard deviation (σ) of 200 m. This indicates that the contact's position (along each axis) will be within ±200 m of the predicted position about 70% of the time.

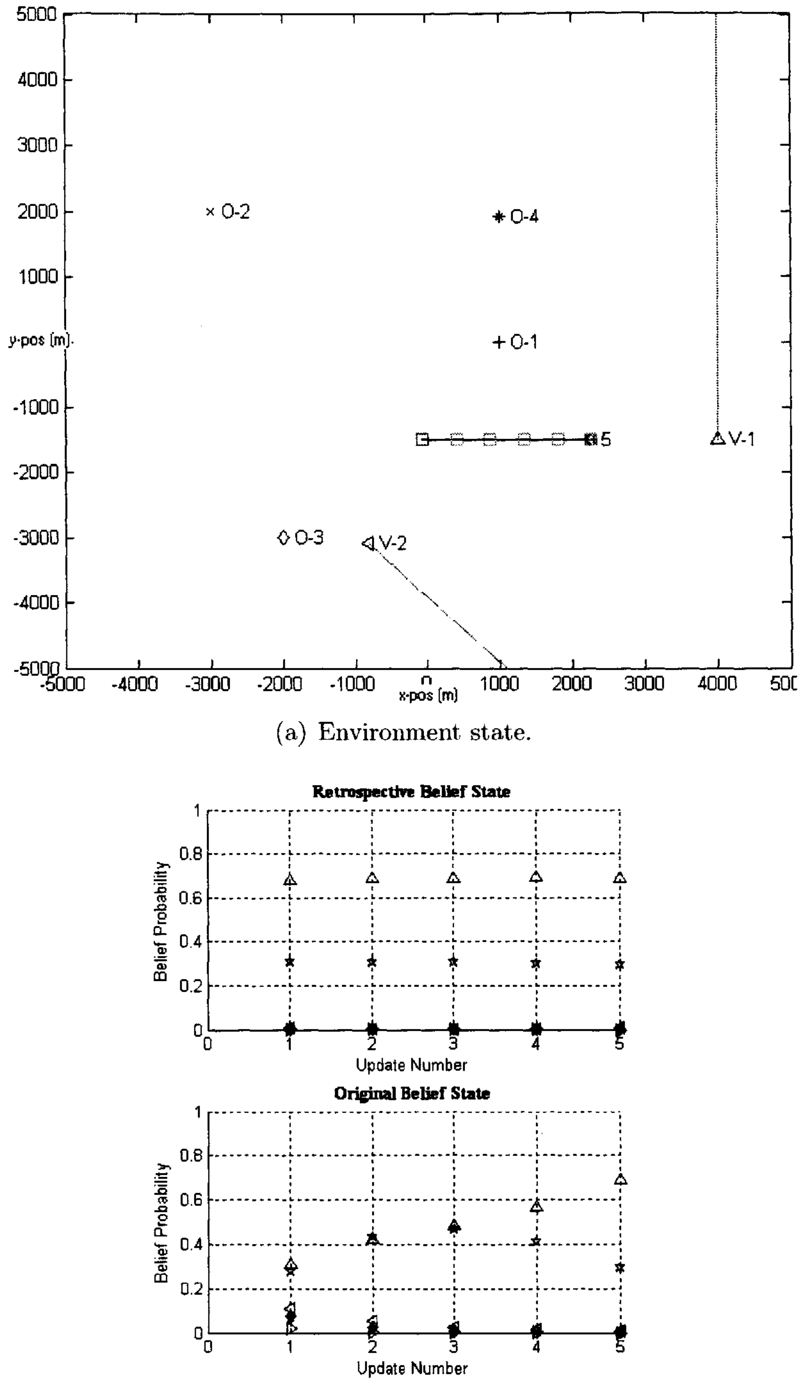
The value of k perform a similar role as σ_x^2 and σ_y^2 but is used by the von Mises distribution. It is used to indicate how likely the contact's observed heading will match the predicted heading. The larger the value for k (where $k > 0$), the higher the belief that the contact's movement will reflect the predicted model (Figure 2-5). The value of 5 was assigned to k (Table 5.4). This value reflects a belief the contact's observed heading will fall within 30° of the predicted heading about 70% of the time.

5.2 Primary Example Simulation

This section presents a simulation using the environment illustrated in Figure 5-1, and the configuration provided in Section 5.1. The contact was steered by the user to reflect a variety of intents using different speeds and rudder amounts. The simulation ran for 28 minutes—with updates at every minute ($\tau = 60$ sec). Figures 5-2, 5-3, and 5-4 shows the simulation environment and belief state plots at the 5, 15, and 28 minute marks respectively. Each intent belief state figure provides both the belief state calculated without the forward-backward procedure (labeled "original belief state") and the intent belief state calculated using the procedure (labeled as the "retrospective belief state"). The intent belief state values shown in the retrospective belief state plot are recalculated at each update. This is necessary because the observation sequence is modified (added to) at each update. Each intent is represented by its symbol from Table 5.2. Since the elapsed time for each update (τ) was set at 60 seconds (1 minute), each update (u) corresponds with the elapsed time (in minutes) of the simulation. Therefore, the update number and simulation time (at integer intervals) will be used interchangeably. For example, the state of the simulation shown in Figure 5-2 can be referred to as either the state "at update 5" or "after 5 minutes."

For the first 5 minutes of the simulation, the contact moves at a medium speed in an easterly direction (Figure 5-2(a)). Each observation of the contact is identified by its symbol from Table 5.1. Every fifth observation (update) is labeled with the update number and a ✡ symbol. Each observation corresponds to its respective update in the belief state plots (Figure 5-2(b)). A line connecting these observations will be used to represent the contact's path. Using the plot of the original belief state (Figure 5-2(b)), it can be seen that the intent estimator initially builds the belief that the contact is trying to follow V-1 since its movements would not place it on a intercept course with that target.

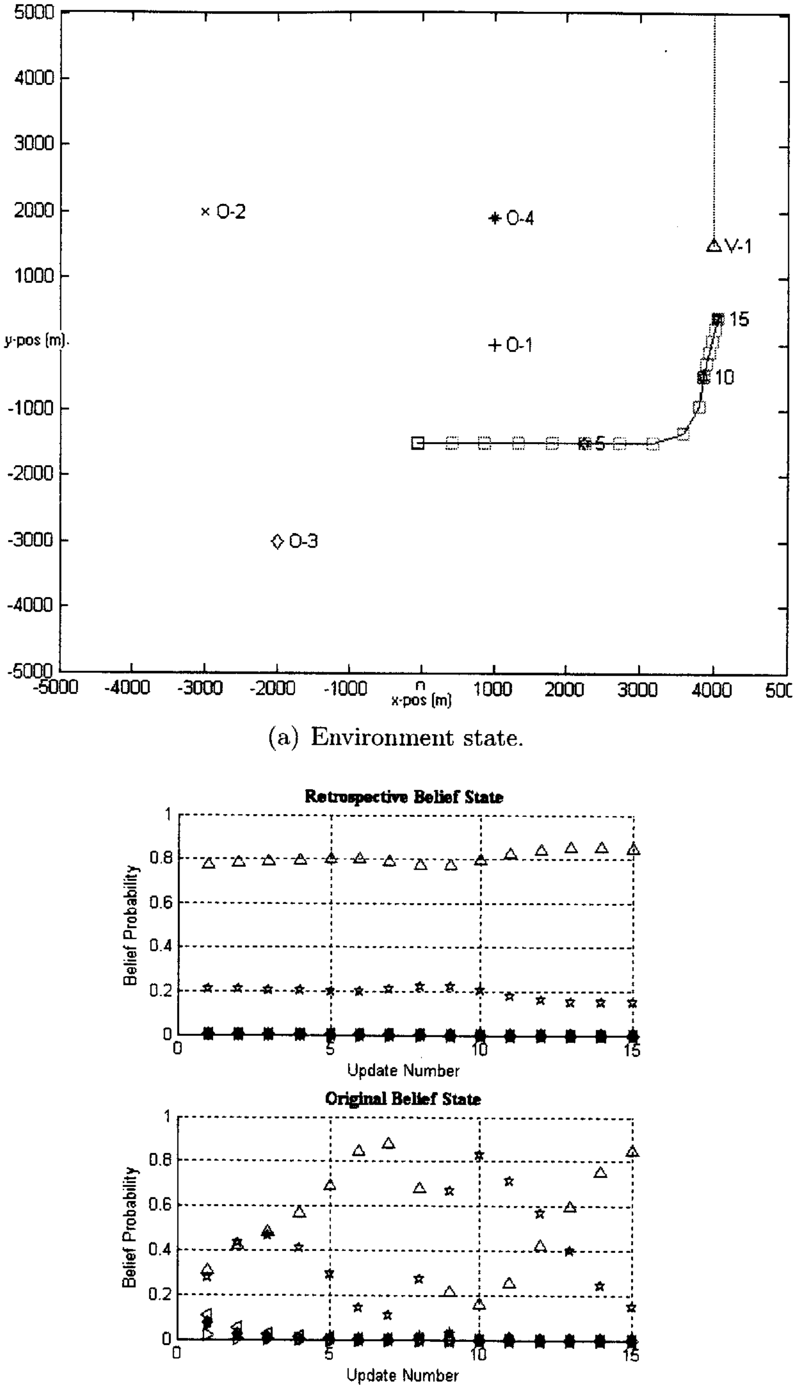
For the next 10 minutes (Figure 5-3), the contact moves along this course and speed for 2 minutes before turning towards V-1 and then slowing down. The decrease in the vessel's speed is evident by the close spacing of the observations. By this time, V-2 has already moved out of the observation area. As the contact starts turning toward V-1 after update 7, the intent estimator starts believing that the contact is



(a) Environment state.

(b) Intent belief state.

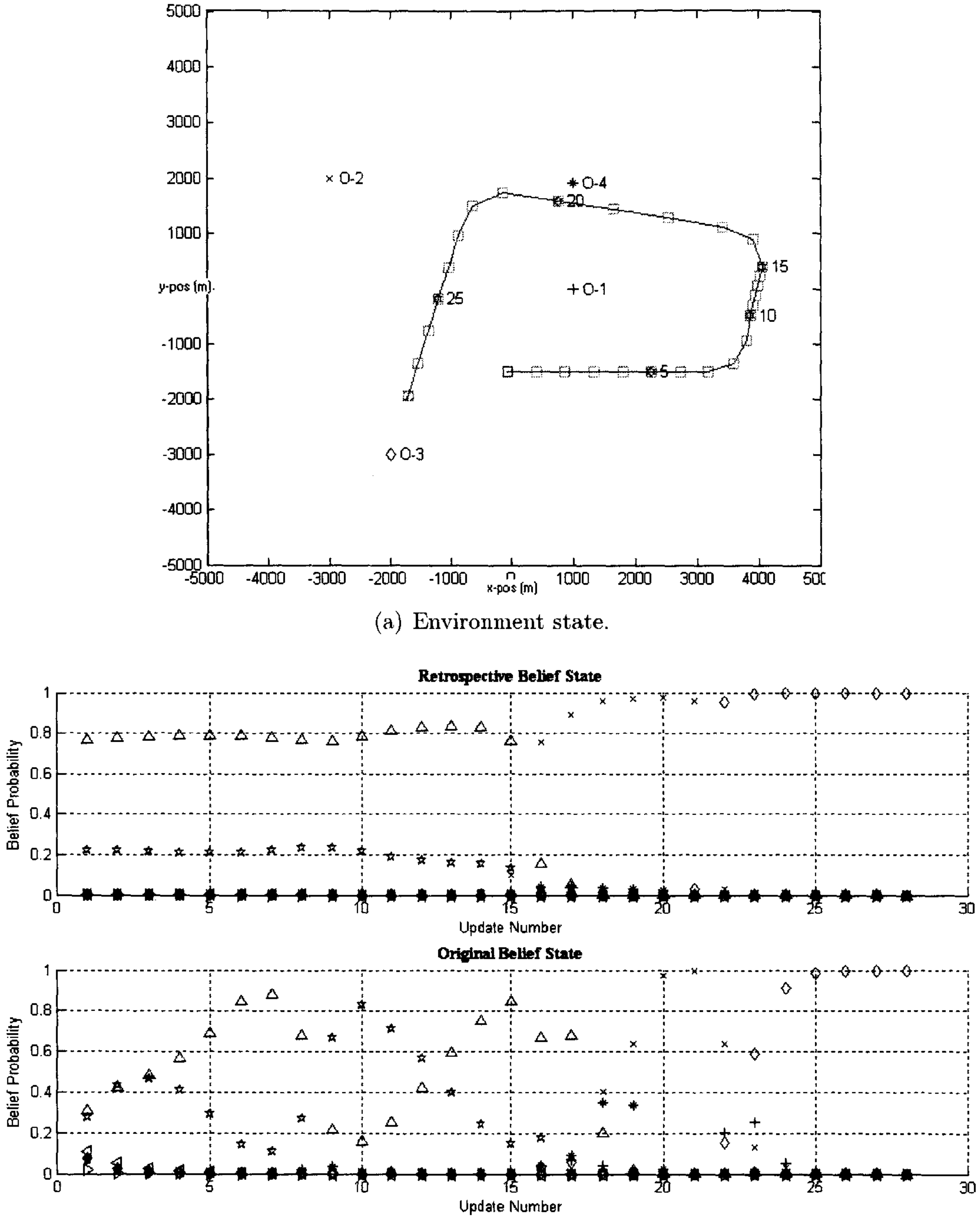
Figure 5-2: Primary simulation after 5 minutes.



(a) Environment state.

(b) Intent belief state.

Figure 5-3: Primary simulation after 15 minutes.



(a) Environment state.

(b) Intent belief state.

Figure 5-4: Primary simulation after 28 minutes.

actually trying to intercept the target. At update 10, the contact slows down and the belief that the contact is actually trying to follow V-1 goes back up again (up to update 15). This occurs because the intent estimator does not believe that the contact would be moving that slowly if it wanted to intercept the target. From analyzing the original intent belief state for updates 1–15, it can be concluded that the intent of the contact during that time period was to follow V-1, though it might have had a slight intention to intercept it. This conclusion is reflected in the retrospective belief state in Figure 5-3(b).

For the next 13 minutes, the contact then increases its speed and heads west for 6 minutes before slowing down and changing to a southerly heading. The entire path of the contact is illustrated in Figure 5-4(a). After update 15, the contact turns towards O-2, moving past but fairly close to O-4. The contact turns to approach O-3 after update 21. The intent belief of the contact during this time is clearly illustrated by the calculations for the retrospective belief state as shown in Figure 5-4(b).

The intent belief state at various points during the simulation is provided in Figures 5-2(b), 5-3(b), and 5-4(b). When comparing these three figures, it is observed that the belief state values for each update in the original belief state plot do not change as more observations are made of the contact. However, the value for each update in the retrospective belief state does change. This is due to recalculating the belief state for every update with each additional observation (to take into account the entire observation sequence when determining each belief state). The original method only uses the previous and current observations for calculating the belief state for each update. As can be seen, the belief state calculated using the forward-backward procedure provides a relatively consistent (when compared to the original belief state) values for the predominant intent for the duration that the intent remained predominant. Once the intent estimator detects an intent change, the change is reflected immediately at the update when the change was determined to have taken place. The original belief state illustrates a very inconsistent belief of the contact's predominant intent throughout the observation period and a distinction of when the contact changed intent is not as evident as from the retrospective belief state plot. For example, after update 15, the retrospective belief state reflected the change in the contact's intent immediately while it took the original belief state two more updates to begin to suspect the new intent. It then took another two updates before the intent was fully realized (reflected by a high probability). The initial delay resulted from having to decrease the high probability value from the previous predominant intent using a single (current) observation. The improved method (which is post-processed) incorporates the future observations in adjusting that update thus creating a quicker transition in the belief state. The second delay was due to the presence of O-4 which caused the *Approach*(O-4) intent to slightly increase since it was along the contact's path for reaching O-2. The calculation used for the retrospective belief state was able to filter this out.

The retrospective belief state (Figure 5-4(b)) illustrates the capability of the intent estimation algorithm to determine a belief state for representing the intent of a contact by observing its movements. The configuration of the environment and the contact's observed path used in this simulation will be referred to as the original scenario. The results of the simulation presented here will be used as a point of comparison for the rest of the simulations in this chapter. These comparisons will illustrate how the various parts of the intent estimation algorithm affect the belief state calculations and how robust the estimator is.

5.3 Multiple Contact Paths Considerations

One of the improvements made in Chapter 3 to the intent estimation model is the incorporation of multiple possible contact paths for creating each intent model. For each intent, nine possible paths were created, each using an unique speed/turn radius combination ($\mathcal{SR}$—Section 3.2.1). The result was an intent model that was capable of representing the different speeds and turn radii that a contact could use to accomplish a given intent.

The benefit that comes from considering multiple contact paths by the intent estimator is illustrated in Figure 5-4 during updates 16–20. During this block of time, the intent estimator correctly identified the predominant intent of the contact as *Approach*(O-2) instead of *Approach*(O-4). This recognition was significant since the path of the contact during that block of time could have been interpreted as the latter intent since it comes fairly close to O-4. The factor which led the intent estimator to conclude that *Approach*(O-2) was the predominant intent resulted from considering the speed of the contact (which was considered when the intent models were created). During this block of time, the contact was considered be relatively far (d_F) from O-2. However, the contact was considered to be at an intermediate distance (d_M) from O-4 at first but then moves into the closer zone (d_C) around updates 18–19. From the modeled distribution of how a contact would maneuver given the distance and bearing to the target (provided by Table 3.3), the intent estimator would attribute the faster speed with which the contact was moving to the pursuit of the target which was in the further zone (O-2).

5.3.1 Single $\mathcal{SR}$ Simulation

Figure 5-5 provides the intent belief states produced by running the original scenario without the multiple path consideration for the intent estimator. From the retrospective belief state graph, it can be seen that the intent estimator had trouble determining the predominant intent of the contact during updates 16–20. During this block of time, the symbols representing *Approach*(O-2) and *Approach*(O-4) are located right next to each other. This reflects the difficultly that the estimator experiences when trying to distinguish which intent the contact had since the observed path could have been used to accomplish either intent if only a single $\mathcal{SR}$ combination is considered. Therefore, the incorporation of the multiple contact model had a noticeable impact on the intent estimator. This can be seen by comparing the results from running the estimator without the consideration of multiple contact paths (Figure 5-5) to the results from incorporating the multiple contact paths into the intent

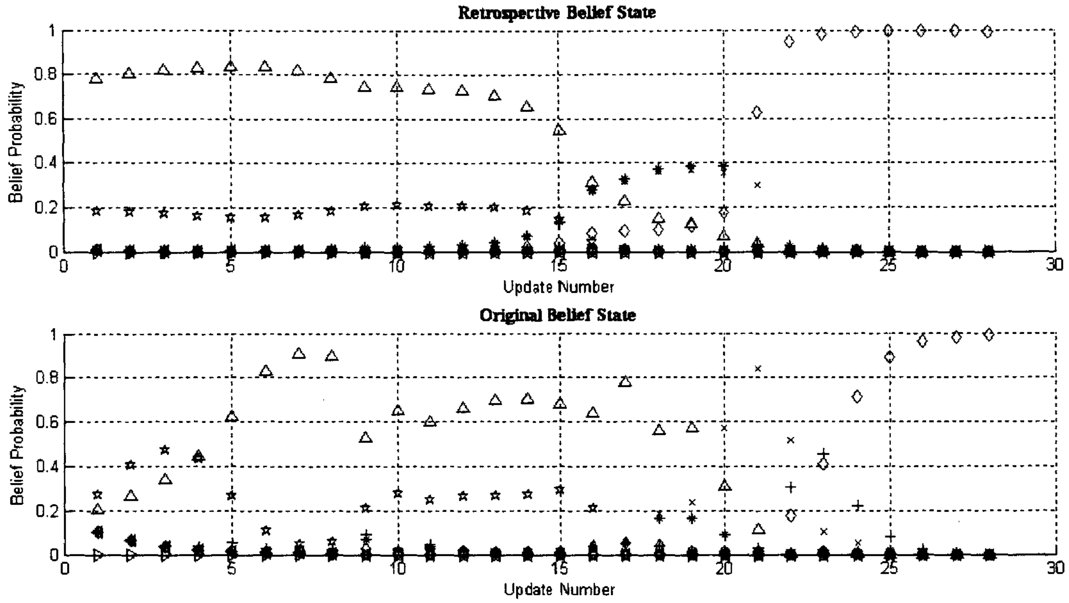
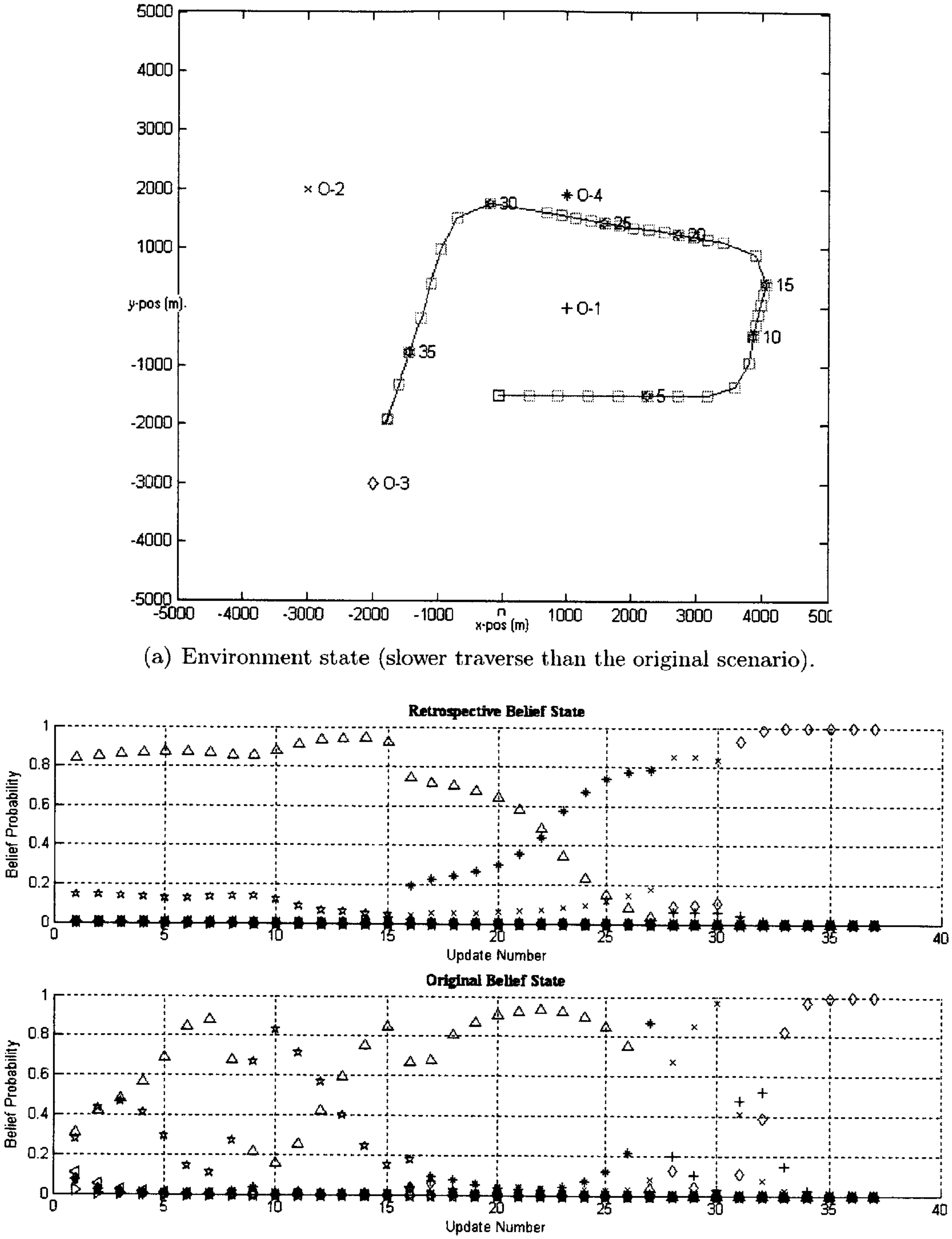


Figure 5-5: Intent belief states for 1 $\mathcal{SR}$ capability.

models (Figure 5-4(b)).

5.3.2 Multiple Path Model Simulation

In the simulation run presented in Figure 5-6, the contact was driven to approach O-4 (instead of O-2) by slowing down its approach speed after it makes the turn to head in the original direction of O-2. The decrease in speed is illustrated by the closeness of the contact's observed positions during this time (updates 17–29). From the retrospective belief state plot, it can be seen that the intent estimator correctly attributes the decrease in speed to the intent of approaching O-4. The belief value for the *Approach*(O-4) intent does not increase dramatically since there is the probability that the contact may still be following V-1. This is because the *follow* intent model predicts that the contact would move in the proximity of the target vessel. Since V-1 was still fairly close to the contact at this point, the intent estimator still considered the *Follow*(V-1) intent to be a reasonable pursuit of the contact. By update 23, the *Approach*(O-4) intent dominated. At update 28, the contact moves past O-4 but



(a) Environment state (slower traverse than the original scenario).

(b) Intent belief state.

Figure 5-6: *Approach*(O-4) simulation.

$\mathbf{i}_i$	$P(\mathcal{I}_8 = \mathbf{i}_i)$	$P(\mathcal{I}_9 = \mathbf{i}_i)$
Approach(O-1)	0.00577	0.00652
Approach(O-2)	0.01269	0.01847
Approach(O-3)	0.00004	0.00303
Approach(O-4)	0.01297	0.01889
Follow(V-1)	0.78317	0.74584
Intercept(V-1)	0.18526	0.20725
Follow(V-2)	0.00010	0
Intercept(V-2)	0	0

Table 5.5: Intent state space before and after the removal of V-2.

is still on a course that would bring it to O-2. At this point, the intent estimator shifts the predominant belief to the *Approach*(O-2) intent which continues until the contact changes course and heads toward O-3 (at update 31). The simulation provides another example where the contact's speed was a significant factor in determining a contact's intent.

5.4 Dynamic Intent State Space

The ability of the intent estimator to handle a changing intent state space can be demonstrated by adding and removing targets from the observation environment. Removal of a target was shown in the original example simulation presented in Section 5.2. The targets V-1 and V-2 were removed from the environment when they moved out of the observation area represented by the figure. V-1 moved out of view after update 8 and the retrospective intent belief state values for update 8 and 9 are provided in Table 5.5. By update 8, the *Intercept*(V-2) intent had already been considered invalid since the intent estimator was unable to generate a path for the contact to intercept V-2. After V-2 left the observation area after update 8, that target (V-2) became invalid and the remaining intent using V-2 as the target is effectively removed from the state space by assigning it a probability value of 0. Similarly, when V-1 moved out of view after update 24, the belief state also reflected the change and is shown in Table 5.6.

