2009-06-30

# Radio Determination on Mini-UAV Platforms: Tracking and Locating Radio Transmitters

Braden Russell Huber
*Brigham Young University - Provo*

# RADIO DETERMINATION ON MINI-UAV PLATFORMS: TRACKING AND LOCATING RADIO TRANSMITTERS

by

Braden R. Huber

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computer Science

Brigham Young University

August 2009

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a thesis submitted by

Braden R. Huber

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

| | |
|---|---|
| Date | Michael A. Goodrich, Chair |
| Date | Bryan S. Morse |
| Date | Scott Woodfield |

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the thesis of Braden R. Huber in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

| | |
|---|---|
| Date | Michael A. Goodrich<br>Chair, Graduate Committee |
| Accepted for the Department | Kent E. Seamons<br>Graduate Coordinator |
| Accepted for the College | Thomas W. Sederberg<br>Associate Dean, College of Physical and Mathematical Sciences |

ABSTRACT

RADIO DETERMINATION ON MINI-UAV PLATFORMS:
TRACKING AND LOCATING RADIO TRANSMITTERS

Braden R. Huber
Department of Computer Science
Master of Science

Aircraft in the US are equipped with Emergency Locator Transmitters (ELTs). In emergency situations these beacons are activated, providing a radio signal that can be used to locate the aircraft. Recent developments in UAV technologies have enabled mini-UAVs (5-foot wingspan) to possess a high level of autonomy. Due to the small size of these aircraft they are human-packable and can be easily transported and deployed in the field. Using a custom-built Radio Direction Finder, we gathered readings from a known transmitter and used them to compare various Bayesian reasoning-based filtering algorithms. Using a custom-developed simulator, we were able to test and evaluate filtering and control methods. In most non-trivial conditions we found that the Sequential Importance Resampling (SIR) Particle Filter worked best. The filtering and control algorithms presented can be extended to other problems that involve UAV control and tracking with noisy non-linear sensor behavior.

# ACKNOWLEDGMENTS

Thanks to my wife for supporting me and giving me up on some late nights. Thanks to Mike Goodrich for his advice and for the fun adventures in the Field on Search and Rescue related work. Thanks to my parents for always supporting my Scientific Endeavors. Thanks to fellow researchers in the Lab for keeping it real.

# Contents

# List of Figures

# Chapter 1

## Introduction

When aircraft experience jarring landings or crashes, their Emergency Locator Transmitter (ELT) units are automatically activated. We endeavor to show that a miniature UAV (5 foot wingspan or less) equipped with a sensor can localize a radio beacon. Mini-UAVs are an ideal platform because of their low cost, size, and ease of operation. We employ Bayesian methods to deal with uncertainty and noise. Research work in this area can be generalized to other UAV-based localization problems such as locating avalanche victims or lost children.

Federal Standard 1037C [8] issued by the General Services Administration provides a glossary of telecommunication terms. This glossary defines radiodetermination as "The determination of the position, velocity and/or other characteristics of an object, or the obtaining of information relating to these parameters, by means of the propagation properties of radio waves." Radiolocation is a type of radiodetermination relating to position that is mainly accomplished through passive means.

Applications for different types of radiodetermination are diverse and far reaching. This work focuses mainly on transmitter localization; the work is therefore applicable to ELT searches, fox hunts, enemy localization, avalanche victims, and rescuing lost individuals who have beacons. ELT searches are for downed aircraft as mentioned above. Fox hunts are events held by amateur radio operators and involve homing in on a particular transmitter. Enemy localization broadly refers to the tracking and homing of any enemy transmission. Backcountry adventurers often carry radio trans-

mitting equipment with them for finding buried individuals in avalanches. Individuals at risk for being lost such as young boy scouts could be encouraged to carry a personal beacon so that if they become lost they can be located easily.