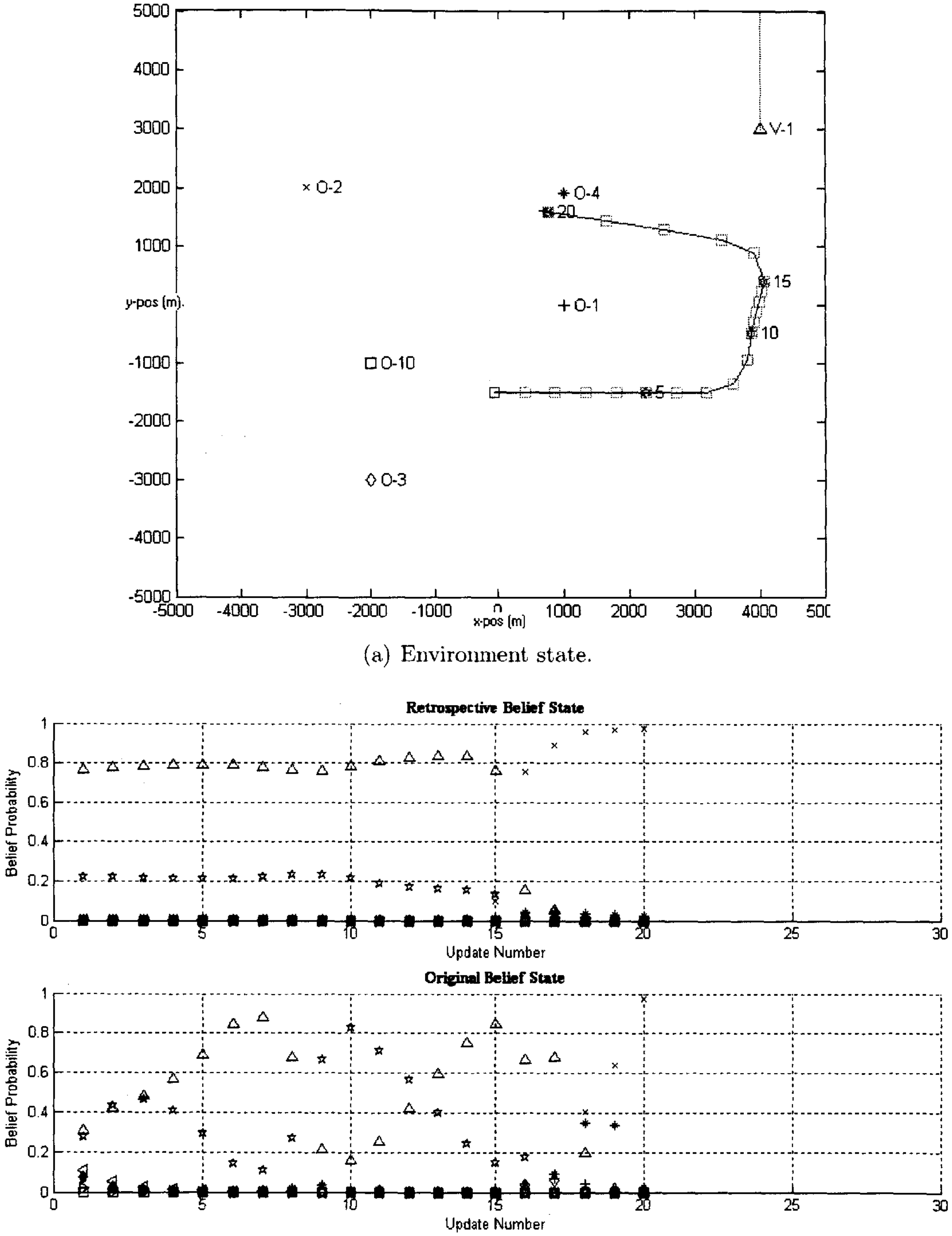
$\dot{\mathbf{t}}_i$	$P(\mathcal{I}_{24} = \dot{\mathbf{t}}_i)$	$P(\mathcal{I}_{25} = \dot{\mathbf{t}}_i)$
Approach(O-1)	0.00507	0.00281
Approach(O-2)	0.00125	0.00079
Approach(O-3)	0.99292	0.99616
Approach(O-4)	0.00028	0.00024
Follow(V-1)	0.00048	0
Intercept(V-1)	0	0
Follow(V-2)	0	0
Intercept(V-2)	0	0

Table 5.6: Intent state space before and after the removal of V-1.

$\dot{\mathbf{t}}_i$	$P(\mathcal{I}_{20} = \dot{\mathbf{t}}_i)$	$P(\mathcal{I}_{21} = \dot{\mathbf{t}}_i)$	$P(\mathcal{I}_{22} = \dot{\mathbf{t}}_i)$
Approach(O-1)	0.00019	0.00689	0.01154
Approach(O-2)	0.97642	0.96133	0.03236
Approach(O-3)	0.00043	0.03047	0.93289
Approach(O-4)	0.02292	0	0.00006
Follow(V-1)	0.00003	0.00018	0.00001
Intercept(V-1)	0.00001	0	0
Follow(V-2)	0	0	0
Intercept(V-2)	0	0	0
Approach(O-10)	0	0.00113	0.02314

Table 5.7: Intent state space before and after the addition of O-10.

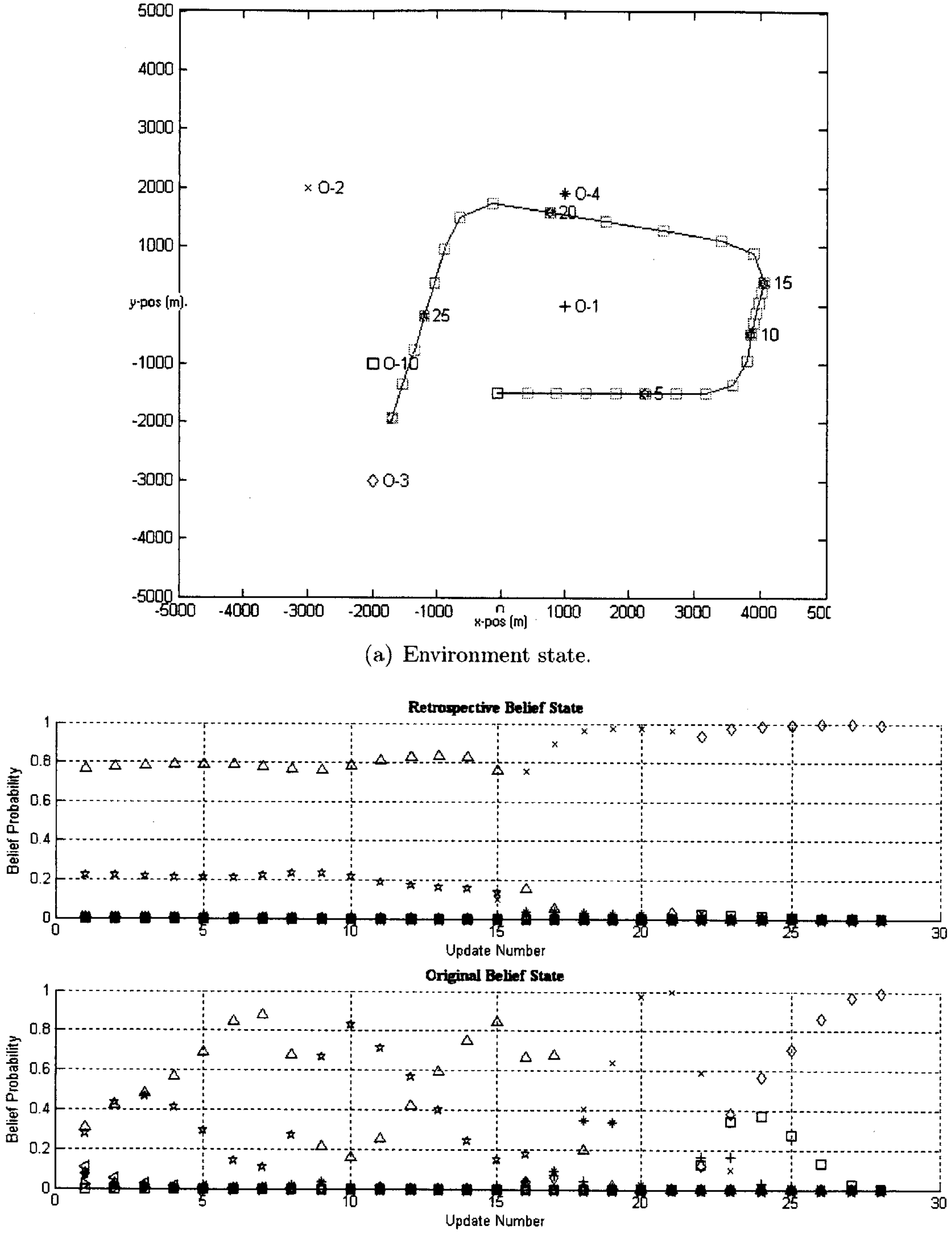
The simulation used to illustrate the ability of the intent estimator to accommodate an increase in the size of the state space is provided in Figures 5-7 and 5-8. This simulation replicates the original scenario until update 20. At this point, another target (O-10) was added to the environment and is shown in Figure 5-7(a). As a result, the intent to approach O-10 is added to the intent state space and represented by the symbol (□) in the belief state plots. The belief value for the *Approach*(O-10) intent at and prior to update 20 is 0 since the target was not valid during those times. The contact then completes the same path as in Section 5.2 and the resulting belief state plots are provided in Figure 5-8(b). The retrospective intent belief states for update 20, 21, and 22 are provided in Table 5.7. The *Approach*(O-10) intent row from this table indicates that the probability value for that belief was 0 at update 20 (before the target was introduced) and increased to a non-zero value once it was detected



(a) Environment state.

(b) Intent belief state.

Figure 5-7: Original scenario after 20 minutes.



(a) Environment state.

(b) Intent belief state.

Figure 5-8: Original scenario run with an added target.

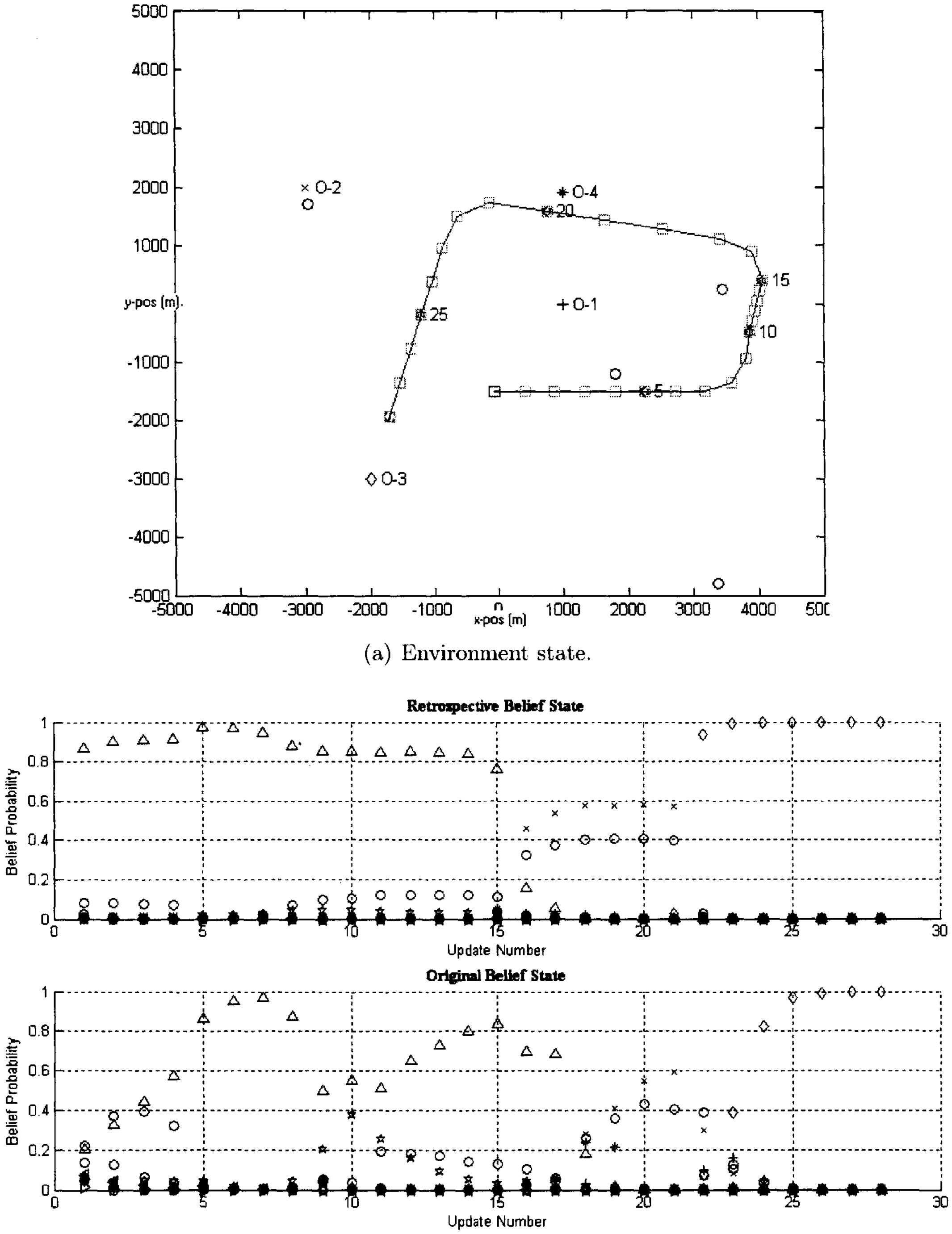
(after update 20). Since the position of O-10 is relatively close to the path that the contact took to approach O-3, it is difficult to determine, by the original belief state values, which target the contact was actually moving towards once it started its turn in the direction of the two targets (after update 21). This was illustrated in the original belief state plot from Figure 5-8(b) at updates 22 and 23 when the values for *Approach*(O-3) (◇) and *Approach*(O-10) (□) rose steadily together. Then as the contact got closer to O-10 without slowing down, the belief in the *Approach*(O-10) intent starts to decrease. This uncertainty of the contact's intent represented by the original belief state was removed by the retrospective belief state in Figure 5-8(b).

A few additional observations can be made based on Table 5.7. First, it can be seen that V-2 is no longer a valid target. This is due to V-2 departing the area as described above. In addition, the *Intercept*(V-1) intent is no longer valid after update 20. Since V-1 is still a valid target (which can be deduced from the non-zero probability of intent *Follow*(V-1)), the invalid *Intercept*(V-1) intent reflects the fact that the contact is no longer capable of intercepting that vessel (this assumes the intent estimator has accurately modeled the capabilities of the contact). Finally, the probability value for the *Approach*(O-4) intent becomes 0 at update 21 but returns to a non-zero value at update 22. This reflects a temporary removal of that intent from the state space because the contact is less than one minimum turning radius from the target (from the assumption made in Section 3.2.4.2). Since the contact was within 400 m of O-4 at update 20 (which is less than the turning radius of both vessel types modeled), the target was no longer considered valid and its associated intent was removed from the state space. Therefore, the update at 21 did not consider the *Approach*(O-4) intent. However, at update 21, the contact had moved past the target and the *Approach*(O-4) intent was reincorporated into the intent state space for update 22. As a result, the probability value for the *Approach*(O-4) intent was non-zero at update 22.

5.5 Target Domain Size

The following simulations run the same contact path from the original scenario with an increasing number of available targets. The purpose of these experiments was to determine how well the intent estimator performs when the intent state space increases. A total of three simulations were performed with 10, 20, and 30 targets. The targets were created using the targets from the original scenario and adding a number of random static targets (obstacles) to bring the total up to the desired number. Each randomly generated target will be represented in the environment state figure and belief state plot by the symbol '○'. The randomly generated targets are not assigned a unique symbol since the focus of the experiments is to determine how the new belief state deviates from the original belief state. Figure 5-9 shows the results from a simulation with 10 targets. Figure 5-9(a) provides a view of the environment after the contact completed its path. The resulting belief state plots are shown in Figure 5-9(b). From the retrospective belief state plot, it can be seen that the intent estimator produced similar result as in the original scenario with the exception that the belief for the *Approach*(O-2) (updates 16–21) was not as high. This can be explained by the presence of a random target fairly close to O-2, which the intent estimator considered as a likely intent given the contact's path during updates 16–21.

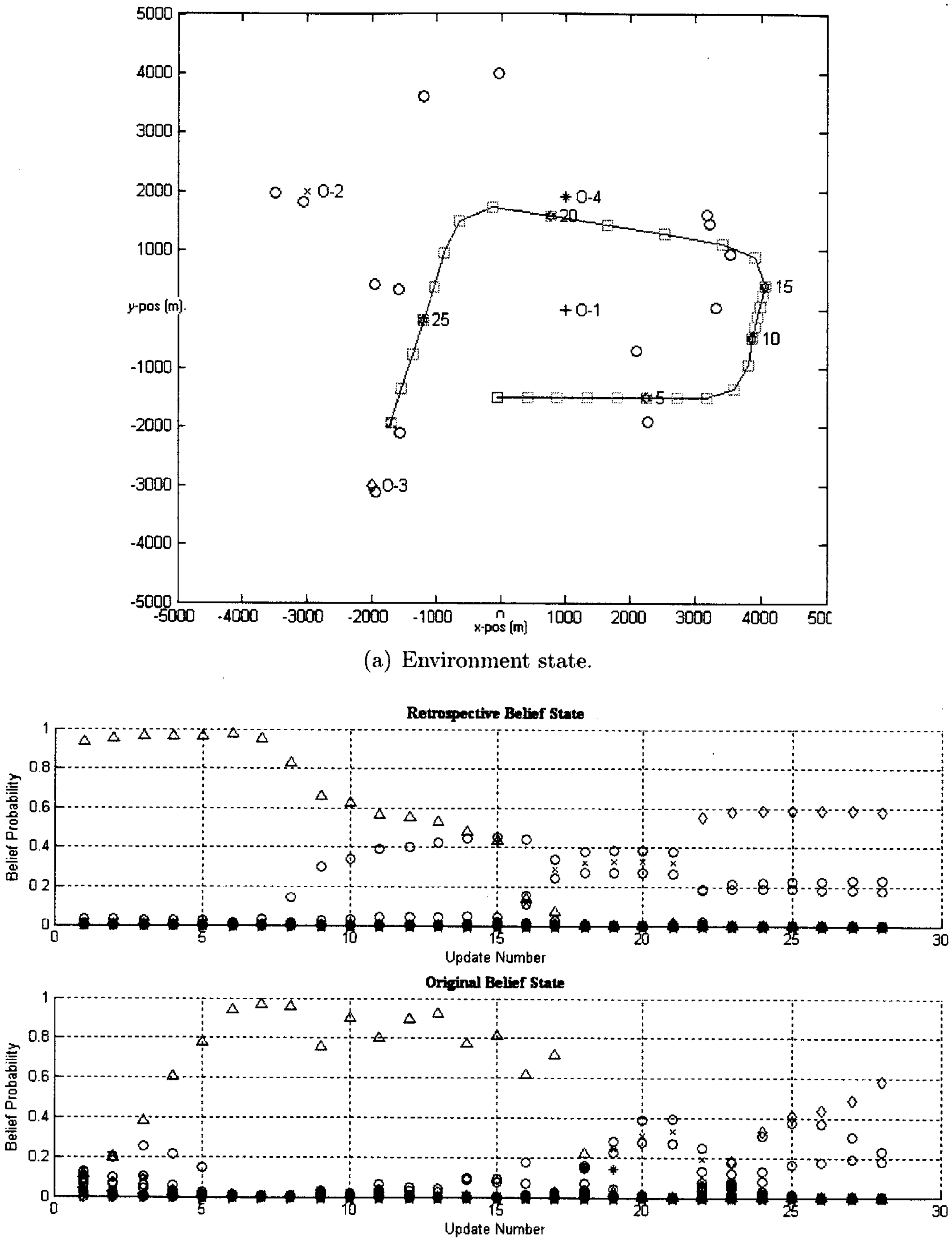
Figures 5-10 and 5-11 present the results from running the original scenario with 20 and 30 targets respectively. Both "environment state" figures show the view of the environment with the completed path and the random targets. From Figure 5-10(b), it can be seen that the additional targets had a larger effect on the calculations of the intent estimator than in the simulation with only 10 targets. However, in Figure 5-11(b), it can be seen that the intent estimator performed better than the simulation run with 20 targets and almost as well as the simulation run with 10 targets. This is due to the fact that the random placement of the targets in the simulation with 30 targets were not at locations where its associated intent would have been assigned a high belief value (along with the actual intent) given the maneuvers of



(a) Environment state.

(b) Intent belief state.

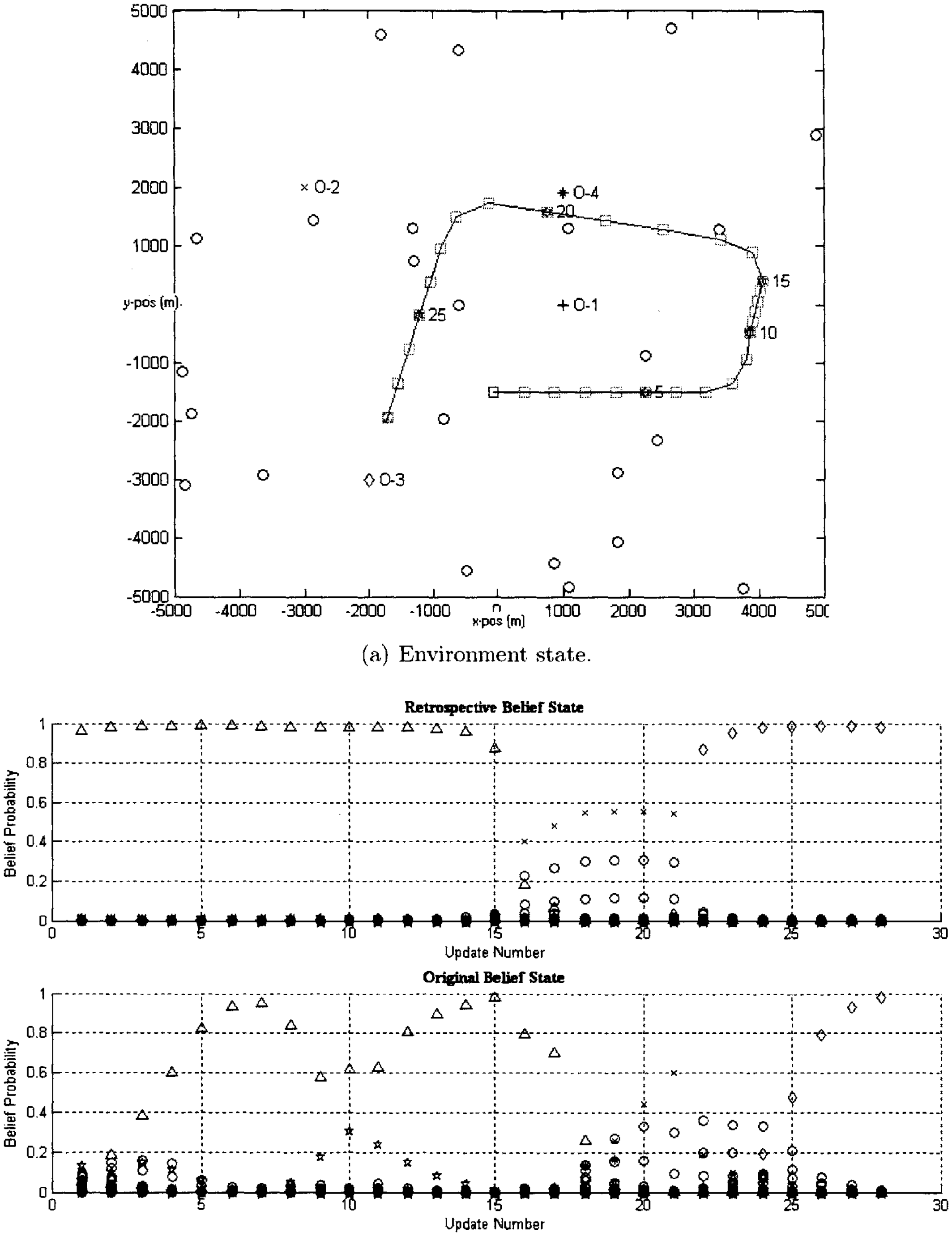
Figure 5-9: Simulation run with 10 targets.



(a) Environment state.

(b) Intent belief state.

Figure 5-10: Simulation run with 20 targets.



(a) Environment state.

(b) Intent belief state.

Figure 5-11: Simulation run with 30 targets.

the contact. From these results, it should be noted that the placement of the targets in an environment would have a greater impact on the ability of the intent estimator to provide and clear predominant intent than the actual number of targets in the environment. Therefore, the location of the targets in an environment should be considered when implementing and then using the results from the intent estimator.

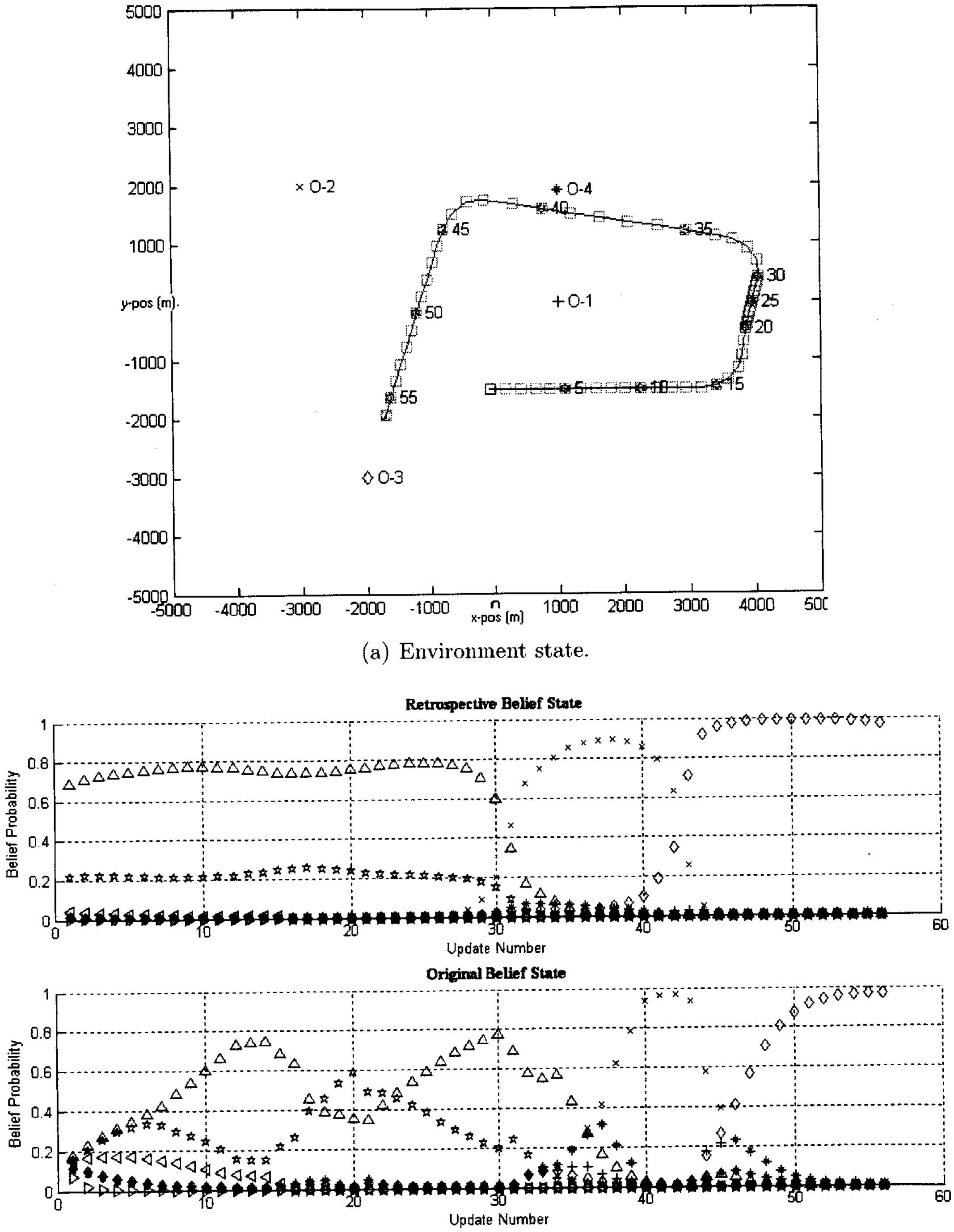
5.6 Parameter Modification Simulations

This section considers the effects of using various values for the run-time parameters provided in Table 5.4. Two pairs of experiments are reported here to illustrate the effects of modifying the elapsed time between updates and the parameters used to create the Observed Intent probability distribution function (OI). The first pair of simulations examine the effects of changing the elapsed time (τ) between updates—one will be with a smaller τ than the one originally used, and the other with a larger τ. The second pair of simulations will examine the effects of modifying k, σ_x^2, and σ_y^2 (the parameters used to create the OI). One will create a more restrictive OI while the other will be more relaxed.