Radio Direction Finding (RDF) is a type of radiolocation that aims to locate the transmitter by several successive measurements that yield the angle to the transmitter. There are several types of sensor systems that can be used to approach this problem. Each system has relative advantages for specific applications. Details for how these systems are designed and used can be found from Moell [24] and are described in Section 2. Initial research indicates that for airborne RDF, homing DF systems are the dominant choice [24]. This is because aircraft can easily change course to (almost any) arbitrary heading. Also airborne platforms escape much of the interference due to multipath problems.

In a typical RDF search using Homing DF equipment: a person would take a reading and use it to travel in the direction of the transmitter. As the person moves he or she will continue to take readings to make sure he or she is on the correct path and make necessary course adjustments. Homing DF equipment offers several advantages: it can be tuned for a wide frequency range, the unit size can be very small (critical for a mini-UAV), the sensor is minimally affected by variations in signal strength, and it is easy for the user to learn to operate [24]. The primary disadvantages of this ground-based process are *(a)* the receiver is susceptible to multipath from reflectors near the antenna, and *(b)* Line of Sight (LoS) with the transmitter can be obstructed by obstacles. Both of these problems are significantly reduced when used on aircraft platforms from altitude.

Currently, Utah County Search and Rescue may be called upon to assist in several ELT searches per month. Over the period of September 2005 to January 2006 there were approximately 9 ELT calls in which Utah County Search and Rescue participated. More than 95% [9] of all ELT searches nationwide are false alarms where

| Altitude Above Ground Level (ft) | Distance in Nautical Miles (nm) |
|---|---|
| 1,500 | 18 |
| 2,000 | 21 |
| 3,000 | 30 |
| 4,000 | 35 |
| 6,000 | 40 |
| 8,000 | 65 |
| 10,000 | 80 |

Table 1.1: Signal reception range for different altitudes [7].

the ELT was activated upon a hard landing and the plane, when located, is found to be in a hangar at the airport where the pilot failed to deactivate the unit after landing. Utah County is no exception to this national statistic, with most of their searches resulting from false alarms. Aside from these airport activations, most of Utah County is sparsely populated and represents a Wilderness Search and Rescue (WiSAR) type of task.

Utah County procedures for ELT searches involve several volunteers working in cooperation. One of the main problems is that direction readings must be taken from relatively high altitudes (see Table 1) to overcome multipath and occlusion problems presented by hills, mountains, and buildings. This means that some volunteers must drive high up on the mountains (Wasatch Front) in order to get an accurate reading. Due to the nature of RDF, one reading is not enough. At least two readings from different locations must be taken to begin triangulation. More readings are necessary to confidently predict where the transmitter is. This represents a significant task as volunteers must drive up mountains multiple times to obtain the readings.

These methods result in a *time* consuming process to *(a)* determine if it is a false alarm or a genuine emergency and *(b)* actually locate the downed aircraft. In the case of a real emergency rescue efforts are very time sensitive. Successful rescue efforts correlate closely with the amount of time it takes to reach the crash site. Therefore, reducing the amount of time to locate the aircraft should be very beneficial. Reducing

the time to locate the aircraft also has the advantageous side-effect of reducing total man hours (and therefore the cost) to complete the search and rescue.

Two significant problems in automating RDF on a UAV platform are estimation and control. Estimation of the transmitter position is difficult because of noisy sensor readings (detailed in Section 3). The accuracy of the estimation is also dependent on UAV position, and therefore the efficiency of any search relies on the control method employed. Control is difficult because we are searching through a large action space to try and obtain the estimate as efficiently as possible.

We have implemented particle filtering and the unscented Kalman filter (UKF) algorithms to deal with the noisy sensor readings. Leveraging Bayesian reasoning, we can obtain an accurate transmitter location estimate. Filtering is discussed in Section 3.2.

We have also developed a path-planning controller to expedite search operations. We do not make the claim of optimality with this controller but instead hope to show performance increase over manned search techniques. Development of this controller also represents a proof-of-concept on the mini-UAV platforms. Control approaches are detailed in Section 3.3.

## 1.1 Thesis Statement

Mini-UAVs with radio sensors can save time and money in ELT radiolocation compared to traditional approaches. Particle filtering can be employed to compensate for noisy sensor readings, and intelligent path planning increases the accuracy and speed of the search.