5.6.1 Using Different Elapsed Times (τ)

The effects of changing the rate at which an observation is made of the contact to be used for updating the intent belief state are presented in this section. Two different values for τ are used, one that is lower than the current value used ($\tau = 60$) and one that is higher.

Figure 5-12 shows the results of running the simulation from Section 5.2 with $\tau = 30$ seconds. This value is half the original $\tau = 60$ seconds used. Therefore, there are twice as many updates displayed by the figure. From Figure 5-12(b), it can be seen that the values for both belief states do not change as rapidly from update to update when compared to the belief states for the original scenario (Figure 5-4(b)). In addition, the belief values for the predominant intent at each update are



(a) Environment state.

(b) Intent belief state.

Figure 5-12: Simulation run with $\tau = 30$ sec.

generally not as high as before. As mentioned in Section 5.1.4, if the time between updates is fairly short, the intent estimator may have difficulty in distinguishing the contact's movement towards a particular target. This is because all the modeled paths generated by the intent estimator for each intent start at the contact's previous location (from Section 2.6.1). These paths will initially be similar (since all paths will be maneuvering from the same position and heading) but diverges as time increases. Because the changes in the contact's movement are not as dramatic from one update to another (when compared to the original results), the change in belief that the contact is pursuing one target over another is not as dramatic. For example, if τ was set at 1 second and the contact was modeled to travel at 5 *mps*, the maximum possible difference between two intent models is 10 m (when the two intents move in opposite directions). Calculating this difference along one axis (to simplify this explanation), the maximum difference in the probability value (using the parameters from Table 5.4 and the normal distribution) is

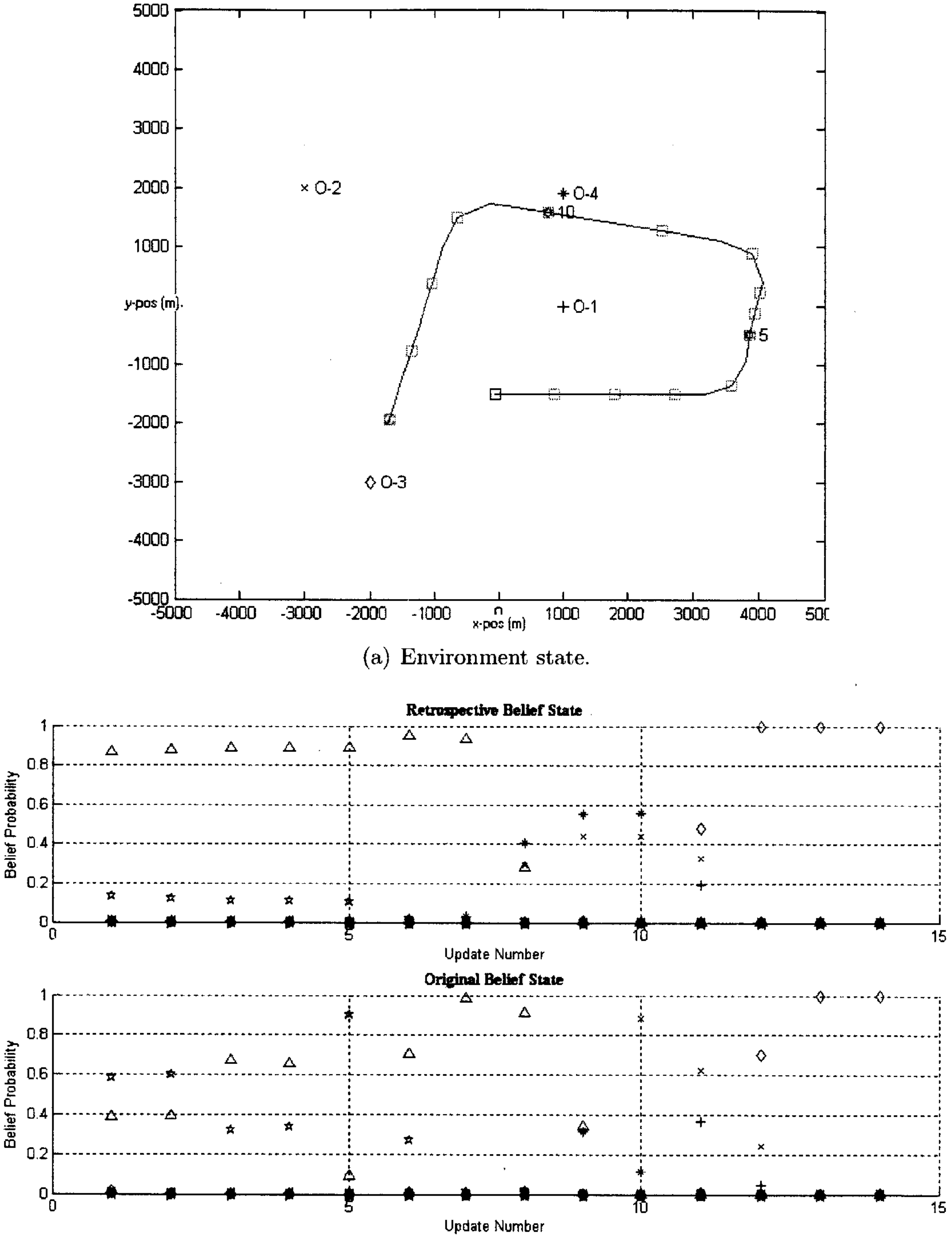
$$\frac{e^{-(0-0)^2/(2\sigma^2)}}{\sqrt{2\pi}\sigma} - \frac{e^{-(10-0)^2/(2\sigma^2)}}{\sqrt{2\pi}\sigma} \approx 2.4918 \times 10^{-6}.$$

Increasing the elapsed time to $\tau = 60$ seconds would result in a larger maximum distance of 600 m and a maximum difference in probability of

$$\frac{e^{-(0-0)^2/(2\sigma^2)}}{\sqrt{2\pi}\sigma} - \frac{e^{-(600-0)^2/(2\sigma^2)}}{\sqrt{2\pi}\sigma} \approx 1.9726 \times 10^{-3}.$$

The larger difference in probability, which resulted from the higher τ, would cause the belief state to change more dramatically from one update to the next, allowing an intent in the belief state to reach a higher belief value. However, the predominant intent for each update (from the retrospective belief state plot) still indicates that the contact's intent was *Follow*(V-1) for the first 15 minutes of the simulation, *Approach*(O-2) for the next 6 minutes, and *Approach*(O-3) for the final 7 minutes. The difference between the results using $\tau = 30$ seconds (Figure 5-12(b)) and $\tau = 60$ seconds (Figure 5-4(b)) is the maximum value of the predominant beliefs.

On the other hand, an increase in τ allows a larger change in the contact's movement from one update to another. Figure 5-13 shows the results from running the original scenario with $\tau = 120$. As can be seen, the inter-update changes in the belief states are larger and the belief value for the predominant intent at each update is generally higher than in the original belief state plot. However, a problem appears in the update range 8–11 (which corresponds to updates 16–22 in the primary simulation). During this time, the intent estimator had difficulty correctly determining the intent of the contact (which was *Approach*(O-2)). During updates 8 and 9, the belief values for *Approach*(O-2) and *Approach*(O-4) were close together (seen at update 10 in the original belief state). At update 10, the belief value for *Approach*(O-2) significantly increases above the value for *Approach*(O-4). However, the retrospective belief state would require more updates thereafter to allow *Approach*(O-2) to emerge as the predominant intent. Since the contact's intent changed soon after (in terms of number of updates), there were not enough updates supporting the *Approach*(O-2) intent to allow the retrospective belief state to reflect it as the predominant intent during that time block. As a result, the single predominant value for the *Approach*(O-2) intent was smoothed out when determining the retrospective belief state. Even though there was no significantly predominant intent indicated by the retrospective belief state plot in updates 8–10, the belief value for *Approach*(O-4) was slightly higher than for *Approach*(O-2). Section 5.3 explained that the intent estimator considers the distance of each target while calculating the intent belief state (through the intent models). Therefore, it would seem that the belief value for *Approach*(O-2) should have been the significantly higher one. However, since the elapsed time between update was set relatively high, the target O-4 was only considered to be in the contact's close zone for only 1 update (update 10). Therefore, only one update was made which was able distinguish *Approach*(O-2) as the dominant intent from the contact's speed. Therefore, even though a large τ may produce a larger change in the retrospective intent belief state, it may filter out intents with shorter time durations or miss the opportunity to extract certain information related to the contact's movement in the



(a) Environment state.

(b) Intent belief state.

Figure 5-13: Simulation run with $\tau = 120$ sec.

	k	σ_x^2	σ_y^2
Loose parameters	1	500^2	500^2
Normal parameters	5	200^2	200^2
Tight parameters	8	50^2	50^2

Table 5.8: Modified run-time parameters.

environment.

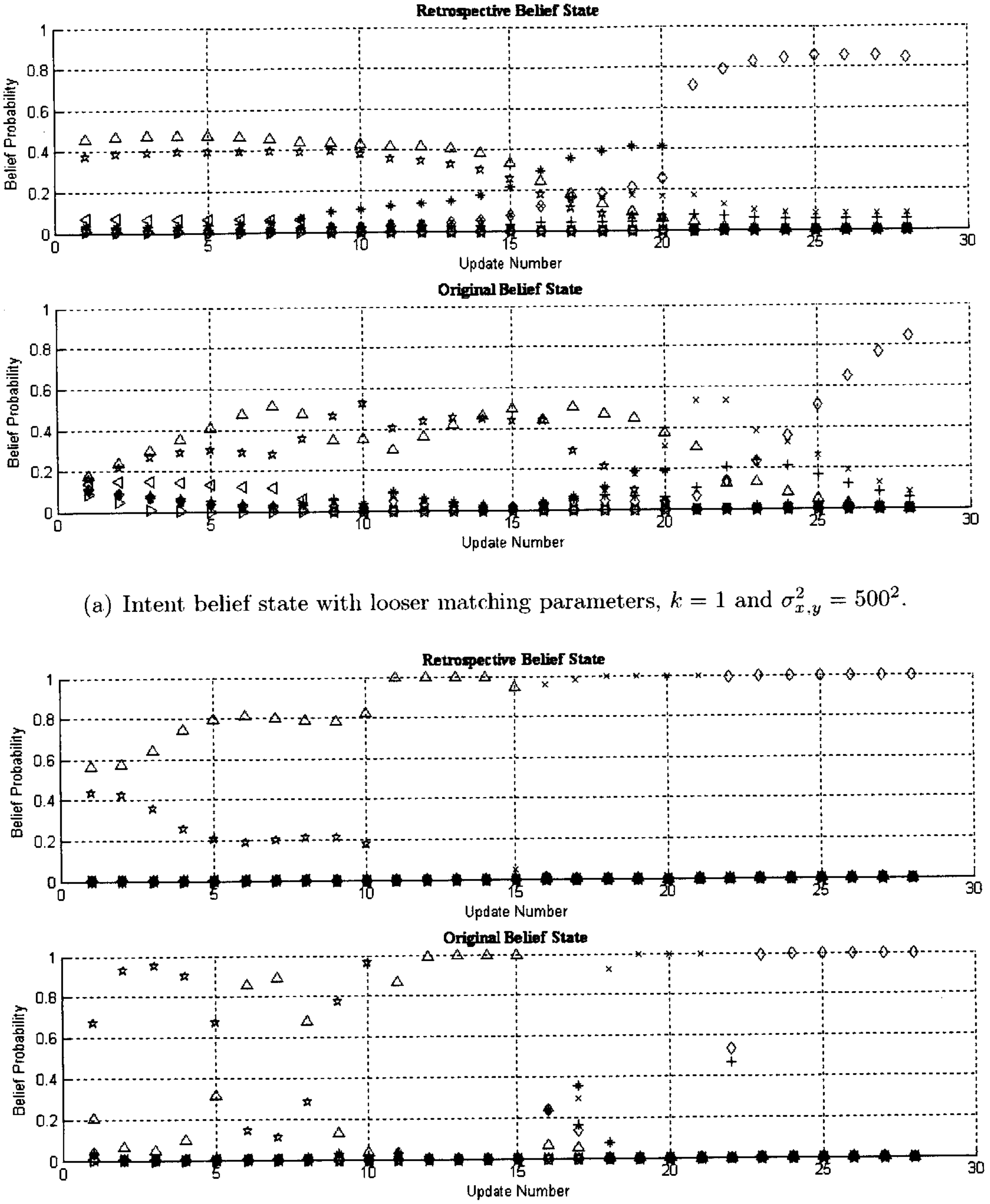
5.6.2 Observed Intent PDF (OI)

The model matching parameters govern how observed paths are interpreted as intent paths. This section shows two sets of results, first with "loose" parameters to create the OI, and then with "tighter" parameters. Running the original scenario (Section 5.2) with 2 sets of values for k, σ_x^2, and σ_y^2 (Table 5.8) will show how the parameters used to create the OI affect the intent estimator. For simplicity, the terms σ_x^2 and σ_y^2 will at times be represented by a single term, $\sigma_{x,y}^2$.

The first set of values for these parameters were chosen to reflect a lower belief that the contact's path would match the predicted path modeled by the intent estimator. For example, the original values of k and $\sigma_{x,y}^2$ were 5 and 200^2 respectively. These values indicate (from the properties of the von Mises and normal distributions) that the contact's future heading and (x, y) position would fall within a 30° and 200 m (along each axis) window respectively of the predicted heading and position about 70% of the time. Using the values $k = 1$ and $\sigma_{x,y}^2 = 500^2$ sets the heading and position window to 75° and 500 m respectively. Assigning these values for k and $\sigma_{x,y}^2$ results in a diminished ability to distinguish between different intent paths since the contact is allowed to deviate from each path to a larger extent (creating a relaxed OI). As a result, the probability values assigned for each intent would not differ as dramatically as before. Therefore, the intent estimator would have a more difficult time determining the predominant intent of the contact for each update. This is illustrated in Figure 5-14(a) which shows the resulting belief state plots from running the original scenario with $k = 1$ and $\sigma_{x,y}^2 = 500^2$. Figure 5-15 provides the belief state plots from the primary simulation run (Section 5.2) with the "normal parameters" (Table 5.8) for comparison. In this figure, it can be seen that the changes in the intent belief state are not as rapid as before with a lower belief value for the predominant intent. In addition, the belief state for the *Approach*(O-4) intent is predominant during updates 16–20 in the retrospective belief state plots. This resulted from the gradual increase of that intent's belief value, which can be seen from updates 9–15 of the same plot. This increase was produced by the relaxed OI model which allowed the contact's movements during this block of time to be attributed to the *Approach*(O-4) intent. By the time the contact was visibly moving towards O-2 at update 21, the belief values assigned to *Approach*(O-2) did not differ as dramatically as before from the other values. As a result, the belief value for the *Approach*(O-2) intent from updates 21 and 22 (in the original belief state plot) was smoothed out as illustrated by the retrospective belief state plot.

A more restrictive OI can be created by setting $k = 8$ and $\sigma^2_{x,y} = 50^2$. These values would set the heading and position window at 22° and 50 m respectively. The results of running the original scenario using these values are shown in Figure 5-14(b). From these belief state plots, it can be seen that changes in the belief state are more rapid and the resulting belief values for the predominant intent is higher than before. However, implementing a more restrictive OI may cause the intent estimator to perform poorly if the intent models provided to the estimator are not extremely accurate or an unexpected event occurs. From the original belief state plot in Figure 5-14(b), it can be seen that the predominant intent was believed to have changed quite frequently and that the beliefs in those changes were quite strong (as evidenced by the dramatic changes in its belief value). Such a view may not be representative of the contact's intent since it can be reasonably assumed that a ship does not generally change its intent frequently, and especially not in a short period of time.

The frequent shifts in the predominant intent from the original belief state plot were eventually smoothed out in the retrospective belief state plot. However, the



(a) Intent belief state with looser matching parameters, $k = 1$ and $\sigma_{x,y}^2 = 500^2$.

(b) Intent belief state with tighter matching parameters $k = 8$ and $\sigma_{x,y}^2 = 50^2$.

Figure 5-14: Simulation run with two modified pairs of model matching parameters. Environment state is the same as Figure 5-4(a).

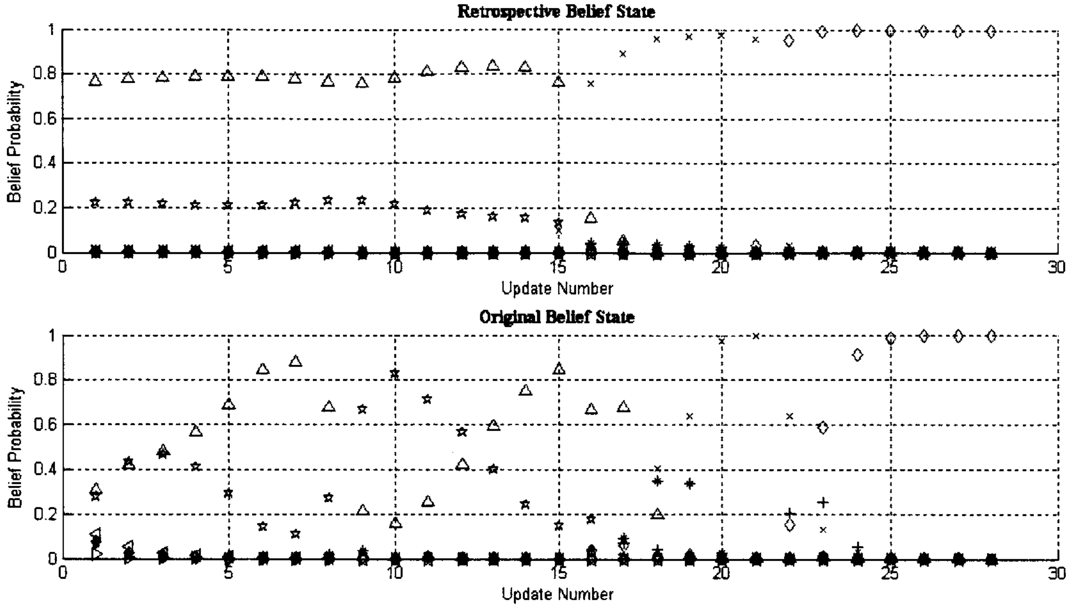


Figure 5-15: Intent belief state with normal matching parameters $k = 5$ and $\sigma^2_{x,y} = 200^2$ (from Figure 5-4).

retrospective belief state was visibly affected by the sudden and frequent changes in the intent probabilities as evident by the abrupt buildup of the *Follow*(V-1) intent. This is a deviation from the relatively consistent belief value of a predominant intent usually seen in the retrospective belief state plot.

5.7 Summary

The simulations provided in this chapter illustrated the ability of the "improved estimator" to calculate the intent belief state of an observed contact. The belief state calculated using the forward-backward procedure was shown to be a more robust, smooth, and representative indication of the contact's belief state than the "original belief state." The added capabilities of the intent estimator from Chapter 3 were also tested. These simulations illustrated how the estimator was able to use the contact's speed to help determine its intent (Section 5.3). In addition, the intent estimator was shown to function in a dynamic environment (Section 5.4). A series of simulations illustrated the performance of the intent estimator with an increasing number of targets in the environment (Section 5.5). Finally, the performance of the intent estimator was shown to vary depending on parameter settings.

[This page intentionally left blank]

Chapter 6

Obstacle Detection Results

The simulations presented in this chapter were designed to illustrate the effectiveness of the obstacle detection algorithm presented in Chapter 4. Several simulations are provided to demonstrate the impact of the intent estimator on the obstacle detection algorithm. The algorithm was then shown to be able to combine the analysis of multiple observation paths to create a single updatable obstacle inference map. Several obstacle inference maps were created for each simulation using a different hypothetical obstacle map generation method. The accuracy of each obstacle inference map is then calculated and analyzed.

6.1 Simulation Configuration

All the simulations presented in this chapter demonstrate the analysis of a provided path through a common environment. The valid targets in the environment are all stationary objects (O). Four objects have been defined and only a combination of these four objects are used as targets in any given simulation. A map of the obstacles in the simulation environment, along with the four targets, is shown in Figure 6-1. There are five obstacles located in the environment identified in the figure and displayed as completely shaded polygons. A series of grid cells overlaps each obstacle and extends a certain distance beyond the obstacle boundary. This distance is the

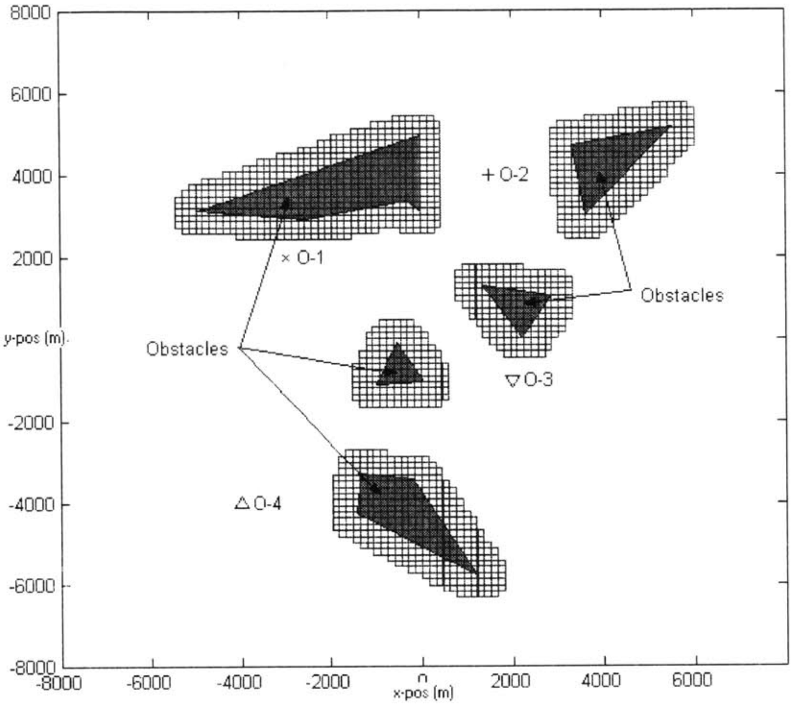


Figure 6-1: Simulation configuration.

Target	Symbol
O-1	×
O-2	+
O-3	∇
O-4	△

Table 6.1: Legend—Obstacle Detection.

modeled contact's minimum turning distance and defines the safety buffer that the contact maintains away from obstacles. Therefore, the grid cells represent areas in the environment through which the contact will not attempt to maneuver. The layout of the obstacles is known only to the contact and is provided to help explain (to the reader) the observed paths used in this chapter. The algorithm will be shown estimating the locations of the obstacles based on the observed paths.

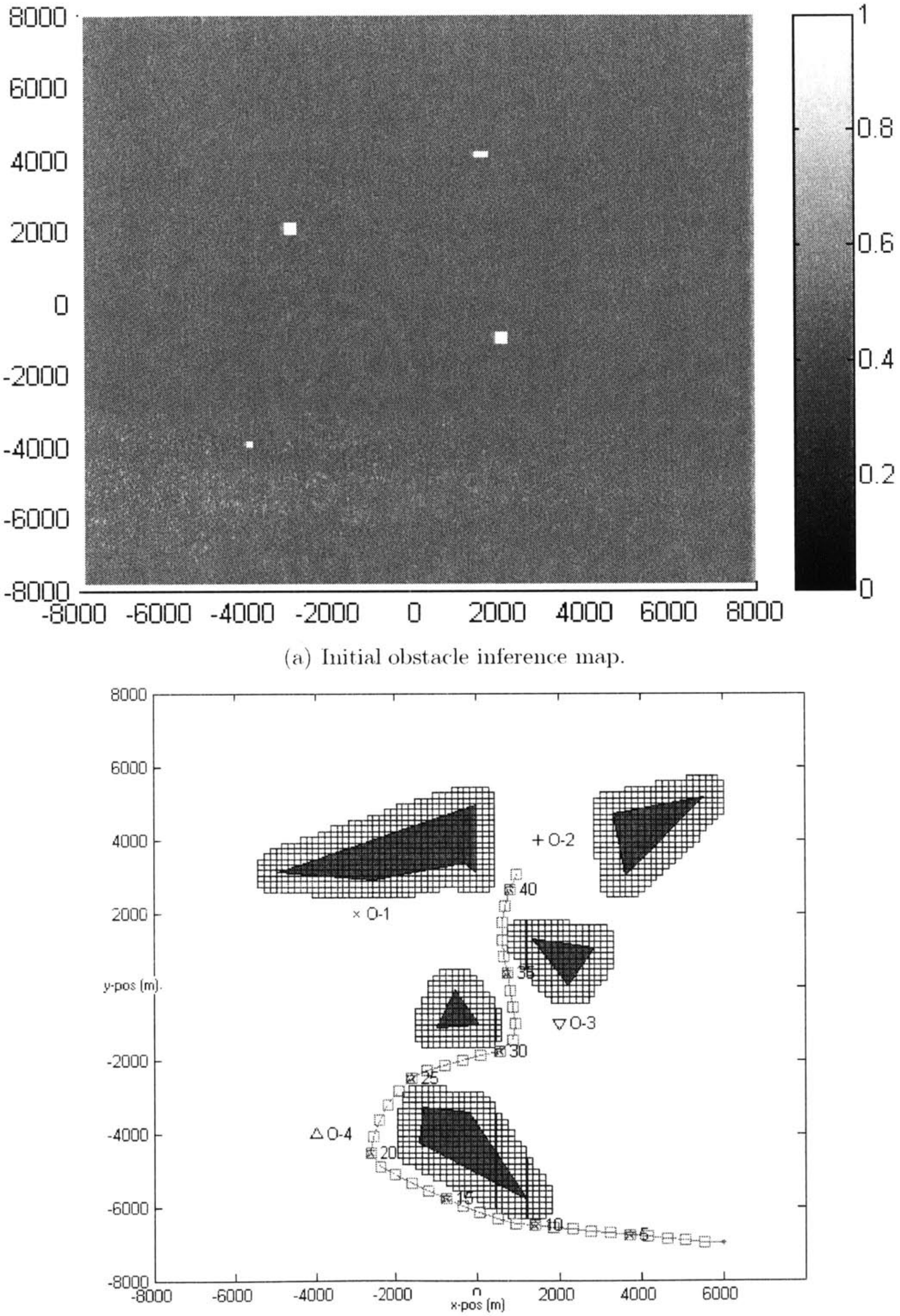
Each target (object) is labeled in the map with a number and prefixed with the letter 'O.' A unique symbol is used to identify each target as shown in Figure 6-1 and the legend is provided in Table 6.1. The location of each symbol in the map reflects the location of the corresponding target in the environment. In addition, since there is only one intent ($Approach$(O-x)) generated for each of these targets, the same symbol will be used to represent the corresponding intent in the belief state graphs.

6.2 Single Observation Path

This section presents the results from implementing the obstacle detection algorithm to analyze a single defined path in the environment described in Section 6.1. Since the obstacle detection algorithm presented in Chapter 4 relies on the accuracy of the intent estimator, the simulations in this section will focus on how the results of the intent estimator affect the obstacle inference maps produced. This is demonstrated by using a carefully selected combination of the four available targets for each simulation. Several methods for creating the hypothetical obstacle map are used for each simulation and the resulting obstacle inference maps are provided.

6.2.1 Example Simulation 1

The first simulation presents the results from implementing the intent estimation and obstacle detection algorithms in the environment provided in Figure 6-1. All four possible targets are considered and incorporated into the initial obstacle inference map (Figure 6-2(a)). Each cell in this map represents a unique area in the environment and stores a probability value of occupancy. This value indicates the belief that the represented location in the environment contains an obstacle. A complete description of the obstacle inference map structure is provided in Section 4.3. The map displayed in Figure 6-2(a) displays the value in each cell using a black to white gradient to represent the probability values between $[0, 1]$. The cells corresponding to the location of the identified targets are assigned a probability value of 1 since those areas are known to be occupied. All other cells are initially set at 0.5 (representing the unknown probability of occupancy). The observed contact's path is shown in Figure 6-2(b) and the obstacle zones created from analyzing that path are provided in Figure 6-3. The



(a) Initial obstacle inference map.

(b) Observed path and contact's knowledge of obstacles.

Figure 6-2: Environment and obstacle inference map for Simulation 1.

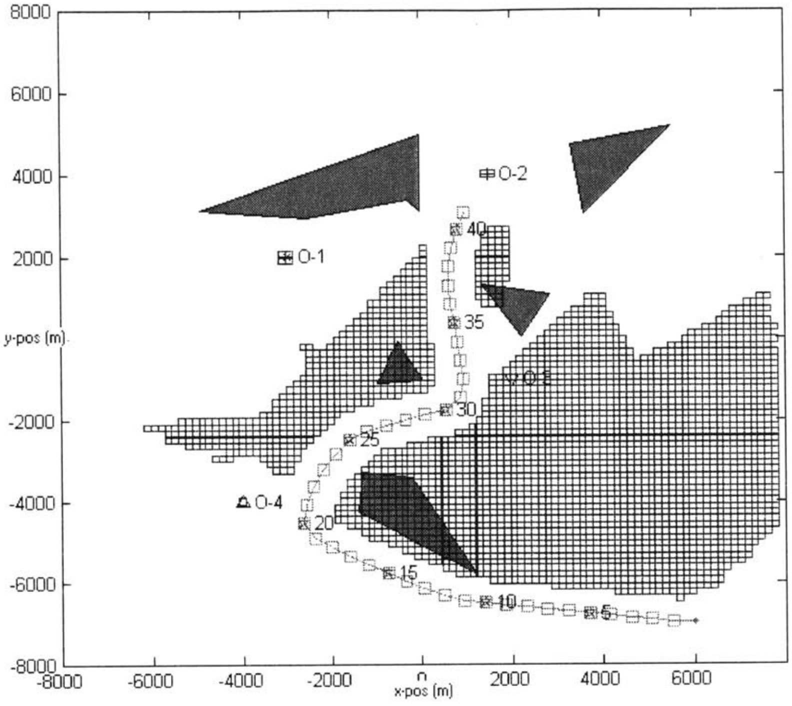


Figure 6-3: Hypothesized obstacle zones for Simulation 1.

observed path is represented by a connected series of contact observations. Each observation is used by the intent estimator to update the intent belief state of the contact. The position of the contact at the initial observation is represented by a '·' at the start of the path. The location of the contact at each subsequent observation is identified by a '□' symbol. Each '□' also represents the location of the contact when an update to the intent belief state is made. The location of the contact at every fifth update is labeled with the update number and a '⋆' symbol in Figure 6-2(b). The obstacle zones in Figure 6-3 are represented by the areas where grid cells are present.

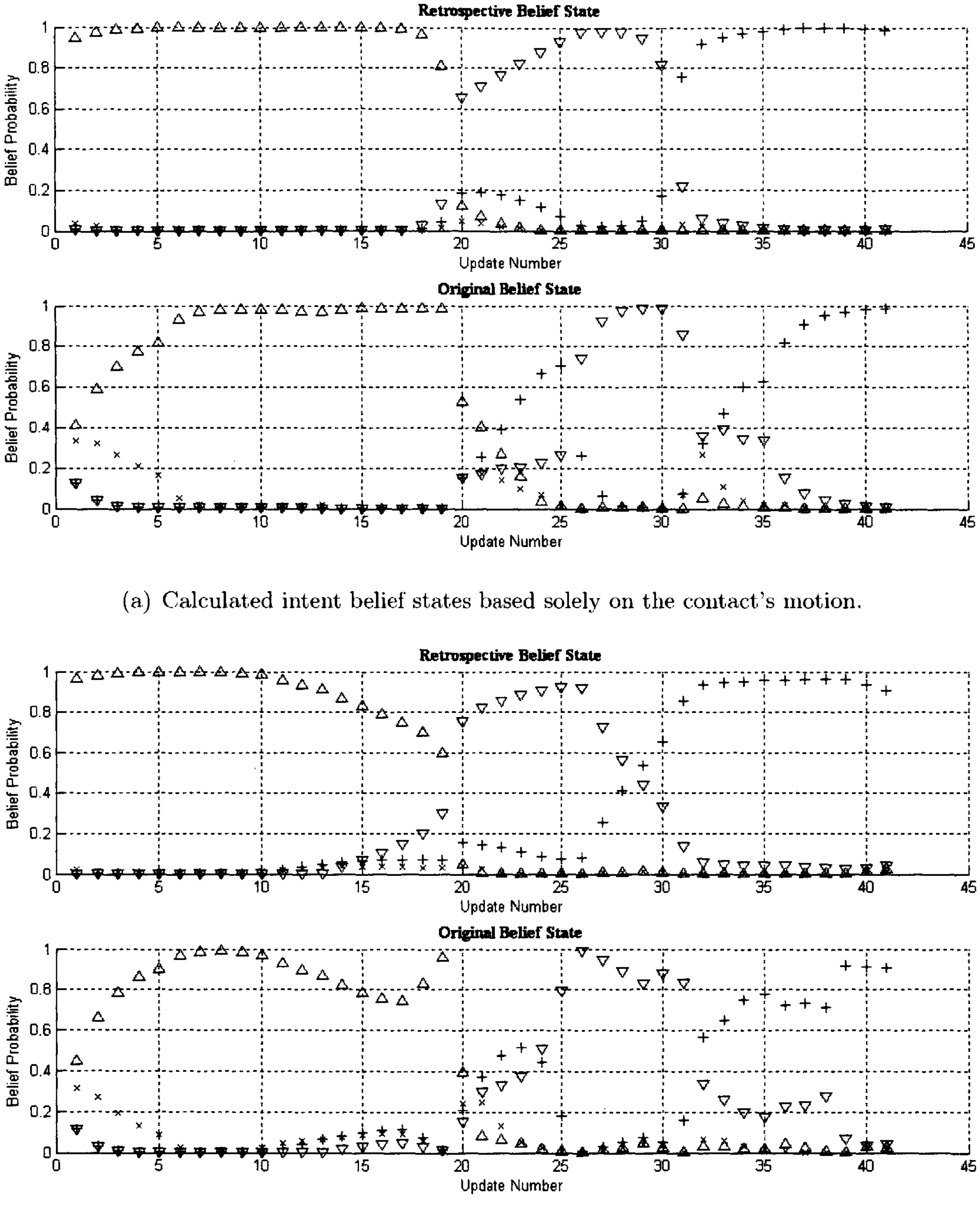
6.2.1.1 Belief State Comparisons

The obstacle zones provided in Figure 6-3 were generated from the intent belief states calculated by the intent estimator. These values are shown in Figure 6-4. The intent estimator was implemented with the same parameters provided in Tables 5.3 and 5.4

for generating these graphs. The graphs in Figure 6-4(a) represent the belief states calculated by the intent estimator without any knowledge concerning the location of obstacles in the environment. This is the belief state that the obstacle detection algorithm uses to infer the location of obstacles in the environment. If the intent estimator had prior knowledge concerning the location of these obstacles, then the intent models that the estimator generates would provide a better prediction of the maneuvers that the contact would make to perform a given intent. The belief states produced by the intent estimator using the location of obstacles in the environment are provided in Figure 6-4(b). Comparing the retrospective belief state in each of the two figures (in Figure 6-4) shows that the two plots are reasonably similar with the shift in the predominant intent between the two graphs differing by 1–2 updates. From this comparison, it can be argued that even though the intent belief state used by the obstacle detection algorithm was not calculated with a complete representation of the environment, the results can still be used since the difference is not dramatic. Therefore, the performance of the intent estimator is not dramatically affected if it does not take into consideration the location of obstacles in the environment.

6.2.1.2 Obstacle Inference Maps

The observed path is then divided into path segments using the retrospective belief state from Figure 6-4(a). For each path segment a set of hypothetical obstacle maps are generated (one for each intent), weighted by the its respective intent belief value, and then combined to create a primary hypothetical obstacle map for that path segment. This map is then combined with the initial obstacle inference map to produce an updated obstacle inference map. The creation of a primary hypothetical obstacle map, and combining it with the updated obstacle inference map, is then performed for each path segment. The probability values for the cells that represent the areas the contact's path moved through (along with a small buffer around that area) is then assigned a value of 0. Two approaches for creating the hypothetical obstacle maps were described in Section 4.6 and the resulting obstacle inference maps are pro-



(a) Calculated intent belief states based solely on the contact's motion.

(b) Calculated intent belief states using information concerning the location of obstacles in the environment.

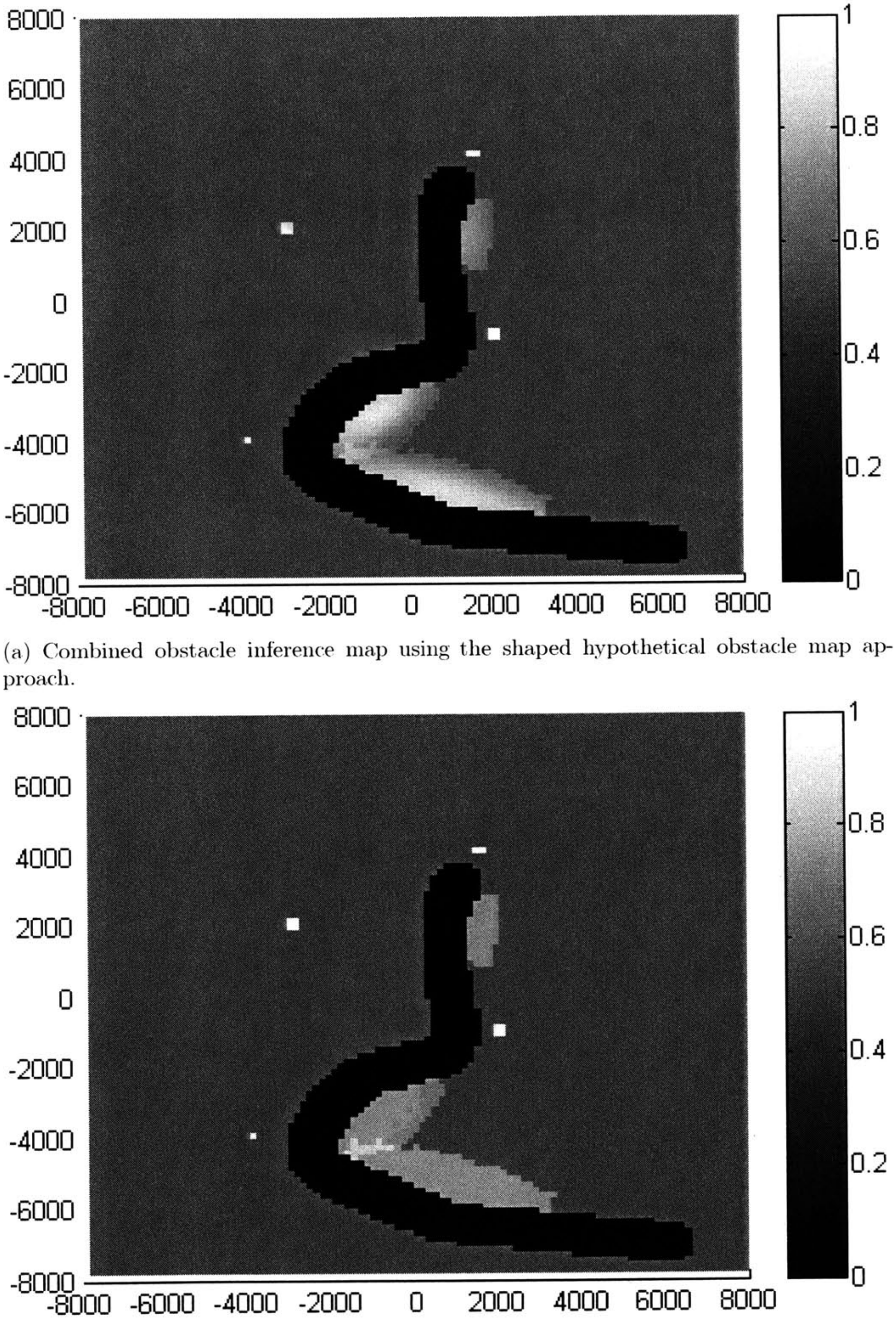
Figure 6-4: Intent belief states for Simulation 1.

vided in Figure 6-5. The blanket hypothetical obstacle map approach uses a constant probability value (p) of 0.75 to create each map, as explained in Section 4.6. Two additional obstacle inference maps were created for this simulation using the blanket hypothetical obstacle map approach with alternative values for p. The results are shown in Figure 6-6. The values used were 0.6 and 0.9 which were arbitrarily chosen to illustrate the effects of using a lower and higher probability value respectively from the original value used (0.75). The result was a generally lower belief that the cells in the obstacle zones are occupied using $p = 0.6$, and a higher belief using $p = 0.9$, when compared to using $p = 0.75$.

6.2.2 Example Simulation 2

The second simulation provides the results from implementing the intent estimation and obstacle detection algorithms in the environment provided in Figure 6-7(a). Only three out of the four possible targets are used. The target that is not included is O-1. The provided path through the environment is shown in Figure 6-7(b). The belief states calculated from the intent estimation process and the generated obstacle zones are provided in Figure 6-8. The resulting obstacle inference maps are shown in Figure 6-9.

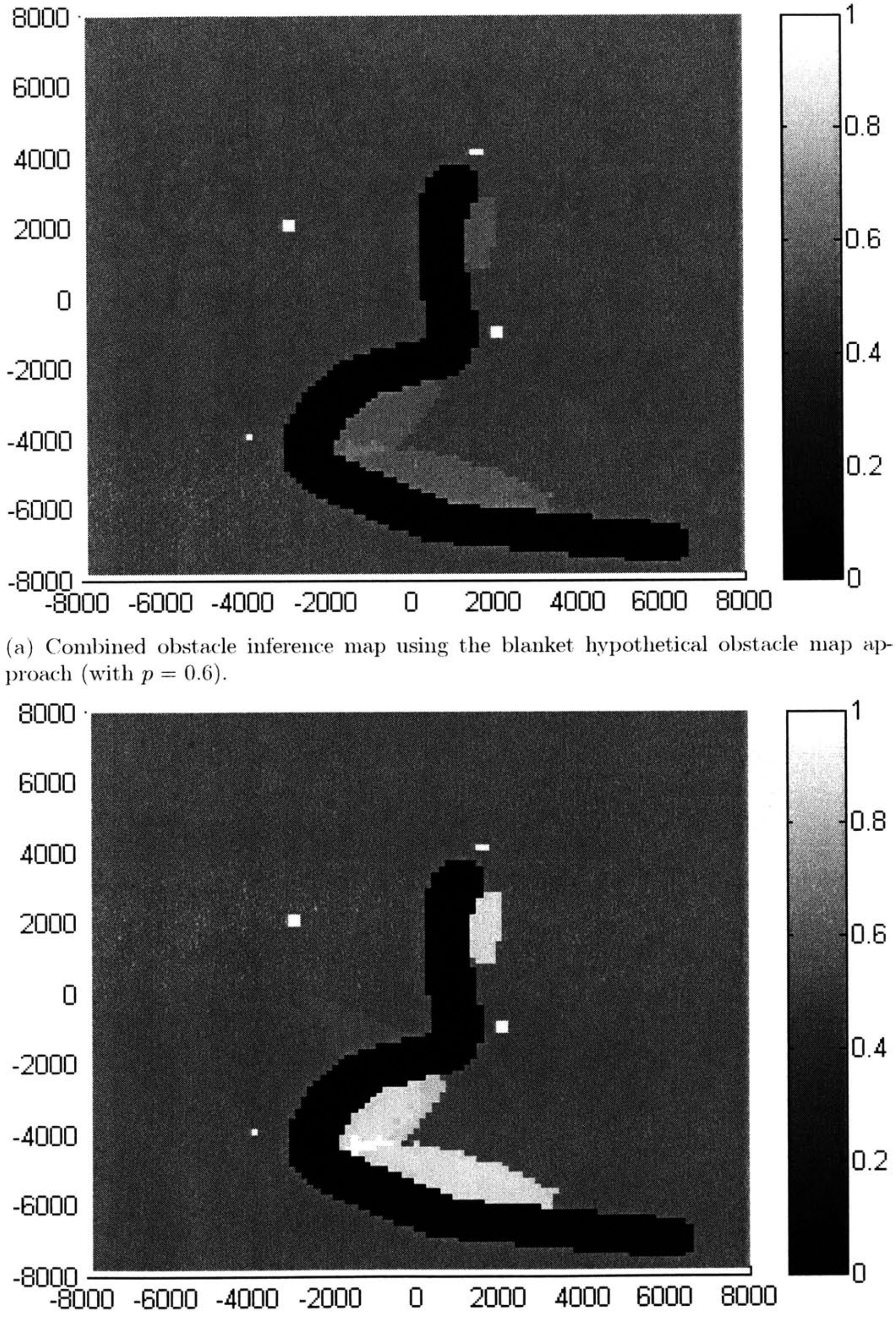
The obstacle inference maps obtain from this simulation are similar to those from the first simulation (Figure 6-5). This is because the removal of O-1 had a relatively small impact on the intent estimation process since the *Approach*(O-1) intent (represented by the symbol '×') was never assigned a high belief value in the first simulation (from the retrospective belief state graph in Figure 6-4(a)). There is a noticeable difference between the two figures representing the obstacle zones from each simulation (Figure 6-3 and 6-8(b)). The disparity is due the fact that each path segment analysis from Simulation 1 created obstacle zones with regards to O-1 and incorporated them into the obstacle inference map. However, since the probability in these zones are weighted by the belief value for the *Approach*(O-1) intent, the change in probability values contributed by these zones are not noticeable.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.

(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

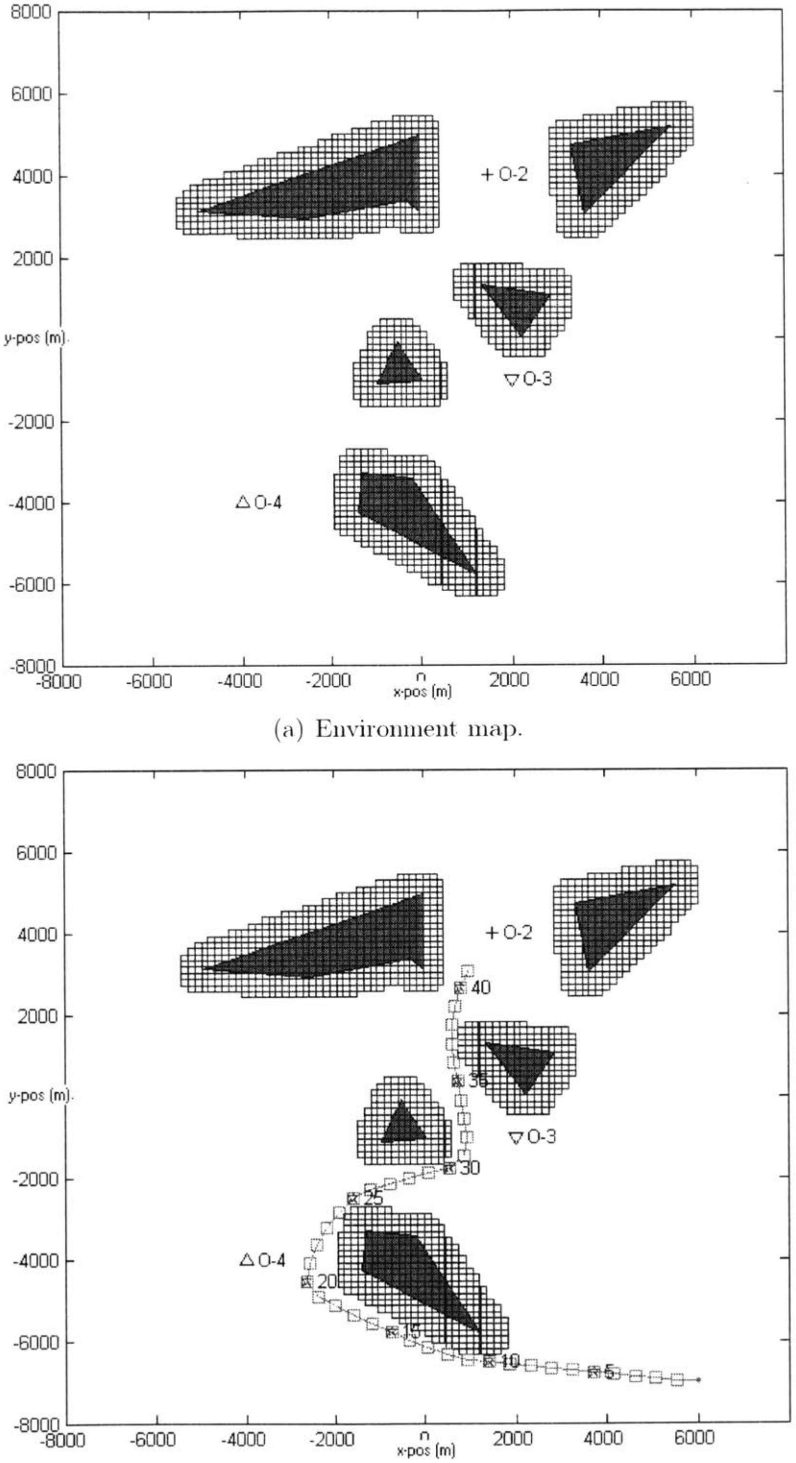
Figure 6-5: Obstacle inference maps for Simulation 1.



(a) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.6$).

(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.9$).

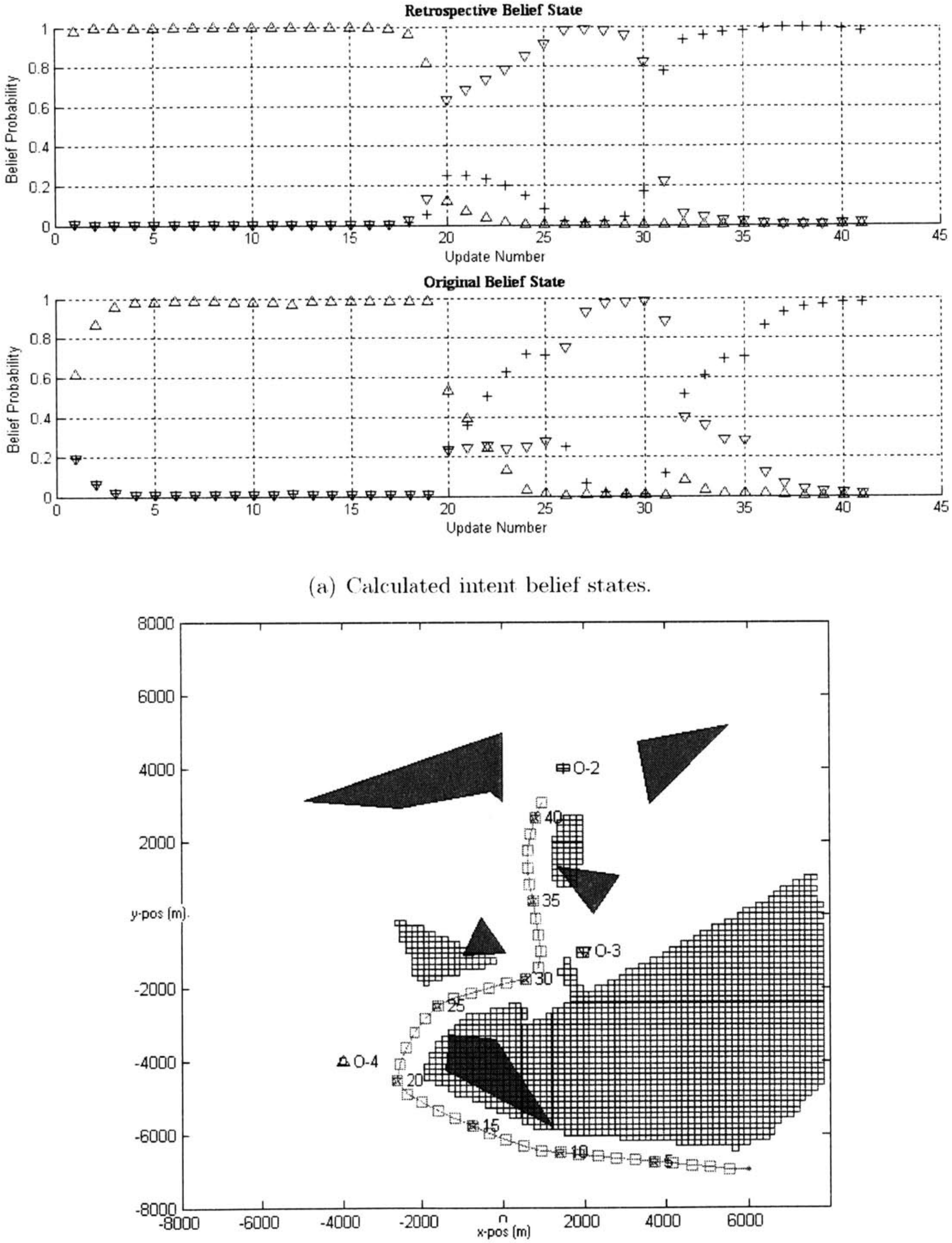
Figure 6-6: Additional obstacle inference maps for Simulation 1.



(a) Environment map.

(b) Environment map with observed path.