# Chapter 2

## Related Literature

### 2.1 UAV History

Use of Unmanned aerial vehicles (UAVs) began early in the history of aviation. During World War II target drones were used to train anti-aircraft gunners without risking human life [26]. These initial UAVs were simply remotely piloted full-size aircraft. Recent innovations in miniaturization and sensor technologies have allowed the autonomy and capabilites of UAVs to greatly increase.

Mini-UAVs are human-packable units. Mini-UAVs developed and flown by the BYU MAGGIC Lab [34] are typically five feet or smaller in wingspan. Mini-UAVs offer several advantages over their larger counterparts. They can fit inside almost any vehicle and therefore do not require special transportation equipment. The aircraft and support systems can be carried to remote sites by a few people. No runway is required because the UAVs are hand-launched and belly-landed.

### 2.2 Radio Determination

Radiodetermination has a rich historical background. Some of its first applications were during WWII. Radar was used during the Battle of Britain by the Royal Air Force as an early warning system for approaching German aircraft. The German bombers used directional radio transmitters to paint an X over targets in Britain. When the German bombers reached the specified location, the radio would tell them

to release the bombs [38]. The Global Positioning System (GPS) is an excellent example of modern radiolocation.

For many years civilian and military aircraft alike have carried various types of homing beacons onboard. In the case of emergency, these beacons can be activated. Through radiolocation, rescuers are able to find the downed aircraft and perform rescue or recovery operations. In the US today, aircraft are equipped with ELTs on a standard basis. These devices operate on international emergency frequencies of 101.5 MHz, 243.0 MHz, and 406.025 MHz.

## 2.3 SARSAT

SARSAT satellites have been in operation in recent years and are able to accurately locate ELT beacon transmissions [22]. These satellites have largely eliminated many of the difficulties experienced in lost aircraft searches in the past. However propagation time between when the beacons are activated and when searchers recieve coordinates is still significant. While we have geared our test toward using an ELT type scenario we believe it also extends to any type of beacon homing that the SARSAT system is not currently equipped to handle.

## 2.4 Flight Control

UAVs are required to exercise flight control either autonomously or under human control. Jonsson [18] defines two separate tasks for flight control: aviation and navigation. Note that Jonsson extends work by Abbott [1], but Abbott himself does not actually define the tasks of aviation and navigation. Quoting from [18]:

> Each of these categories [aviation and navigation] may be defined by using a modified series of definitions described in Abbott (1993). Aviation may be defined as the process "of adjusting or maintaining the flight

> path, attitude, and speed of the [aircraft] relative to flight guidance requirements." Navigation may be defined as the process of "developing the desired plan of flight. . . and monitoring its progress."

In the context of a UAV, aviation represents the actual second-to-second flying of the UAV. This can include maintaining a constant altitude, holding an attitude, keeping sufficient airspeed, etc. Aviation can be accomplished either manually or autonomously with the autopilot mentioned below. Navigation is often accomplished through the development of waypoints that can be followed by the autopilot. Waypoints can be produced by human operators or algorithmically from some meaningful sensor inputs.

## 2.5 Filtering and Control

A great deal of work has been done regarding the problem of obtaining filtered estimates of state variables from noisy sensor measurements. Classic approaches such as the Kalman Filter [23, 36] do not perform well in non-linear, non-gaussian environments, which are often present in tracking and target motion analysis (TMA) applications. Adaptations such as the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF), as well as working in polar coordinates [2], improve performance but are often not sufficient. Recent work involving Sequential Monte Carlo Methods [10] has shown that Particle Filters are effective at handling non-linear system and sensor functions as well as non-Gaussian noise distributions [3, 13, 14, 16].

The problem of transmitter localization from a UAV platform is very similar to the problem of Bearing-Only Tracking (BOT), also refered to as Angle-Only Tracking. In fact the UAV application with the sensor described above is a special case of BOT because bearings are only available when the UAV is pointed directly at the transmitter. BOT has been looked at by several researchers including [4, 10, 13, 14,

19–21]. The idea of changing the coordinate system and employing particle filtering has been discussed by papers such as [6].