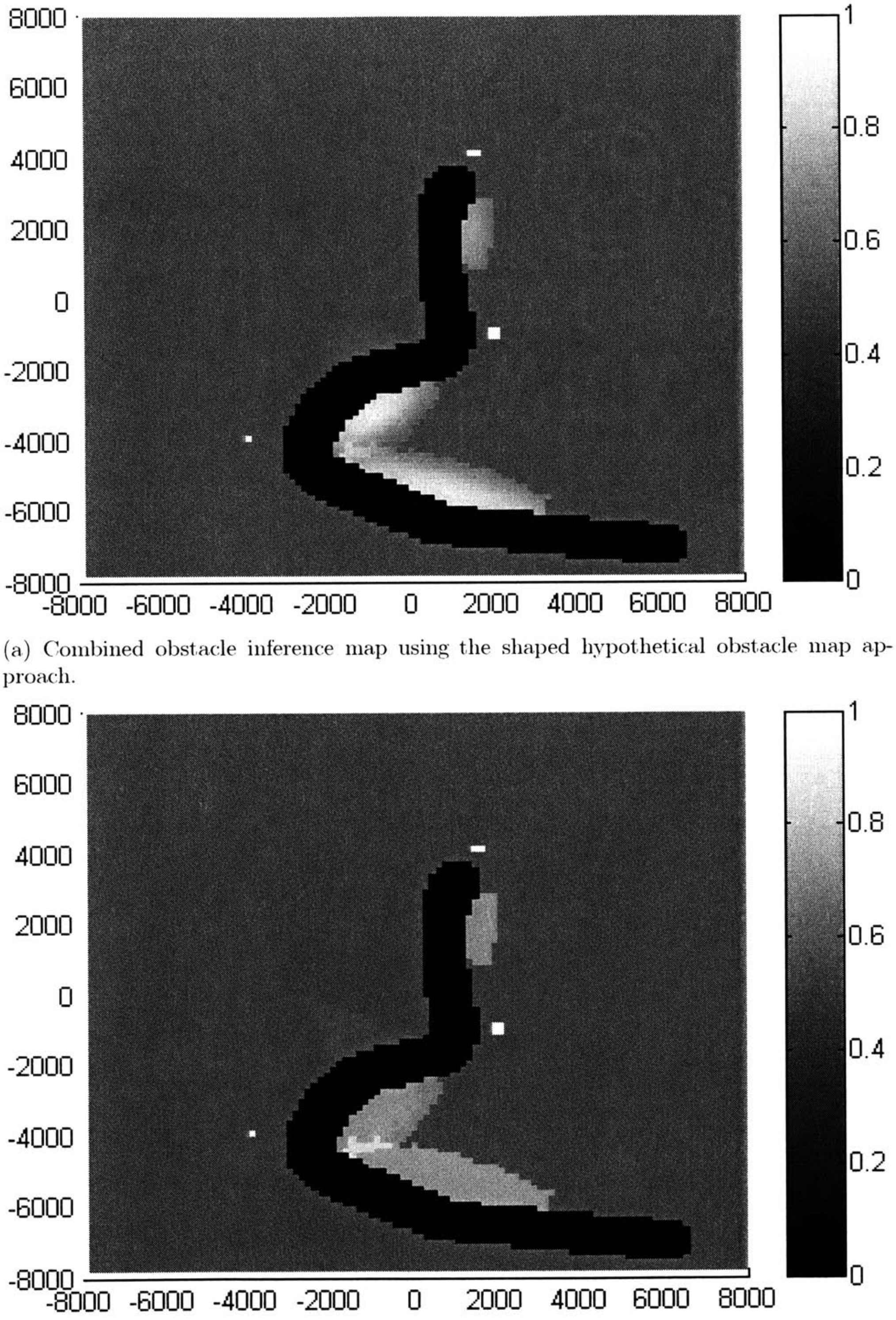
Figure 6-7: Environment map with observed path for Simulation 2.



(a) Calculated intent belief states.

(b) Hypothesized obstacle zones.

Figure 6-8: Intent belief states and hypothesized obstacle zones for Simulation 2.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.

(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

Figure 6-9: Obstacle inference maps for Simulation 2.

6.2.3 Example Simulation 3

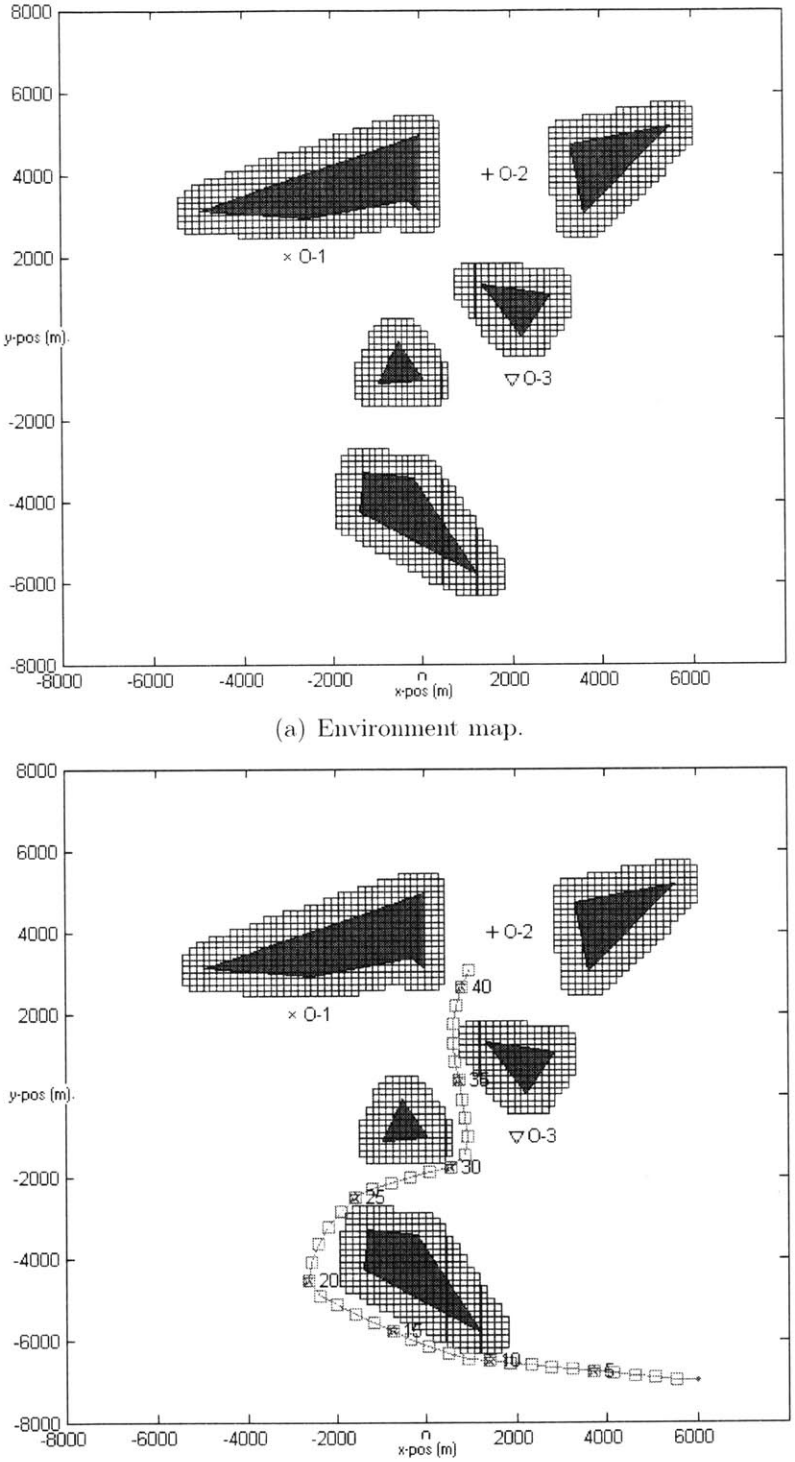
The third simulation provides the results from implementing the intent estimation and obstacle detection algorithms in the environment provided in Figure 6-10(a). Similar to Simulation 2, only three out of the four possible targets are used. The target that is omitted is O-4. The provided path through the environment is shown in Figure 6-10(b). The belief states calculated from the intent estimation process and the generated obstacle zones are provided in Figure 6-11. Two of the resulting obstacle inference maps are shown in Figure 6-12.

The obstacle zones obtained from this simulation (represented by Figure 6-11(b)) and the zones produced by Simulation 1 (Figure 6-3) are fairly similar. However, the obstacle inference maps from each simulation are noticeably different (Figures 6-5 and 6-12). This disparity is a direct result of the difference between the retrospective intent belief state produced by each simulation (Figures 6-4(a) and 6-11(a)). When the *Approach*(O-4) intent was removed from the intent state space, a shift in the intent belief state was seen which increased the belief value associated with the *Approach*(O-1) intent—which provides a plausible explanation for the initial portion of the provided path (updates 1–17).

6.2.4 Example Simulation 4

The fourth simulation provides the results from implementing the intent estimation and obstacle detection algorithms in the environment provided in Figure 6-13(a). In this simulation, only two of the four possible targets are used (O-2 and O-4). The provided path through the environment is shown in Figure 6-13(b). The belief states calculated from the intent estimation process and the generated obstacle zones are provided in Figure 6-14. Two of the resulting obstacle inference maps are shown in Figure 6-15.

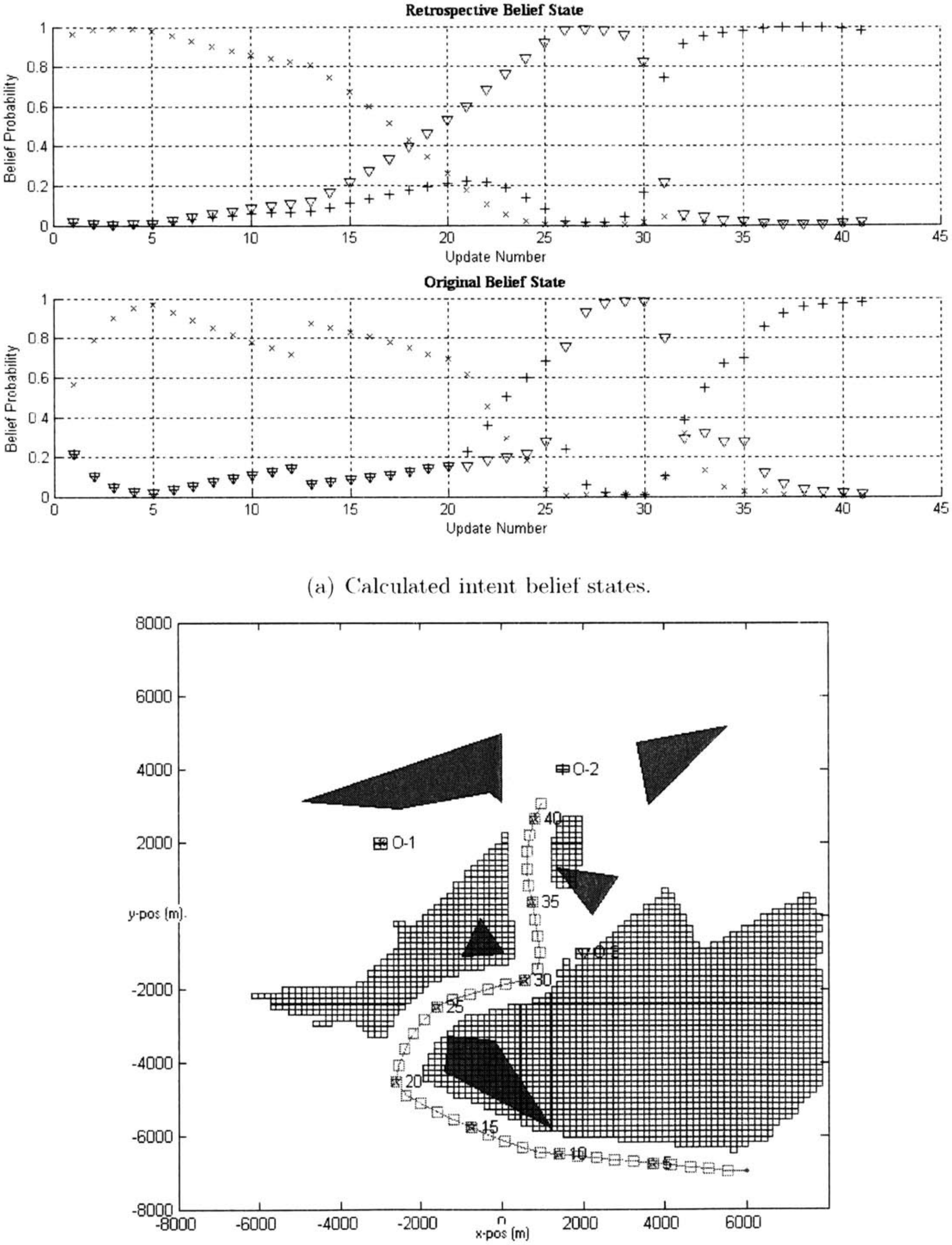
The most noticeable difference when comparing the results obtain from this simulation and the results from the previous simulations is the increased probability for the cells in the region covering the obstacle identified as *Obstacle X* (from Figure 6-



(a) Environment map.

(b) Environment map with observed path.

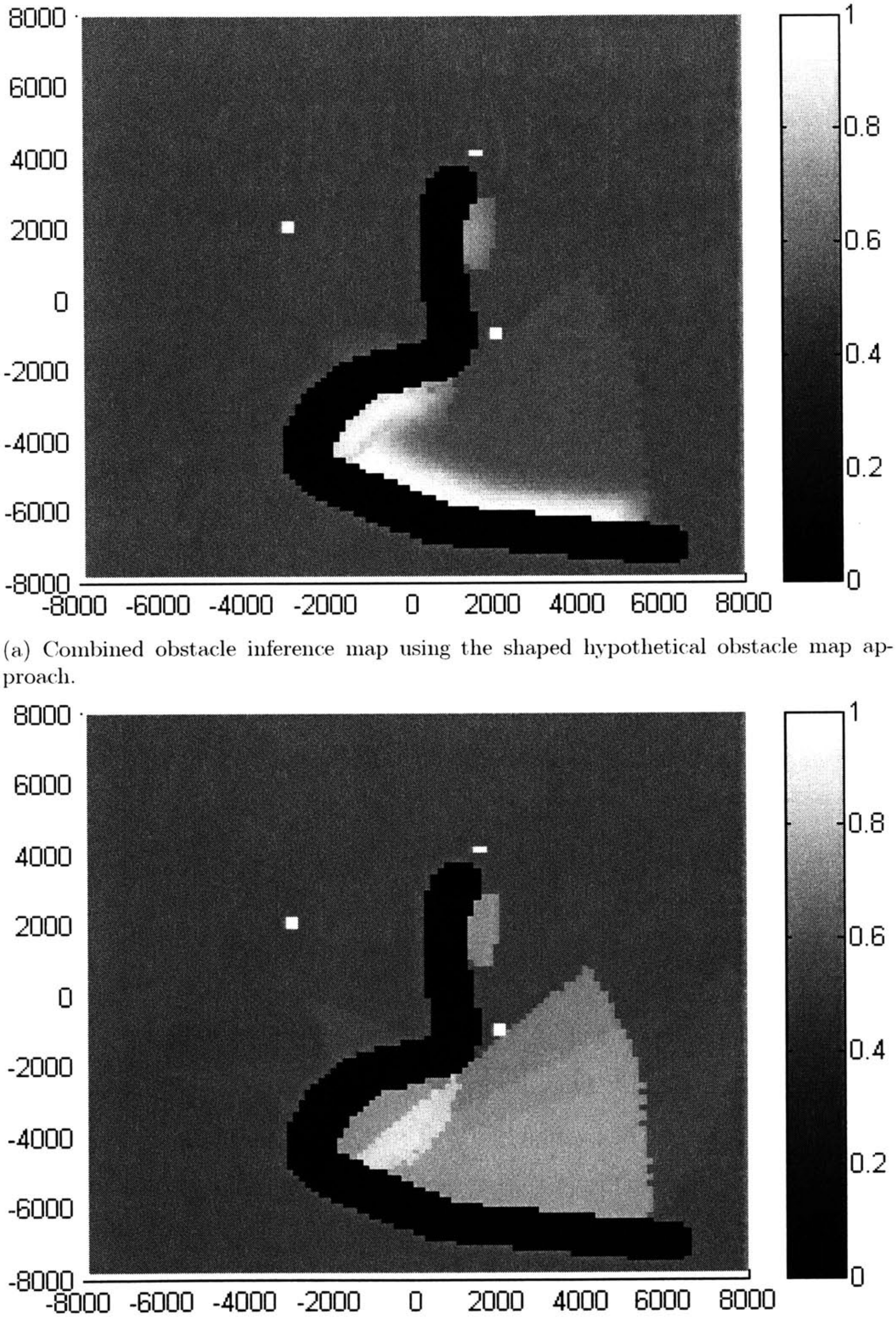
Figure 6-10: Environment map with observed path for Simulation 3.



(a) Calculated intent belief states.

(b) Hypothesized obstacle zones.

Figure 6-11: Intent belief states and hypothesized obstacle zones for Simulation 3.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.

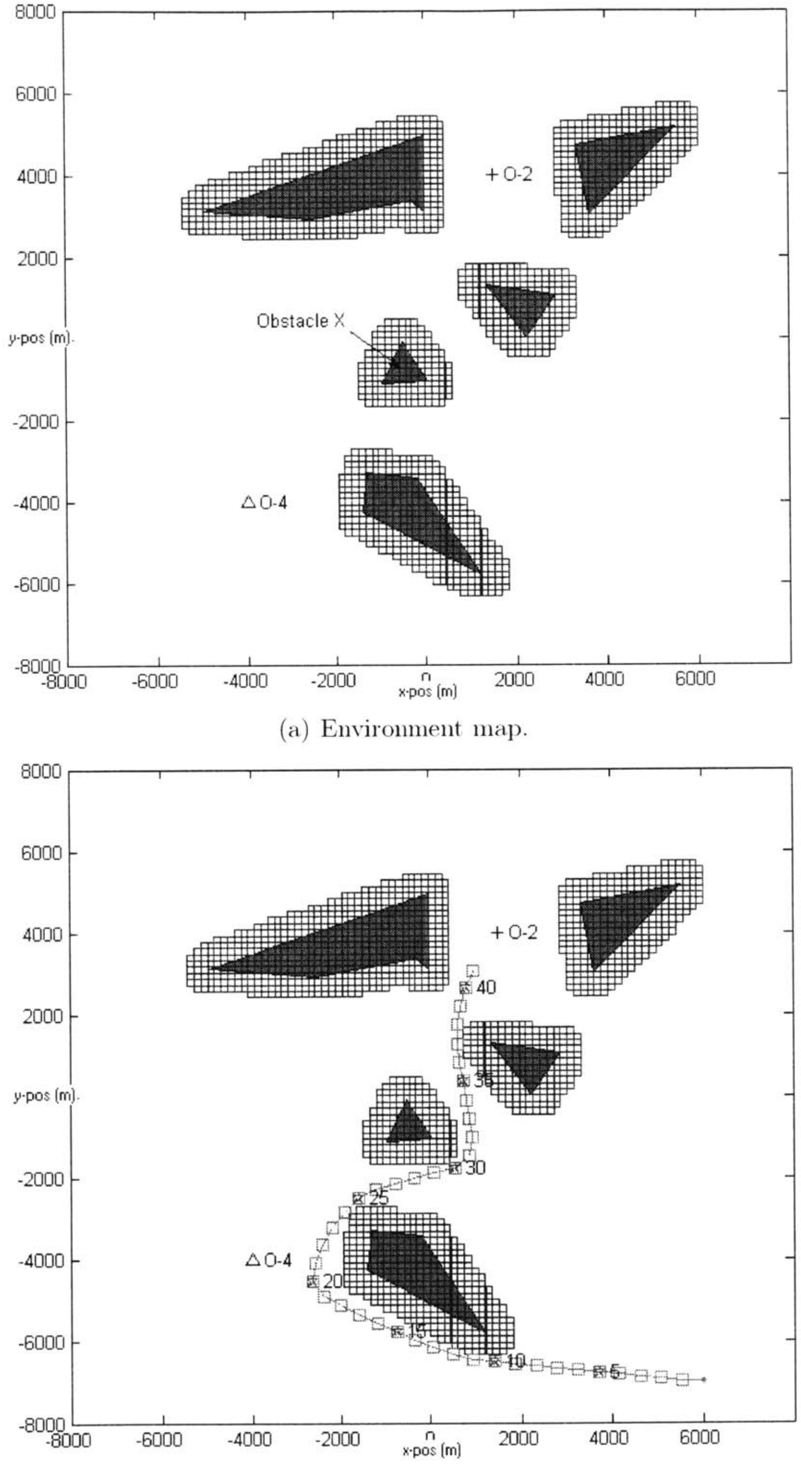
(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

Figure 6-12: Obstacle inference maps for Simulation 3.

13(a)). The reason why the probability values of the cells representing this region, from Simulations 1–3, did not reflect a noticeable belief that an obstacle was present was due to the inclusion of *Approach*(O-3) in the intent state space. The retrospective belief states from Simulations 1–3 (Figures 6-4(a), 6-8(a), and 6-11(a)) illustrate that the predominant estimated intent of the contact from update 20–30 was *Approach*(O-3). Regardless of whether or not *Approach*(O-3) was the actual intent of the contact, the assignment of a high belief value to that intent is reasonable since the observed path (during that period of time) could be strongly interpreted as an approach to O-3. Therefore, the belief values for *Approach*(O-3) dominate the intent belief state during updates 20–30 and the *Approach*(O-2) intent received a lower belief value. Since an obstacle zone covering the area of Obstacle X would only be created for the *Approach*(O-2) intent during updates 20–30, the low belief value for that intent prevented a noticeable increase in the probability values for the cells in that area. With the *Approach*(O-3) intent removed in this simulation, the *Approach*(O-2) intent became the dominant intent from updates 20–30. The result was a noticeable increase in the belief that a obstacle is present in the general area around Obstacle X.

6.3 Combined Obstacle Inference Map

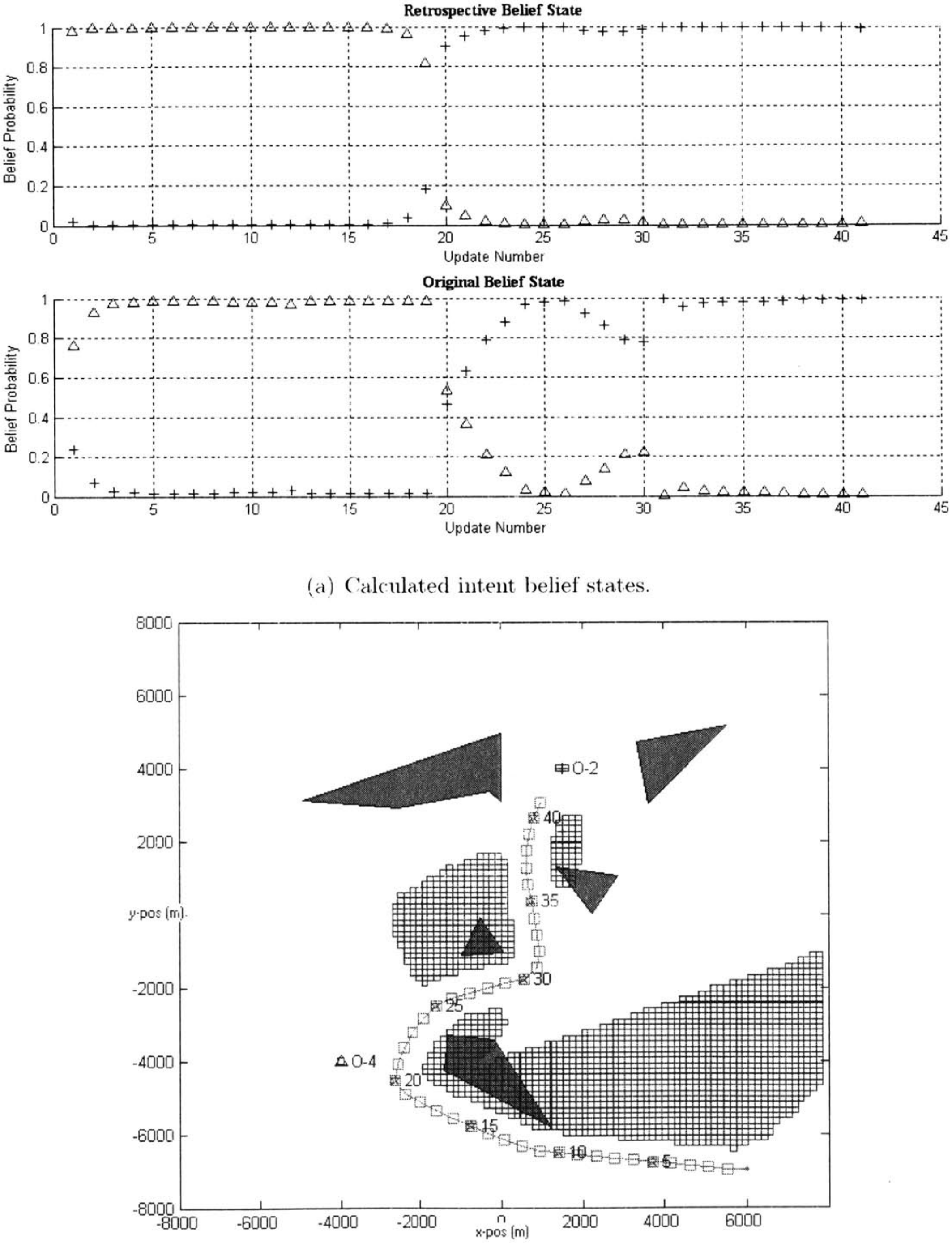
Simulations 1–4 presented the effect that the intent estimator has on the obstacle detection algorithm. The obstacle inference maps presented for each of the simulations only incorporated the analysis of a single path. Three additional paths are analyzed by the obstacle detection algorithm in this section. The configurations of these three simulations are the same as Simulation 1. The results are provided in Figures 6-16 through 6-21. The obstacle zones, intent belief states, and two obstacle inference maps obtained by the analysis of each path are provided in the figures. Figure 6-22 illustrates the obstacle zones from combing the analyses of the path from Simulation 1 and the three presented in this section. The combined obstacle inference maps are



(a) Environment map.

(b) Environment map with observed path.

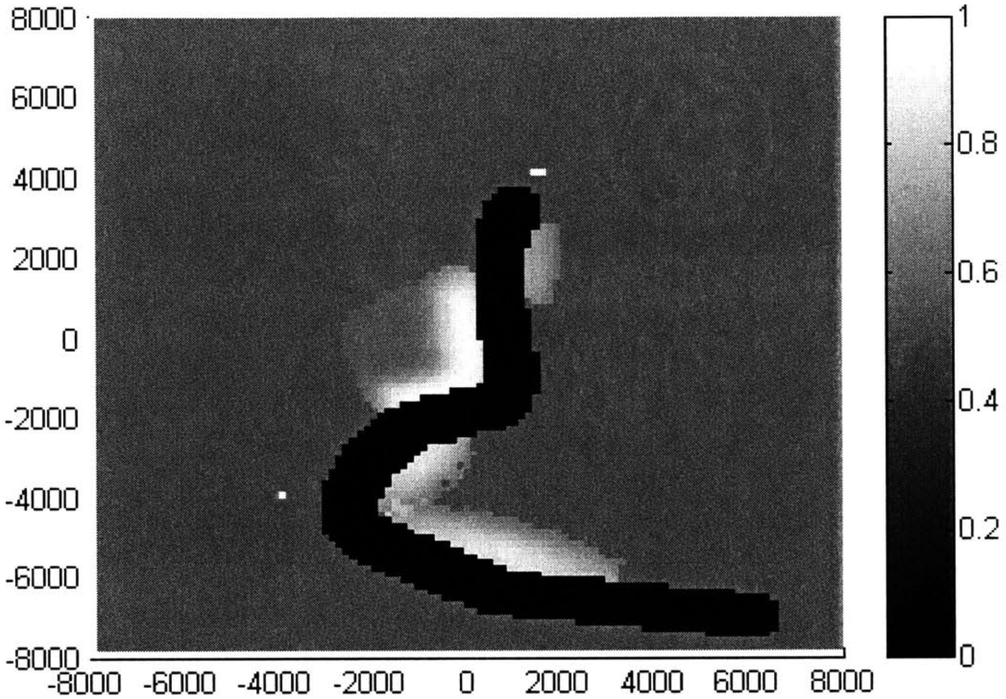
Figure 6-13: Environment map with observed path for Simulation 4.



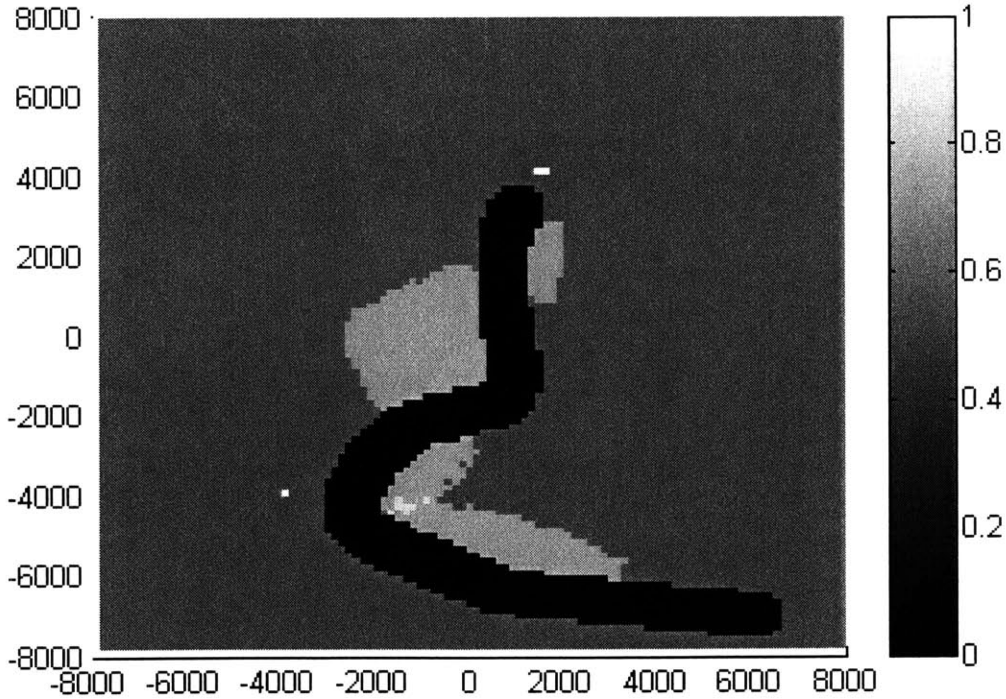
(a) Calculated intent belief states.

(b) Hypothesized obstacle zones.

Figure 6-14: Intent belief states and hypothesized obstacle zones for Simulation 4.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.



(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

Figure 6-15: Obstacle inference maps for Simulation 4.

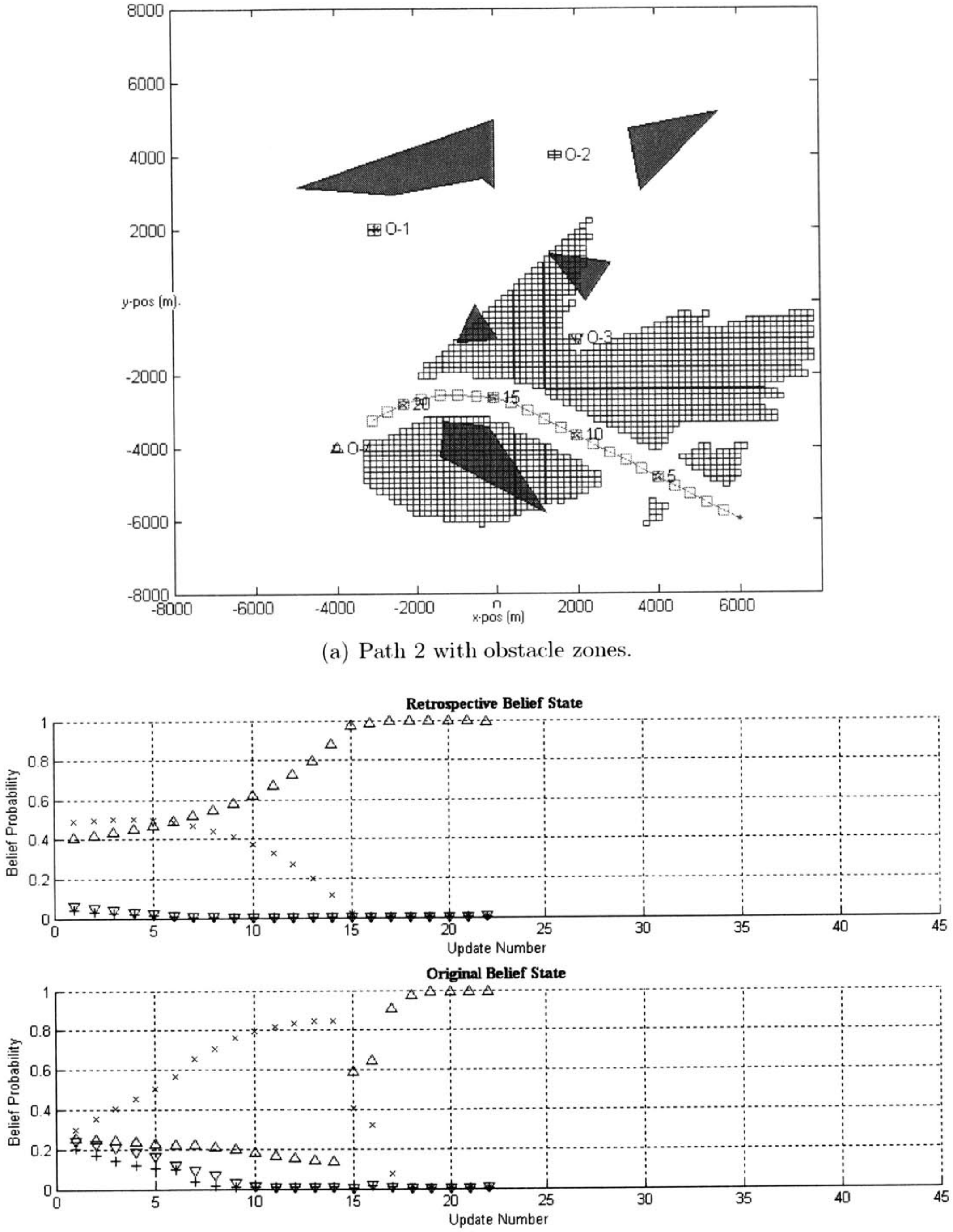
shown in Figure 6-23. An observation can be made from Figure 6-16(a) that the hypothesized obstacle zones do not extend into the upper-left half of the map. This is a result of relying on the observed path to generate the obstacle zones. Since the path does not extent into the specified area, the obstacle zones cannot be generated (using the method described in Section 4.5.1) for that area.

6.4 Error Analysis

This section compares each of the obstacle inference maps created in Sections 6.2 and 6.3 to the actual obstacle map of the environment (which is represented using the same structure as the obstacle inference maps). For a simulation which included all four targets, the actual map is illustrated in Figure 6-24. Since there is no ambiguity as to whether a location in the environment is occupied or not, the probability value for each cell in this map is either 0 (black) or 1 (white), representing unoccupied areas and occupied areas respectively. The disparities between an obstacle inference map (which will be represented by B) and the actual obstacle map (represented by A) is quantified by calculating the mean square error as follows:

$$e = \frac{1}{C}\sum_{c=1}^{C}[A(c) - B(c)]^2,$$

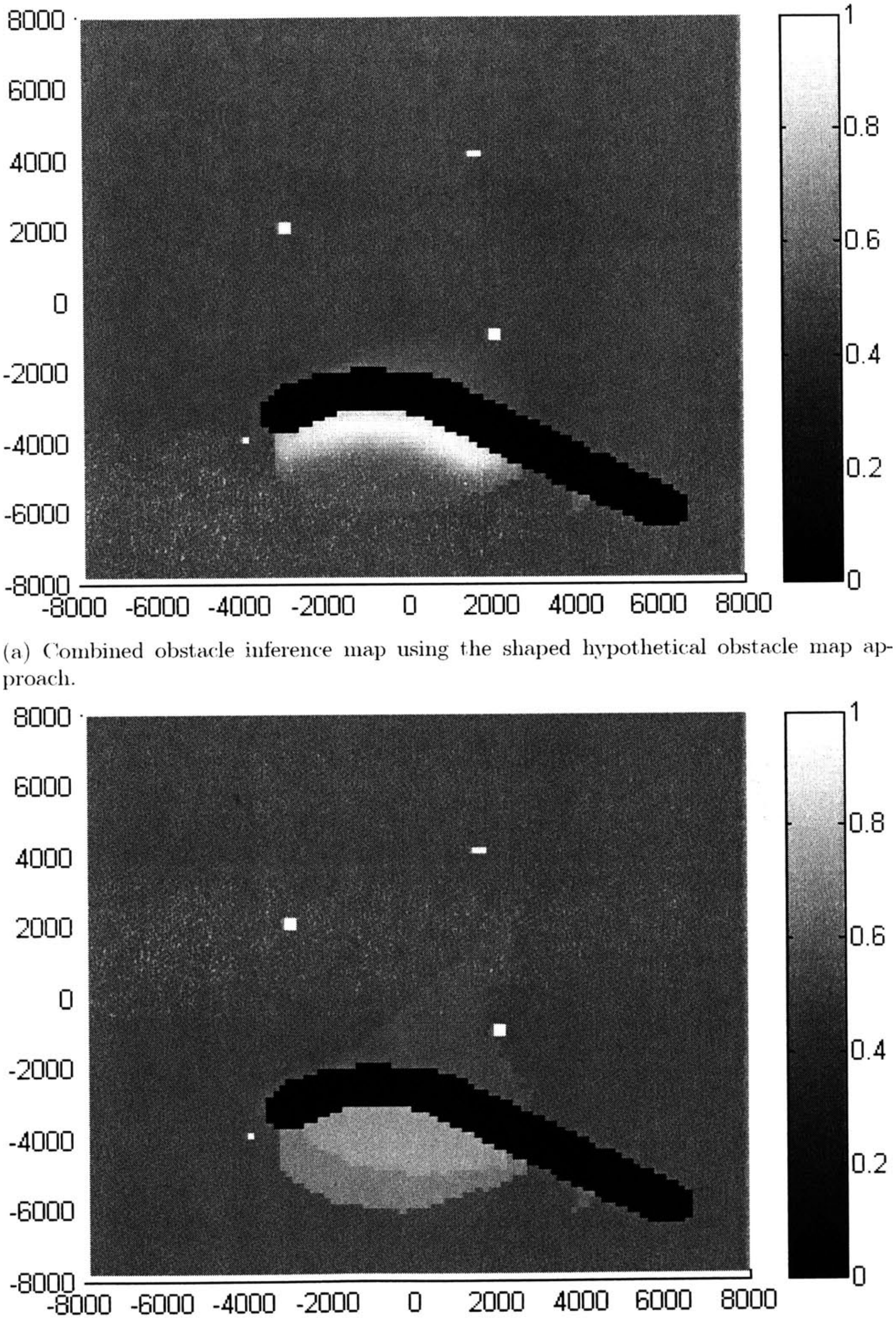
where e is the error value, C is the number of cells in the map, $A(c)$ is the actual value that the area represented by cell c is occupied, and $B(c)$ is the inferred value. Since the domain for the values in each cell is [0,1], the domain for the calculated error value is [0,1]. Four obstacle inference maps are analyzed for each of the simulations in Sections 6.2 and 6.3. Each of these maps were generated using a different approach for creating the hypothetical obstacle map. The four methods used were 1) the shaped approach, 2) the blanket approach with $p = 0.75$, 3) the blanket approach with $p = 0.6$, and 3) the blanket approach with $p = 0.9$. The results are shown in Table 6.2. The table also provides the error values calculated for two other obstacle



(a) Path 2 with obstacle zones.

(b) Calculated intent belief states.

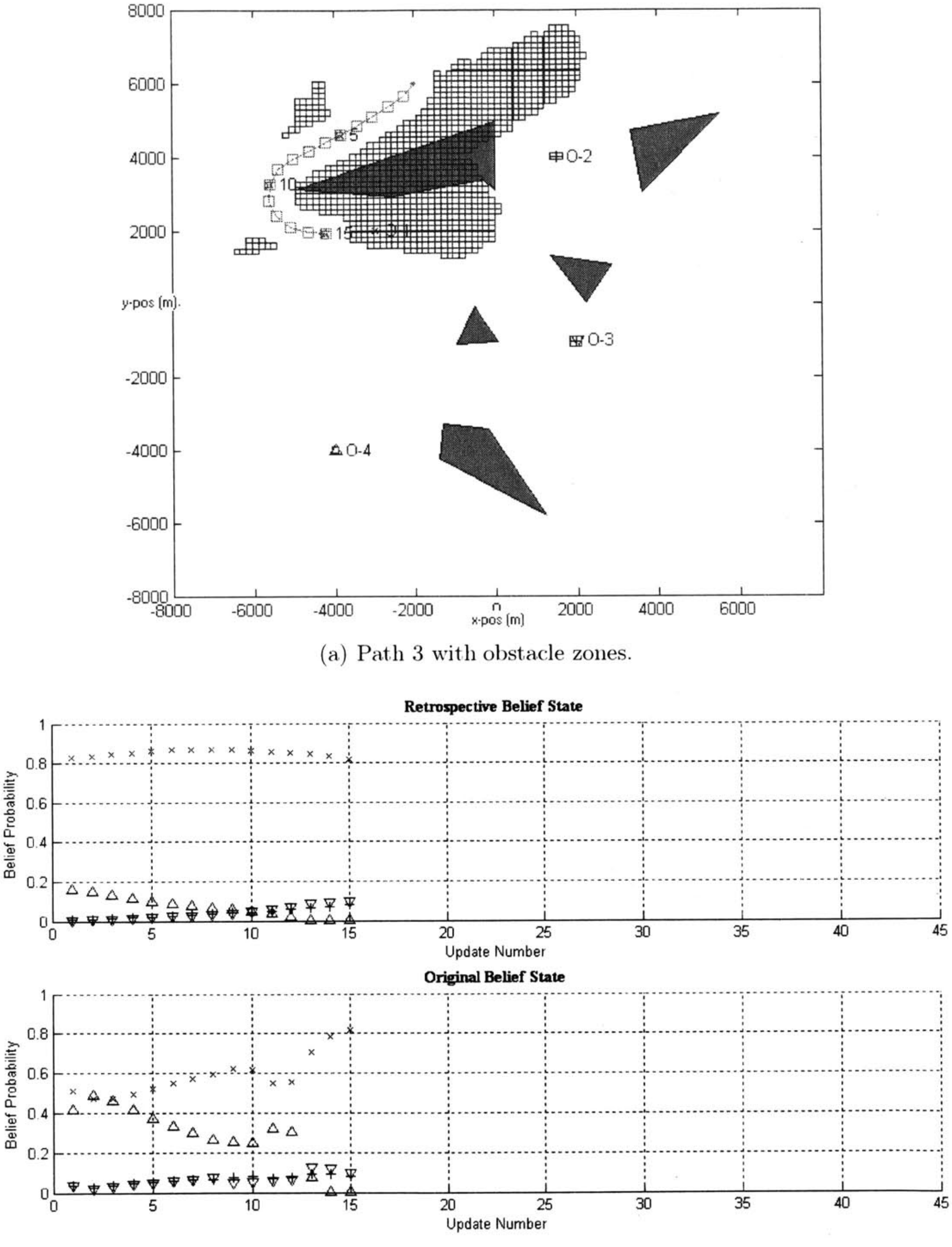
Figure 6-16: Obstacle zones and intent belief states for Path 2.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.

(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

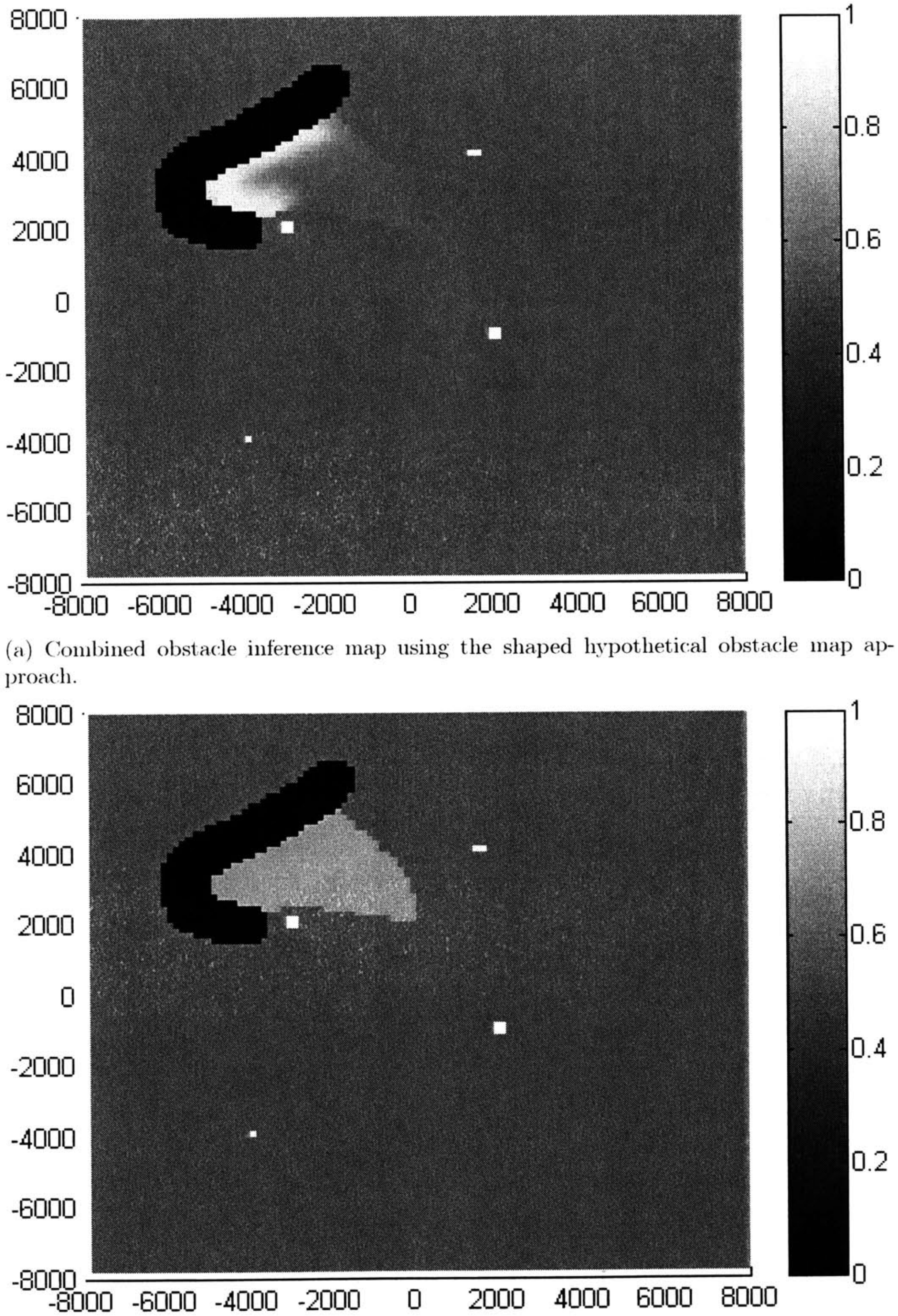
Figure 6-17: Obstacle inference maps for Path 2.



(a) Path 3 with obstacle zones.

(b) Calculated intent belief states.

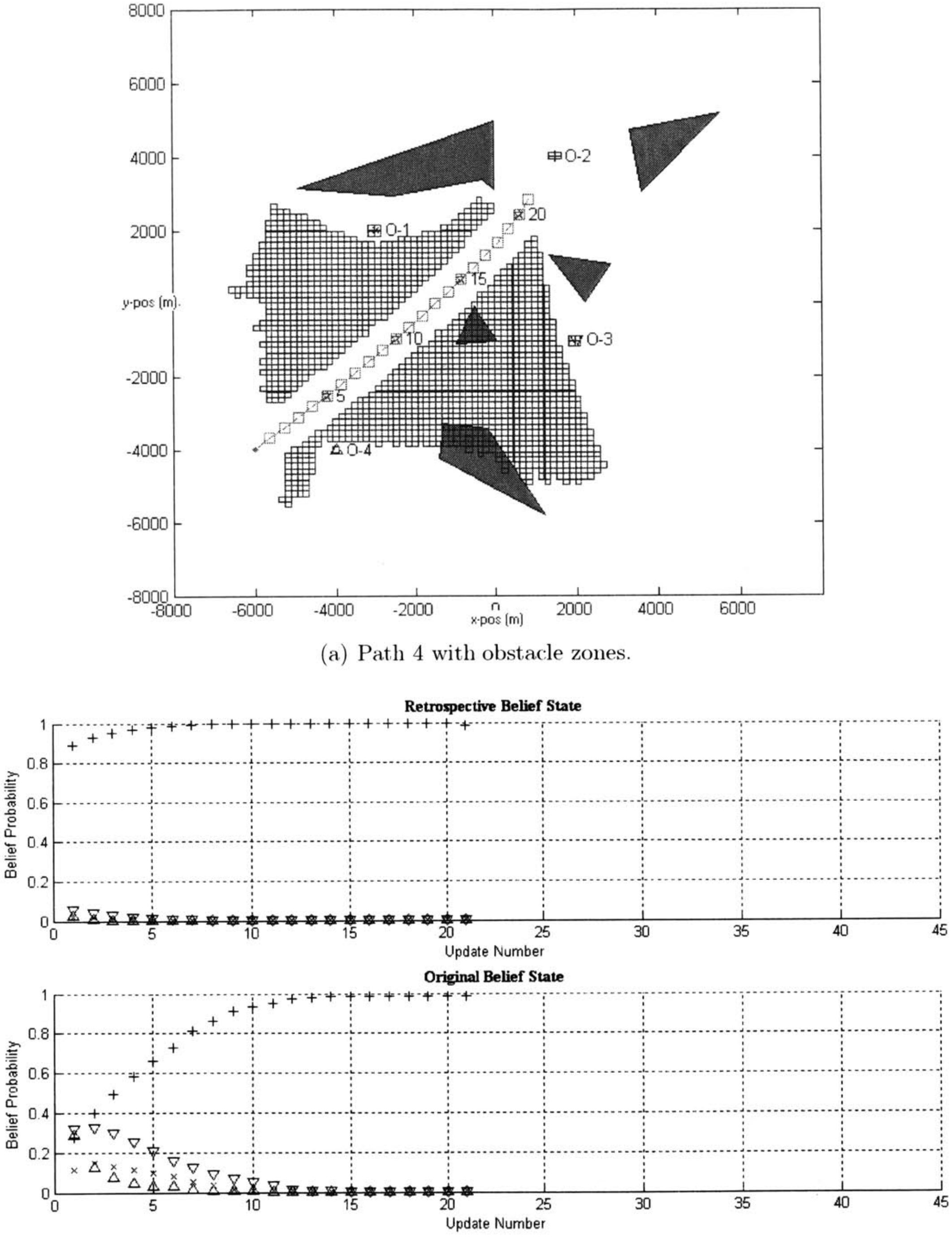
Figure 6-18: Obstacle zones and intent belief states for Path 3.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.

(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

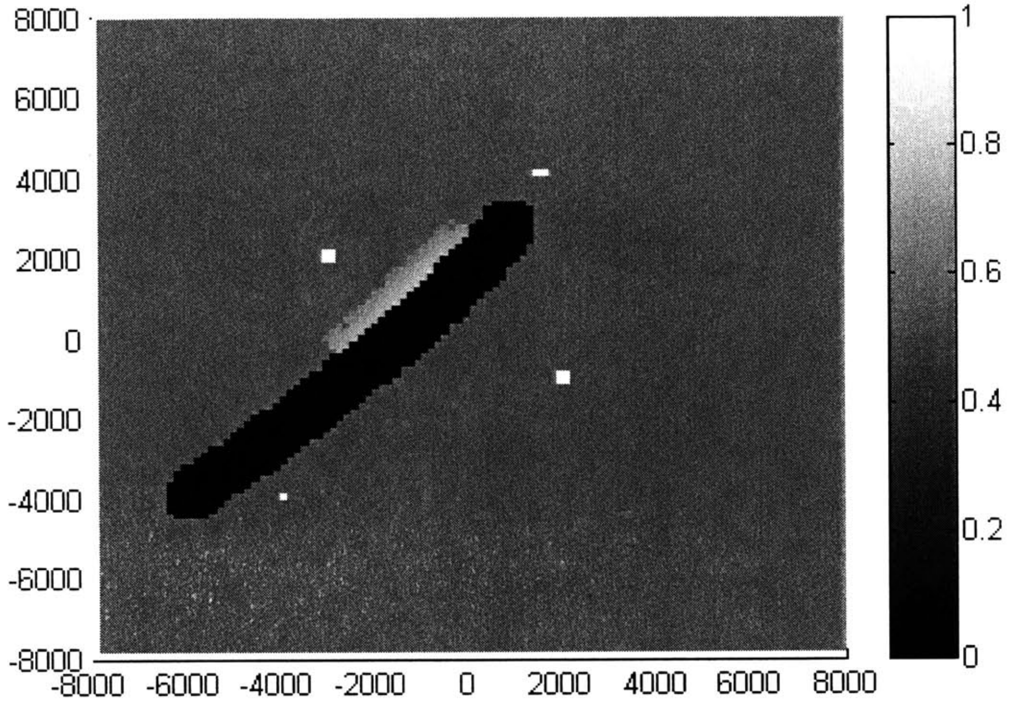
Figure 6-19: Obstacle inference maps for Path 3.



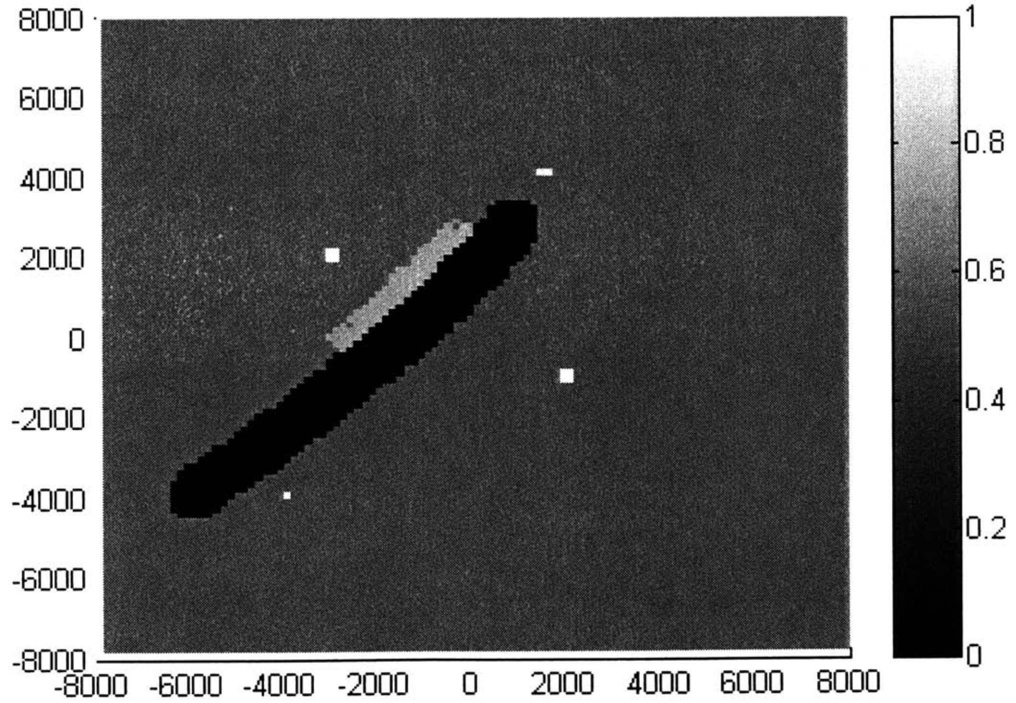
(a) Path 4 with obstacle zones.

(b) Calculated intent belief states.

Figure 6-20: Obstacle zones and intent belief states for Path 4.



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.



(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

Figure 6-21: Obstacle inference maps for Path 4.

	Sim 1	Sim 2	Sim 3	Sim 4	Path 2	Path 3	Path 4	Combined
Shaped	0.2316	0.2316	0.2421	0.2372	0.2447	0.2439	0.2391	0.2098
Blanket ($p = 0.75$)	0.2312	0.2312	0.2609	0.2386	0.2451	0.2432	0.2394	0.2082
Blanket ($p = 0.6$)	0.2276	0.2277	0.2381	0.2303	0.2391	0.2408	0.2375	0.1986
Blanket ($p = 0.9$)	0.2363	0.2362	0.2886	0.2496	0.2529	0.2470	0.2418	0.2196
Cleared Path Only	0.2261	0.2262	0.2261	0.2263	0.2363	0.2401	0.2364	0.1948
Blank Map	0.2498	0.2498	0.2498	0.2499	0.2498	0.2498	0.2498	0.2498

Table 6.2: Error values.

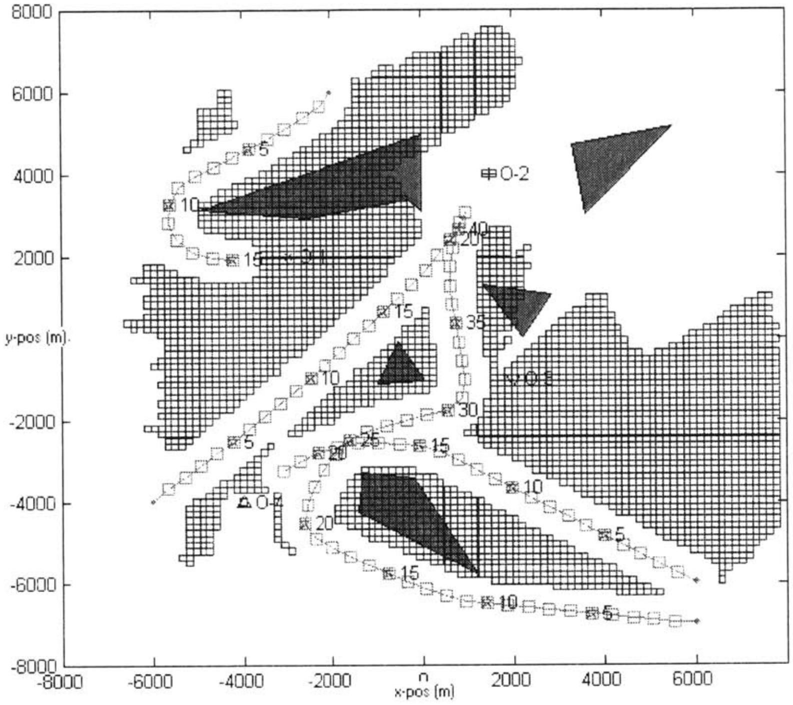
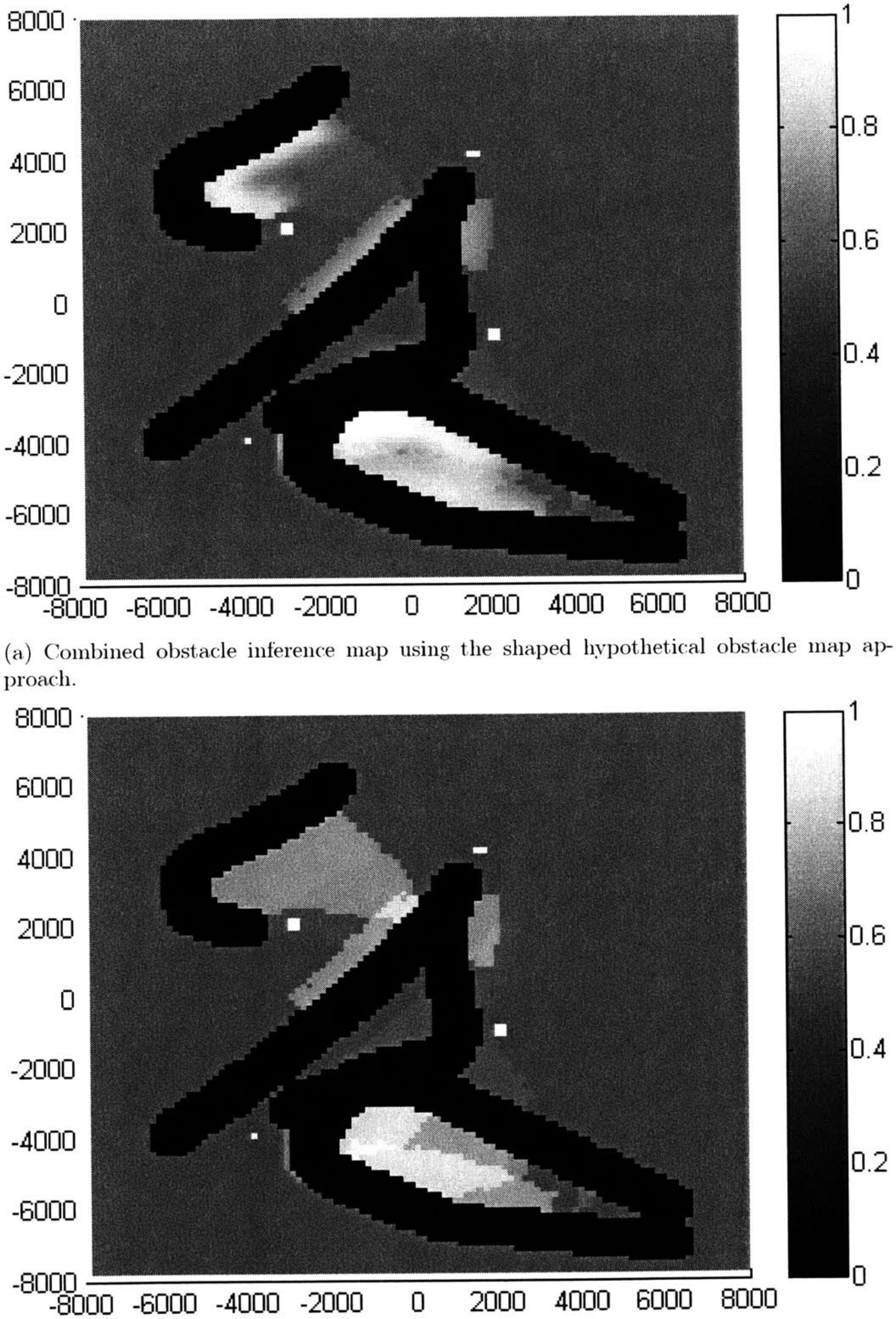


Figure 6-22: Four paths with obstacle zones.

inference maps. The *Cleared Path Only* map is an obstacle inference map where the contact's path is only used to set values of the cells along the contact's path (and a buffer area around it) to 0. The *Blank Map* is an obstacle inference map where every cell value has a constant value of 0.5, with the exception of the cells representing the area of known obstacles. The values of the cells representing those areas are set to 1. Since this map is fairly similar for all the simulation (listed in the columns) provided in Table 6.2, the changes in these error values are not dramatic. The error values for these two maps are used as benchmarks for comparing the error values calculated from the other obstacle inference maps. A graph of the error values is provided in Figure 6-25 and the legend is provided in Table 6.3. The error values for the blank map and cleared path only benchmarks are connect by a solid and dashed line respectively. From the error analysis graph, it can be seen that the error values for the generated obstacle inference maps generally fall between the two benchmark values. The most visible exception occurs in the case of Simulation 3 where the blanket method (using



(a) Combined obstacle inference map using the shaped hypothetical obstacle map approach.

(b) Combined obstacle inference map using the blanket hypothetical obstacle map approach (with $p = 0.75$).

Figure 6-23: Obstacle inference maps incorporating the analysis of four paths.

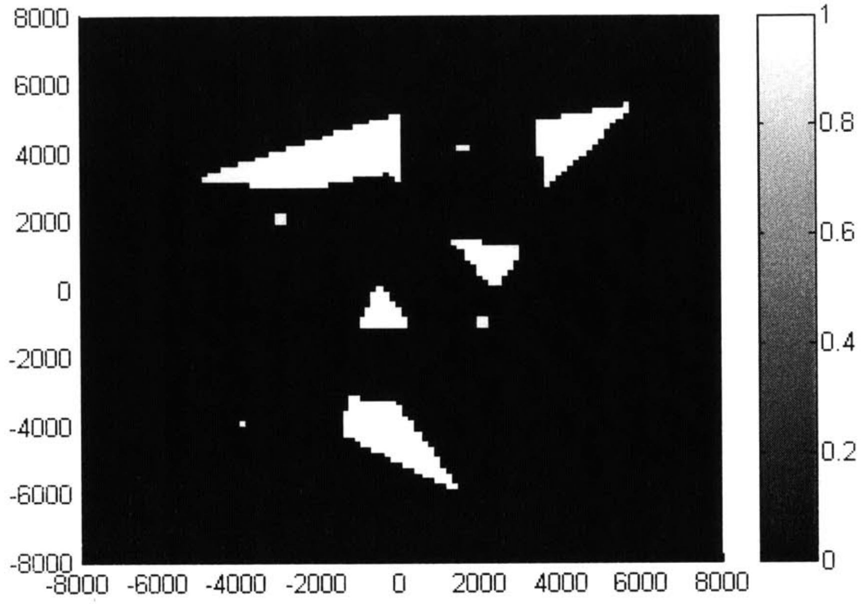


Figure 6-24: Actual obstacle map for a simulation including all four targets.

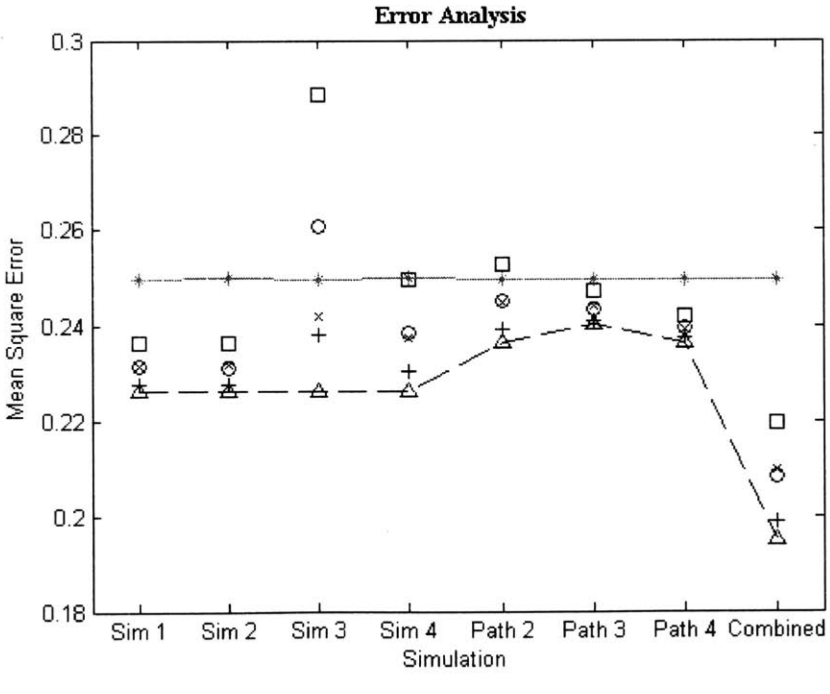


Figure 6-25: Error analysis graph with a 0.5 initial probability of occupancy (see Legend in Table 6.3).

Method	Symbol
Shaped	×
Blanket ($p = 0.75$)	○
Blanket ($p = 0.6$)	+
Blanket ($p = 0.9$)	□
Cleared Path Only	△
Blank Map	✳

Table 6.3: Legend for Figure 6-25—Error Analysis.

a p value of 0.75 and 0.9) exceed the error value for the blank map. The error analysis of the results for Simulation 3 indicates that it may be more beneficial not to infer the obstacles in the environment than use the blanket method with p values ≥ 0.75. The reason why the error values for these maps greatly exceed the error value for a blank map is because the observed path of the contact in Simulation 3 did not following the most direct route to any target. This violated the assumption made in Section 4.2 that the contact would pursue the most direct route to its target. The result is that the indirect actions of the contact were attributed to obstacles which produced a relatively inaccurate obstacle inference map.

It can also be seen from the error analysis graph that the error values for the cleared path only map were consistently lower than the values for the other maps. This indicates that all four methods used to infer the obstacles in the environment performed worse than simply clearing the location of the observed path from an inference map *with completely uncertain values* (0.5). Using a 0.5 initial probability of occupancy for each cell can also reflect the belief that the area is half-filled with obstacles. If a lower initial probability value of occupancy is used, the error should be less since it is reasonable to assume that most areas contain a small percentage of obstacles (< 0.5). Figure 6-26 shows the error values from using a 0.1 initial probability value of occupancy for each cell in the obstacle inference map. From this figure, it can be seen that the error values are lower than before (Figure 6-25) and all four inference maps performed worse than the blank and cleared path only maps. However, an obstacle inference map created by only decreasing the probability of

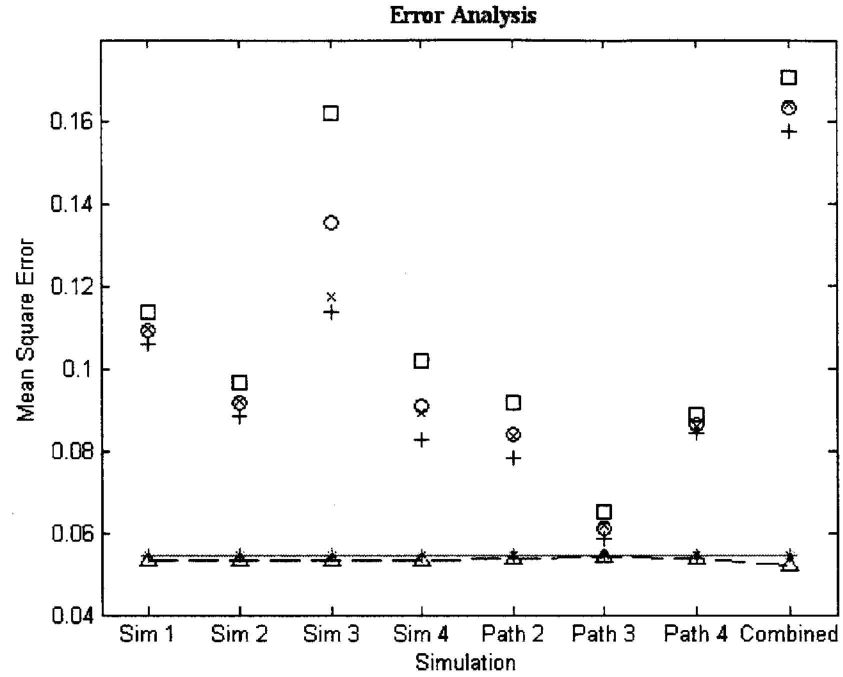


Figure 6-26: Error analysis graph with a 0.1 initial probability of occupancy (see Legend in Table 6.3).

cell values would not provide any information for indicating the possible locations of obstacles. This approach may be useful if a large number of vessels can be observed transiting through the area and if their tracks cover a large portion of the environment. In that case the observed paths would cover the environment fairly thoroughly (e.g., they would not all transit along the same channel). Using this approach, the location of obstacles can then be inferred as the areas represented by the cells which do not have a probability value of 0. However, since observing an adequate number of targets would most likely be time intensive, this may not be a viable option. In addition, many shallow waterways are controlled by some traffic scheme which may force the majority of the observed contacts to follow a similar path. In that case, even though the obstacle inference maps produced are considered less accurate (using the mean square error comparison) than the cleared path only map, they do provide an estimate of the possible locations of obstacles in the environment. The obstacle inference maps generated using the shaped and blanket approach (where $p = 0.6$) generally performed slight better than using the blanket approach with p values of 0.75 and 0.9. In addition, those methods were more robust in handing the indirect path of the contact in Simulation 3. In that simulation the map produced using the blanket approach (with $p = 0.6$) had a lower error value since the method does not increase the cell values as quickly as the other methods. The shaped method was able to produce a map with a low error value by only considerably increasing the probability value of selective cells.

6.5 Summary

The simulations presented in this chapter provided a number of obstacle inference maps produced by analyzing several observed contact paths through a common environment. Simulations 1–4 analyzed the same observed path using different combinations of four possible targets in the environment. These simulations demonstrated the relationship between the intent estimator and the obstacle detection algorithm. Three additional simulations were then provided, each analyzing a different path through the same environment (with all four targets). The final simulation then combined the observation paths from Simulation 1 and the latter three simulations to produce several obstacle inference maps, each illustrating the results from combining the analyses of four different observation paths.

An analysis of the error was done where the obstacle inference maps were compared to the actual map of the obstacles in the environment. The resulting mean square error value for each inference map was compared to two benchmarks values. The first is the error that would exist if the obstacle detection algorithm was not performed (producing a blank map), and the second is the error generated from only removing the belief of obstacles in the environment (producing a cleared path only map). The error analysis indicates that the obstacle inference methods produced a larger amount of error than the cleared path only method. However, even though the error was higher, the obstacle inference maps still provide an estimated belief of obstacles in the environment which the cleared path only map does not. In addition, the obstacle inference maps showed that the obstacle detection algorithm does provide some useful information (over a blank map) in the majority of the simulations. An exception was seen from simulation 3 using the blanket hypothetical obstacle map approach with a large value for p. From the error analysis graph (Figure 6-25) the error was higher than for a blank map. Such a comparison may indicate that the generated inference map produces a large number of false-positives and should therefore be used cautiously.

The performance of the obstacle detection algorithm was also shown to vary depending on the contact's relationship to the targets in the environment and the indirectness of the maneuvers made by the contact. The former may affect the ability of the obstacle detection algorithm to identify some obstacles in the environment based on the contact's path (as illustrated in Simulation 4), and the latter may significantly increase the area perceived to be occupied by obstacles (Simulation 3). Simulation 3 shows the effects of using the obstacle detection method presented without the assumption that the contact does not follow the most direct path to its target (which was made in Section 4.2).

Chapter 7

Conclusions

The objective of this thesis was to contribute to the development of the maritime reconnaissance and undersea search and survey signature capabilities for an Unmanned Undersea Vehicle (UUV). This chapter presents a summary of the contributions made by this thesis and provides suggestions for future research.

7.1 Thesis Contributions

This thesis provided an intent estimation algorithm that can assist a UUV in estimating the intent of the maritime vessels it is observing (*intent estimation*). This algorithm was then leveraged to infer the locations of possible obstacles in the environment (*obstacle detection*).

Chapter 2 presented a basic intent estimation model and an algorithm that creates probabilistic models for a selected domain of intents that an observed vessel (contact) may have. These models are then compared against the observed maneuvers of the contact to identify which intent model best explains the actions of the contact. This calculation was performed using Bayes' rule and the result is a belief state over the intent domain. This belief state provides a probabilistic belief value that the contact has a certain intent.

When the basic intent estimation model was developed (Chapter 2), a number of assumptions were made to manage the uncertainties concerning the contact and the environment. These assumption simplified the creation of the model but also restricted its use. Chapter 3 extended the intent estimation model. A number of the assumptions made were removed and the resulting uncertainties were modeled by the intent estimation algorithm. The result was an intent estimator which was able to function in a more dynamic and uncertain environment. The resulting belief state calculations were further improved by incorporating a retrospective calculation using a *forward-backward procedure*. The result was a post-processed belief state calculation which took into consideration the entire observation sequence made of the contact.

Chapter 4 presented a method for inferring the locations of possible obstacles in the environment using the belief states produced by the intent estimator. The obstacle detection algorithm analyzed the observed path of the contact to determine any indirect movements it may have taken to reach its projected destination. Such movements were assumed to be due to possible obstacle zones in the environment. These zones were identified and recorded in an obstacle inference map. The obstacle inference map is capable of being updated using the analysis of multiple observation paths (from different contacts) to provide an indication of the locations of possible obstacles.

Chapters 5 and 6 presented the results from testing the performance and capabilities of the intent estimation and obstacle detection algorithm. The simulations from Chapter 5 demonstrated the ability of the intent estimator to accurately determine the intent of an observed contact (which was independently steered by the user). The refinements made to the intent estimator in Chapter 3 were demonstrated and the ability of the intent estimator to function in a dynamic environment was also shown. That chapter also presented the effects on the performance of the intent estimator from modifications to the model's run-time parameters.

Chapter 6 provided the results from using the obstacle detection algorithm to analyze several contact paths transiting through a common environment. The results produced by the obstacle detection algorithm was shown to be heavily influenced by the inferred contact intents. An error analysis performed on the results generated from the simulations illustrated the capabilities and limitations of the algorithm. The results were shown to be useful but not extremely precise. This algorithm could be used to assist the UUV in identifying regions of interest to investigate if it only had a limited amount of resources to explore an area.

In conclusion, the intent estimation and obstacle detection algorithms presented in this thesis provide certain capabilities for a UUV to estimate the intent of the contacts it observes and later use that information to approximate the potential location of obstacles in the environment. These two capabilities can be used by the UUV to extract the necessary information from the data it collects to help determine its next course of action.

A UUV with these capabilities can be used by the military to monitor and gather information from hostile areas without requiring a large amount of oversight from human operators. It can be sent to observe an area and decide on its own the best way to use its resources. For example, the UUV can determine which vessels to follow and observe or which areas should be explored for possible obstacles. These capabilities would allow a UUV to be deployed to an unfamiliar region and have it determine its next course of action based on opportunities it finds from the data it collects.

7.2 Future Research

Future research in several key areas may provide additional capabilities for the intent estimation algorithm and improve the quality of the obstacle inference map.

7.2.1 Intent Estimation Improvements

The intent estimation algorithm currently models three possible contact behaviors (*follow, intercept,* and *approach*). All of these behaviors are focused on a physical target in the environment (i.e., another vessel or an object). By modeling a different class of behavior (i.e., one that does not focus on a defined object in the environment), a more representative state space of the contact's possible intent can be considered. For example, the intent estimator currently does not have a model of a *patrol* behavior. Therefore, if the observed contact was patrolling an area, the estimator would incorrectly identify its actions as some other intent and probably a varying set of intents due to repeated motions during the patrol.

The intent estimator presented also relies on having full knowledge of the potential targets in the environment such as buoys and other ships. If the contact was maneuvering with regards to an unknown target, such as a submarine, then the intent estimator would not be able to correctly interpret the contact's actions for two reasons. First, the contact's intent would not be in the intent state space for the estimator to consider. And second, the intent estimator would not be able to generate an intent model for that intent since it does not have any information concerning the intent target. Therefore, it may be beneficial to modify the algorithm to take into consideration that the contact may be focused on a target which is not detected by the UUV.

Another improvement that can be made to the intent estimator is to have the algorithm calculate and maintain a belief state over the variable representing the type of vessel that the contact may be (e.g., cargo ship, war ship, etc.). The estimator currently uses the same probably distribution over the modeled vessel types for determining the intent models. If the type of the contact can be identified, then the models can be made more precise since the capabilities of the contact will be known. In addition, the contact's vessel type can also be used to help determine the likelihood of an intent. For instance, a cargo ship is less likely to intend to intercept another ship than a military or police vessel is. The resulting algorithm could therefore produce a better estimation of the contact's intent.

7.2.2 Obstacle Detection Improvements

The current obstacle detection algorithm considered several approximation methods to translate the map cells representing a suspected obstacle zone into a probability value of occupancy. The resulting obstacle inference maps using each method were shown to contain some amount of error. This error would be reduced if a more accurate method was used to determine a probability value of occupancy for the cells in an obstacle zone. Multiple contact paths could also be analyzed together to determine the probability values for each cell. The current model assumes the observed paths are independent. Such an assumption simplifies the calculations in updating the obstacle inference maps but precludes the use of any information which may be available from the relationships among the paths. In addition, the obstacle inference procedure could be reiterated and the map refined with each path provided. One approach would be to use the Expectation-Maximization (EM) algorithm for occupancy estimation.

Several limitations of the obstacle detection algorithm were also visible thorough the development of the algorithm and the simulations. First, only one vessel type was considered for the contact. This approach relies on being provided the information of the contact's type. This dependency can be resolved by incorporating the consideration of multiple vessel types into the algorithm, or by determining the type of contact through the mentioned improvement for the intent estimator. Second, determining obstacle zones for a destination behind the contact (at its current heading), from Section 4.5.3, was difficult. Further research concerning this scenario may provide a better estimate of the possible locations of obstacles in the environment. Finally, the segmenting of the observed path into intent blocks, for determining the duration of a contact's estimated intent, may be improved. The algorithm currently creates these intent blocks by focusing on one predominant intent. A method for determining these blocks by using the entire intent belief state may produce a better representation of the duration for an estimated intent.

It was also visible from Simulation 3 (in Section 6.2.3) that the obstacle detection algorithm may produce an obstacle inference map with a greater degree of error if the contact's maneuvers are fairly indirect (non-ideal) in reaching its target. The consideration of non-ideal contact movements by the algorithm (by relaxing the assumption that the contact takes the most direct path possible) may result in a more accurate obstacle inference map.

Appendix A

Path Planner

A central component to the behaviors modeled in this thesis is a path planner. The determination of the initial position and destination of a behavior (as described in Section 2.3.2) contributes only part of the model. The actual path that the contact takes to move from its current position to its destination needs to be determined to complete the model. To perform this action, a path planner is required. An overview of the path planner used in this thesis is first presented followed by several example paths generated by the path planner. Finally, a description of the input values to the path planner, and the process by which those values were generated, are provided.

A.1 Overview

The expected path (intent model) is generated using the path planner from the Autonomous Minehunting and Mapping Technologies (AMMT) program [16]. This path planner implements the A* search algorithm [14, 10]. The basic operation of the path planner (from the standpoint of the current research) is as follows:

1. The user provides the path planner with a map of the environment as a fixed rectilinear grid. Each cell in the grid represents a small area in the environment and indicates with a binary value whether or not a vessel can safely traverse that area. Any obstacles in the environment will be represented by the corresponding

grid cell(s) being marked as non-traversable.

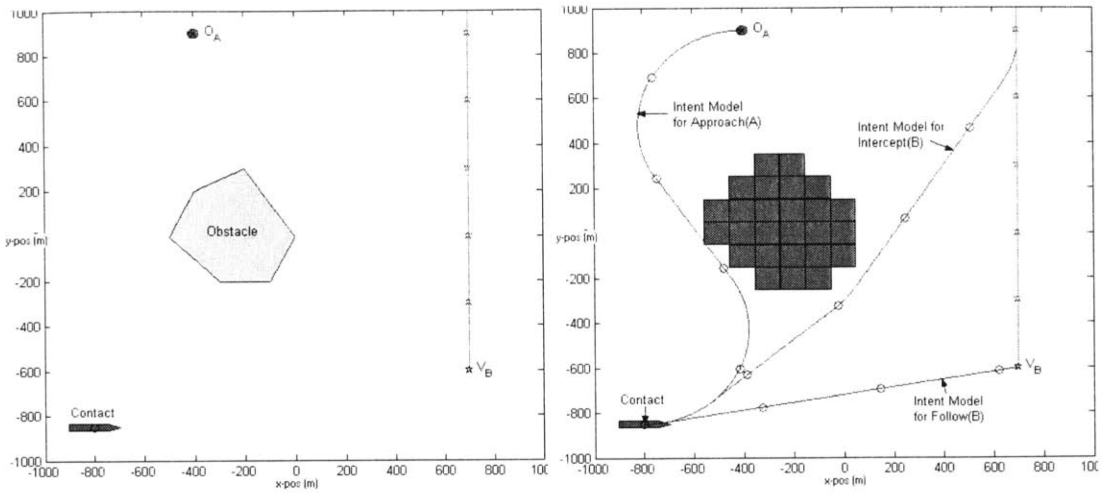
2. The user then provides the path planner with the vessel's current (x, y) position, speed, heading, and yaw rate along with the goal (x, y) position and heading range. Yaw rate is defined as the angular rate at which the heading of the vessel is changing and is equal to the speed of the vessel divided by the radius of the turn. The heading range is the desired range between which the user wants the vessel to be heading once it reaches the destination. This range will be referred to as the heading and represented by a single value with an assumed ±5° allowance on either side.

3. The path planner calculates a lowest cost path using the provided map and user inputs (from step 2) to move the vessel from its starting location and heading to its intended location and heading while avoiding any obstacles along the way. The path produced is comprised of a series of straight lines and constant radius and turns. Each turn can be represented by an arc of a circle with a radius equal to provided vessel's speed divided by the yaw rate.

A.2 Examples

Figure A-1 provides a basic illustration of the operation of the path planner. The environment represented in Figure A-1(a) consists of the contact being observed (labeled *Contact*), one stationary target (O_A), one moving target (V_B—represented by a '$\star$' symbol), and one obstacle. The line extending up from V_B is the projected path of the vessel assuming a constant heading North. The additional $\star$ symbols along the path are the projected future positions of the target at 60 second intervals (assuming a constant vessel speed).

Figure A-1(b) displays both the same environment from Figure A-1(a) using the view of the map provided to the path planner and three planned paths for the contact. The gridded map is able to roughly represent the obstacle as a group of non-traversable map cells. In this example, each map cell represents a square area in the



(a) Example environment.

(b) Grid map with planned paths.

Figure A-1: Path planner example.

	Approach(O_A)	**Intercept(V_B)**	**Follow(V_B)**
Current x-position (m)	-800	-800	-800
Current y-position (m)	-850	-850	-850
Current speed (mps)	8	8	8
Current heading (deg)	90	90	90
Current yaw rate (dps)	1.1	1.1	1.1
Goal x-position (m)	-400	700	700
Goal y-position (m)	900	815	-600
Goal heading (deg)	90	0	70

Table A.1: Input parameters for the path planner.

environment measuring 101 meters along each side. The three paths created by the path planner illustrate the behaviors listed in Section 2.3.2.

A.3 Input Parameters

The input parameters to the path planner for each one of the paths in Figure A-1(a) are provided in Table A.1. The x and y positions are in *meters* (m) using the coordinate system of the map. Speed is provided in *meters per second* (mps), and the heading is an absolute degree measurement (deg) where 0° is North and the values increase in a clockwise direction as in a normal compass rose. North is defined along the positive y-axis and East is along the positive x-axis. The yaw rate (speed divided by radius of turn) is provided in *degrees per second* (dps). The examples in Figure A-1(b) show a turning radius of 420 m to model a large ship. Using a speed of 8 mps, the yaw rate is calculated as follows

$$\text{yaw rate} = \frac{\text{speed}}{\text{radius of turn}} = \frac{8mps}{420m} \approx 1.1\ dps.$$

Modifying the speed or desired turning radius of the contact will affect the yaw rate, producing a different contact path. The units of the desired goal position and heading are the same as those for the contact.

The contact input values are rather straightforward. They represent the current location and heading of the vessel along with a user-defined speed and turn radius (which determines the yaw rate). The values for the desired goal position and heading are not as apparent in certain situations. These values will be determined to reflect the behaviors defined in Section 2.3.2. The methods used in this thesis to calculate these values are described as follows.

A.3.1 Approach Parameters

The goal position and heading of a contact with an *approach* behavior are determined by the location of the target (which is the contact's destination for such a behavior) and the heading that the contact needs to have (if any) once it reaches the target.

Since the target (which is stationary) for this behavior never move, the desired goal position provided to the path planner is simply the position of the target. The goal heading represent some external assumptions about the environment or contact goal—such as that the contact wants to enter a channel that begins at the target location. In that case, the goal heading would be the heading suitable for entering the channel. When the goal orientation of the contact is unconstrained by the environment, the goal heading can be arbitrarily determined. The following two methods were considered.

A.3.1.1 Shortest Path Method

The shortest path method (SPM) determines the heading that would produce the shortest path to get from the contact's current location and heading to its intended destination. This heading can be obtained by running the path planner over a set number of different ranges to cover the possible goal headings of [0°, 360°). For example, 36 paths can be generated using various goal headings (0°, 10°, ..., 350°) and the shortest path among the 36 would indicate the ideal heading.

A.3.1.2 True Bearing Method

Since the SPM requires iterating over a number of possible headings, the computational cost can increase dramatically as the number of paths that needs to be calculated increases. Therefore, a less computationally intensive way of producing a reasonable goal heading is needed. A viable option is to determine the absolute bearing from the contact to the target and use that value as the goal heading. This method will be referred to as the true bearing method (TBM). The TBM would work well if the environment does not contain any major obstacles and the separation between the contact and the target is reasonably large.

Since the TBM is capable of producing reasonable results with less computational cost than the SPM, the former method is implemented. A combination of both methods could be considered (e.g., use the TBM when the distance between the contact and target is great and obstacles are few, switching over to the SPM when the two are closer together).

A.3.2 Intercept Parameters

The goal position and heading for an *intercept* behavior are determined by calculating the values which would bring the contact to a certain location and heading that the target is expected to have at some future time.

```
INTERCEPT-PATH(contact, target)
max_try ← 10
try_future_time ← 1
allowable_time_window ← 10
for i ← 1 to max_try
do target_future ← MOVE-TARGET(target, try_future_time)
   goal.x ← target_future.x
   goal.y ← target_future.y
   goal.heading ← target_future.heading
   path ← GENERATE-PATH(contact, goal)
   if path not created
     then error "invalid intent"
     else path_duration ← CALCULATE-PATH-DURATION(path)
          time_difference ← |path_duration − try_future_time|
          if time_difference ≤ allowable_time_window
            then return path
            else average_value ← (try_future_time + path_duration)/2
                 try_future_time ← ROUND(average_value)
error "invalid intent"
```

Figure A-2: Algorithm for determining the intercept goal position.

Since it is reasonable to assume that the contact intends to interact with the target after it has intercepted it, the most desirable goal heading for the contact when it reaches the target would be in the direction the target is heading. Therefore, the goal heading will be the target's heading. The goal position is calculated as the location along the target's projected path where the two vessels' paths will intersect given the target vessel's speed and the contact's desired heading. The method used to determine this position was to run the path planer with various points along the target's projected path as the input for the goal position until the time it took the contact to reach that point (as determined by the generated path) was within a user-defined window of when the target will be at that location. A window of 10 seconds is used. This algorithm is provided in Figure A-2. The goal position and heading parameters are now defined.

Situations can occur when the contact is unable to intercept another vessel (e.g., the other vessel is traveling directly away from the contact at a speed greater than the contact's maximum speed). In such situations, the intent itself will be invalid and removed from the intent state space. A further discussion concerning such scenarios is provided in Section 3.2.4.2.

A.3.3 Follow Parameters

The goal position and heading for a *follow* behavior are determined by calculating the values which would bring the contact to the target's current position. These values are determined in the same way as the approach parameters (from Section A.3.1) with the exception that the target's current location is the point used for the contact's goal position.

The path planner does not deal with moving obstacles. Therefore, this work assumes that path crossings do not result in collisions.

[This page intentionally left blank]

Appendix B

Conditional Probability Distribution for $\mathcal{P}$

The Conditional Probability Distribution (CPD) tables in this appendix were arbitrarily created to reflect the probability that a contact would use a certain speed ($\mathcal{S}$) and turn radius ($\mathcal{R}$) as parameters for creating a path ($\mathcal{P}$) to perform a given behavior. The probability distribution across the possible $\mathcal{SR}$ combinations are dependent on the intent behavior and the relationship between the contact and the target ($\mathcal{DB}$)—as determined by $\mathcal{I}_u$ and $\mathcal{O}_{u-1}$. The CPD for two of the behaviors used in this thesis is provided in this appendix.

	$d_C b_F$	$d_C b_S$	$d_C b_A$	$d_I b_F$	$d_I b_S$	$d_I b_A$	$d_F b_F$	$d_F b_S$	$d_F b_A$
$P(\mathcal{P}_{i,s_F r_S}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.19	0.33
$P(\mathcal{P}_{i,s_M r_S}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.25	0.25	0.01	0.18	0.33	0.01	0.18	0.25
$P(\mathcal{P}_{i,s_S r_S}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.16	0.25	0.01	0.19	0.19	0.01	0.01	0.01
$P(\mathcal{P}_{i,s_F r_M}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.01	0.01	0.01	0.19	0.33	0.19
$P(\mathcal{P}_{i,s_M r_M}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.18	0.25	0.20	0.35	0.33	0.25	0.18	0.25	0.18
$P(\mathcal{P}_{i,s_S r_M}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.25	0.16	0.25	0.10	0.25	0.18	0.01	0.01	0.01
$P(\mathcal{P}_{i,s_F r_L}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.01	0.01	0.01	0.33	0.01	0.01
$P(\mathcal{P}_{i,s_M r_L}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.19	0.05	0.01	0.40	0.01	0.01	0.25	0.01	0.01
$P(\mathcal{P}_{i,s_S r_L}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.33	0.10	0.01	0.10	0.01	0.01	0.01	0.01	0.01

Table B.1: Probability distribution table for $P(\mathcal{P}_{i,\mathcal{SR}}|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$ where $\dot{\mathbf{t}}_i$ reflects a *follow* behavior.

	$d_C b_F$	$d_C b_S$	$d_C b_A$	$d_I b_F$	$d_I b_S$	$d_I b_A$	$d_F b_F$	$d_F b_S$	$d_F b_A$
$P(\mathcal{P}_{i,s_F r_S}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.10	0.20	0.20	0.07	0.25	0.25
$P(\mathcal{P}_{i,s_M r_S}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.20	0.19	0.19	0.15	0.20	0.20	0.10	0.33	0.33
$P(\mathcal{P}_{i,s_S r_S}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.10	0.33	0.33	0.01	0.01	0.01	0.01	0.01	0.01
$P(\mathcal{P}_{i,s_F r_M}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.12	0.15	0.15	0.20	0.18	0.18
$P(\mathcal{P}_{i,s_M r_M}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.27	0.18	0.18	0.20	0.20	0.20	0.20	0.19	0.19
$P(\mathcal{P}_{i,s_S r_M}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.10	0.25	0.25	0.01	0.01	0.01	0.01	0.01	0.01
$P(\mathcal{P}_{i,s_F r_L}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.01	0.01	0.01	0.20	0.10	0.10	0.20	0.01	0.01
$P(\mathcal{P}_{i,s_M r_L}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.20	0.01	0.01	0.20	0.12	0.12	0.20	0.01	0.01
$P(\mathcal{P}_{i,s_S r_L}\|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$	0.10	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Table B.2: Probability distribution table for $P(\mathcal{P}_{i,\mathcal{SR}}|\dot{\mathbf{t}}_i, \mathcal{O}_{u-1})$ where $\dot{\mathbf{t}}_i$ reflects an *intercept* behavior.

Appendix C

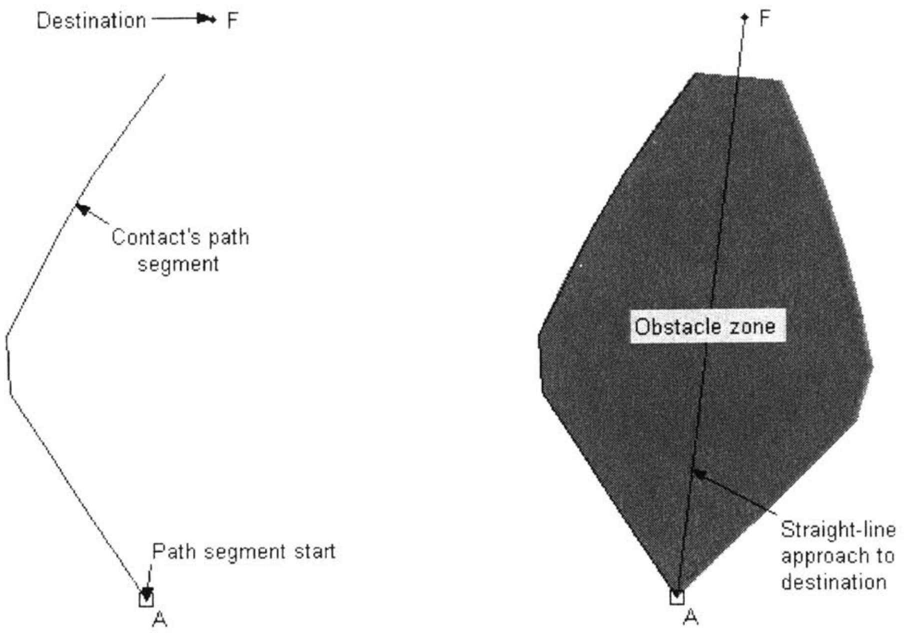
Shaped Hypothetical Obstacle Map

The motivation for creating a hypothetical obstacle map is to provide a tractable method for establishing a probability of occupancy for the cells in an obstacle zone. The approach presented here generates a hypothetical obstacle map using the assumption that obstacles are more likely to be present in the area of the zone that is relatively close to, but not directly next to, the contact's path. Therefore, a contour view of the belief for the cells in the obstacle zone would follow a shape that is defined by the contact's path and will be referred to as a shaped hypothetical obstacle map. The probability value for a cell (i.e., the cells' value) will also be referred to as the probability of occupancy.

This section will first present the general method developed to create the shaped hypothetical obstacle map. A special consideration to the general method is then discussed. Next, the method used to assign the probability values for each cell in the obstacle zone is described. Finally, the parameters used for generating the probability values for the shaped hypothetical obstacle maps in this thesis are provided.

C.1 General Approach

The obstacle zone created from the observed path and destination (F) illustrated in Figure C-1(a) is shown in Figure C-1(b). This zone is represented by a number of



(a) Path segment.

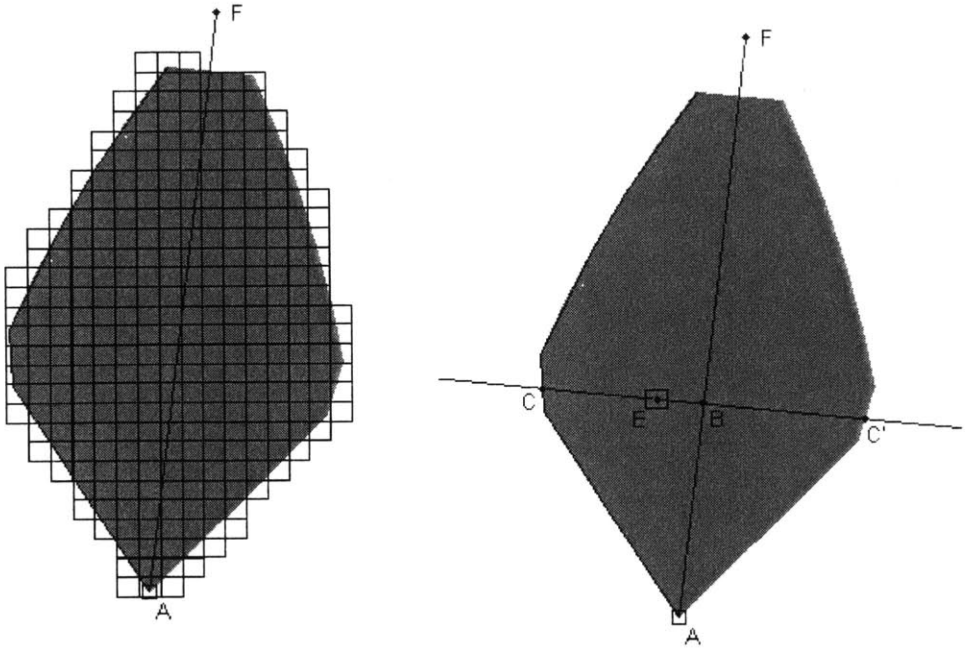
(b) Derived obstacle zone.

Figure C-1: Path segment analysis.

grid cells in the hypothetical obstacle map as shown in Figure C-2(a). For each of these grid cells, the following method is used to assign it a probability that the area it represents in the environment is occupied.

The position of each cell will be represented by a point located at its center. Figure C-2(b) considers a single cell (represented by point E) from Figure C-2(a). A line perpendicular to the line $\overline{AF}$ is drawn through E. The intersection of these two lines is identified in the figure by point B. The intersection of $\overline{EB}$ with the path segment is labelled C. A one-dimensional view of the line connecting points E, B, and C is shown in shown in Figure C-3(a). On this line, point C represent the left boundary of the obstacle zone and point C' represents the boundary on the right.

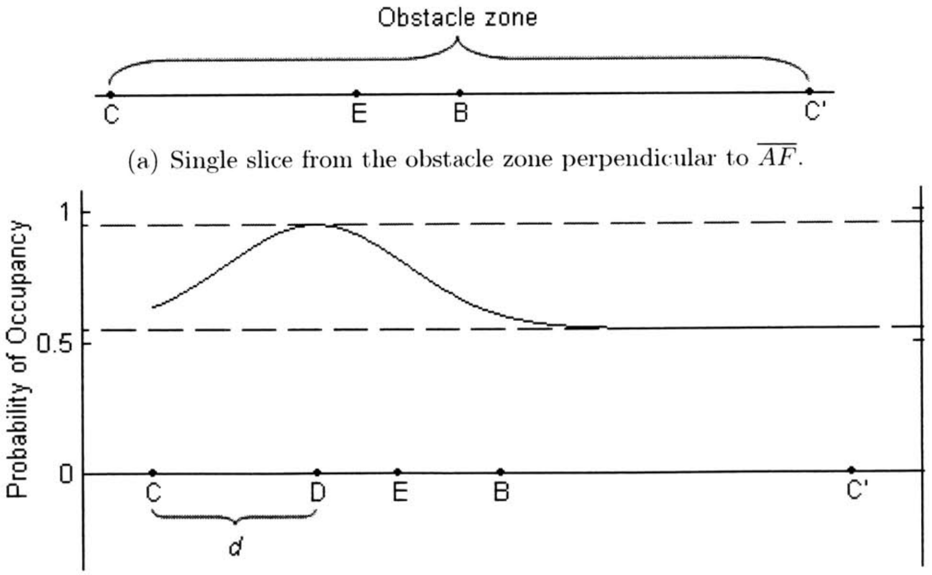
The area of the obstacle zone with the highest probability of containing an obstacle is identified by a user-defined distance from the contact's path. This approach "shapes"



(a) Gridded obstacle zone.

(b) Arbitrary cell E in grid.

Figure C-2: Obstacle zone cells.



(a) Single slice from the obstacle zone perpendicular to $\overline{AF}$.

(b) Probability of occupancy.

Figure C-3: Probability determination for point E.

the area of highest probability by using the path of the contact and allows the area to be placed at a distance from the path. This distance will be represented by d and the corresponding peak is indicated by point D. As the distance of a cell from this point increases, there is still the belief that the location being considered contains an obstacle (as long as it is still within the obstacle zone bounded by points C and C') but the probability decreases significantly at first and then virtually levels off. This relationship is illustrated in Figure C-3(b). The maximum probability of occupancy (which will be described in Section C.4) is represented by H (high) and is located at point D. The range of the possible value of H is

$$L \leq H \leq 1,$$

where L (low) is the minimal assigned probability of occupancy. The range for L is

$$0.5 < L \leq H.$$

The value of L must be above 0.5 since anything at or below that value would indicate respectively that its probability of occupancy is unknown or that the space represented by the cell is more likely empty than occupied.

C.2 Small Path Deviations

If the distance from point B to point C is less than d, then the contact did not travel too far off a direct approach course to the destination. In such a situation, point D would then be located at the center of the obstacle zone (which would be at the same location as point B). In addition, since the path deviation was not that large, the maximum probability of an obstacle being present is decreased. The modified maximum probability value is represented by H' and decreases linearly with the decrease in distance between points C and D. This relationship is illustrated in Figure C-4 and represented by the equation

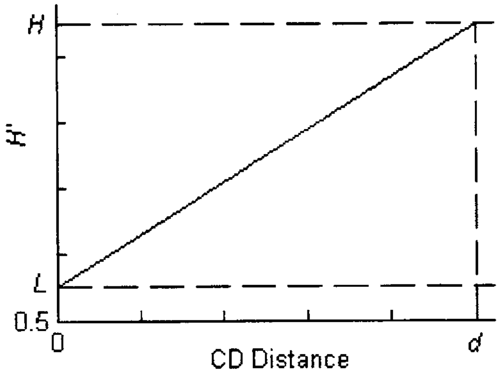


Figure C-4: Modified maximum probability.

$$H' = L + \frac{H - L}{d} \cdot |C - D|.$$

C.3 Probability Function

The function used to adjust the probability of occupancy according to the distance of the cell's location (A) from the area of highest probability (D) is as follows

$$f(A) = \frac{N(A, D, \sigma)}{N(D, D, \sigma)} * (H - L) + L,$$

where the function $N(x, D, \sigma)$ is the normal distribution centered on D with the standard deviation σ and evaluated at x. The normal distribution was used to provide the probability drop-off effect desired for illustrating a decreasing probability as the distance from point D increases. The normal distribution value at A is first normalized by the normal distribution value at D. This value scales the probability range defined by H and L. In the scenario where

$$d > |B - C|,$$

from Section C.2, the maximum probability value (H) is replaced by H'.

Variable	Value
d	2 $*$ vessel's minimum turning radius
H	0.95
L	0.55
σ	vessel's minimum turning radius

Table C.1: Variable values.

C.4 Variable Values

The values used in this thesis for the variables presented in this appendix are provided in Table C.1. The distance from the contact's path with the highest probability of containing an obstacle (d) was chosen as being twice the minimum turning radius of the contact. This value was selected since it would place the obstacle reasonably close to the contact's path (at one minimum turning diameter away) without making it too close, which would break into the contact's safety zone (this was assumed to be one minimum turning radius from Section 4.2). The values for H and L were selected to represent (with a buffer of 0.05) the maximum and minimum probability values respectively for the probability range to indicate that the area within the obstacle zone is actually occupied (0.5,1]. The standard deviation (σ) value used in the normal distribution calculations was chosen to represent a significant decrease in a cell's occupancy probability after one minimum turning radius from the highest probability point (D). This is the distance from point D where the contact's safety zone begins.

Bibliography

[1] B. Barksdale. Boeing unmanned combat aircraft makes aviation history. *Boeing News Release*, April 18, 2004.
http://www.boeing.com/news/releases/2004/q2/nr_040418m.html [04-May-2004].

[2] D. P. Bertsekas and J. N. Tsitsiklis. *Introduction to Probability.* Athena Scientific, Belmont, MA, 2002.

[3] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267, 2001.

[4] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. *In Proceedings of IEEE CVPR97*, 1996.

[5] R. Draim. System specification for the auxiliary dry cargo ship, T-ADC(X). Technical report, Naval Sea Systems Command, SEA 03, Arlington, VA, March 1998. http://www.fas.org/man/dod-101/sys/ship/docs/adcxspec.htm [03-Apr-2004].

[6] P. Dunn. The Navy Unmanned Undersea Vehicle (UUV) Master Plan. In *Proceedings of the International Unmanned Undersea Vehicle Symposium*, Newport, RI, April 2000.

[7] R. R. Hobbs. *Marine Navigation: Piloting and Celestial and Electronic Navigation.* U.S. Naval Institute Press, Annapolis, MD, 4th edition, 1998.

[8] F. V. Jensen. *Bayesian networks and decision graphs.* Springer Verlag, NY, 2001.

[9] N. Johnson and D. Hogg. Learning the distribution of object trajectories for event recognition. *Image and Vision Computing*, 14:609–615, 1996.

[10] S. D. McKeever. Path planning for an autonomous vehicle. Master's thesis, Massachusetts Institute of Technology, June 2000. Draper Laboratory Technical Report T-1372.

[11] A. P. Mierisch. Situational awareness for a Navy unmanned undersea vehicle. Master's thesis, Massachusetts Institute of Technology, June 2003. Draper Laboratory Technical Report T-1449.

[12] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9(2):61–74, Summer 1988.

[13] R. R. Murphy. *Introduction to AI Robotics.* The MIT Press, Cambridge, MA, 2000.

[14] J. Pearl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving.* Addison-Wesley Publishing Company, Reading, MA, 1984.

[15] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, February 1989.

[16] M. Ricard. Mission planning for an autonomous undersea vehicle: Design and results. In *Proceedings of the Technology and the Mine Problem Symposium*, Monterey, CA, November 1996.

[17] J. Risen and J. Miller. Threats and responses: Hunt for suspects; C.I.A. is reported to kill a leader of Qaeda in Yemen. *The New York Times*, November 5, 2002. Late Edition-Final, Section A, Page 1, Column 6.

[18] S. J. Russell. *Artificial intelligence: A modern approach.* Prentice Hall, Upper Saddle River, NJ, 2003.

[19] H. Shatkay and L. P. Kaelbling. Heading in the right direction. In *Proceedings 15th International Conference on Machine Learning*, San Francisco, CA, 1998.

4115 - 10