The BOT problem is difficult because, as Tremois [32] points out, target estimation is dependent on observer control. Many other estimation problems rely on the separation principle where control and estimation are independent. Researchers [17, 25, 27, 32] have tried to pose this correlation in terms of an optimization problem in order to try and find the optimal observer control algorithms. Key tools in the evaluation are the Fisher Information Matrix (FIM) and Cramer-Rao Lower Bound (CRLB). The Fisher Information quantifies the information that an observable variable gives about an unobservable variable. The observer wants to obtain sensor readings that maximize the amount of information contributed with each new reading.

According to our knowledge, not a lot of research has been done with the RDF problem from UAV platforms, especially mini-UAVs. Frew et al. [12] discuss using a team of UAVs that are part of a heterogeneous sensor net to locate transmitters. This discussion involves the UAVs measuring signal strength and coordinated communication to localize transmitters. However signal strength in natural environments is not always closely correlated with distance.

Pine et al. [28] simulate two UAVs searching for a maritime beacon in a BOT scenario. Control of the UAVs heading is determined by minimizing a cost function at each time step. Several cost functions are evaluated and show how the area of uncertainty about the transmitters location is reduced. The resulting flight paths are not directly to the transmitters but instead resemble arcs and angular legs that allow accurate localization of the transmitter before the transmitter is intercepted.

# Chapter 3

## Methods

Manned search aircraft can be very fast at homing in on beacons. However, the cost of operating these vehicles during search operations is significant. Additionally there can be troublesome "scramble" time associated with getting these aircraft into the air and searching. Elimination of this type of searching could represent significant *monetary* savings to county search and rescue budgets. We have designed, created, simulated, and tested various approaches to mini-UAV based RDF. The focus is on filtering methods. We *(a)* identify hardware, *(b)* identify metrics to evaluate and understand our techniques and performance, *(c)* implement filtering techniques on real data, and *(d)* implement control techniques for automated radiolocation in simulation.

## 3.1 System Architecture

### 3.1.1 Hardware

#### Aircraft

Our airframe is a 5-foot-wingspan flying wing UAV similar to those already used in the BYU MAGICC Lab. This UAV uses the Kestrel Autopilot (see Section 3.1.1). The autopilot's sensor suite includes a 3-axis accelerometer, a 3-axis gyro, a GPS reciever, pressure sensors [static and dynamic], and a video camera.


Figure 3.1: The Virtual Cockpit program. This constitutes BYU groundstation control.

### Radio Equipment

We utilized a dual-dipole homing DF system in our experiments because of our small platform and airborne search setting. Our work includes building the RDF sensor and UAV. See Section 3.1.3.

## 3.1.2 Software

We used a small set of software programs to support our search task. Virtual Cockpit (Figure 3.1) is a ground control program used for the actual UAV operation. We developed a graphical UAV simulation environment with radio simulation support called RAH-DEE-OH to test filtering and path planning methods. This simulator models radio propagation including multipath using a ray-tracing engine.

The Kestrel autopilot developed by BYU [34] offers a high level of autonomy. The autopilot is typically controlled via a Ground Station which consists of a laptop, radio modem, and video relay equipment. The task of aviation can be performed in a

Figure 3.2: The Little L-Per from L-Tronics. This is an example of a dual-dipole type homing RDF unit.

number of different modes and does not require user control. This allows the operator to focus on the task of navigation. This means that the operator can assign waypoints which the autopilot will automatically fly to. The operator is then free to monitor progress and plan future actions. The autopilot also offers heading, roll, and pitch control. Ground station management is typically done through the Virtual Cockpit program (Figure 3.1) also developed at BYU [34]. The UAV offers a programatic interface either directly from a radio modem or through the Virtual Cockpit.

### 3.1.3 Radio Equipment

Research indicates there are diverse direction-finding systems. A brief description of select systems is given below. We implemented a Homing DF system because of its relative advantages. We present the others because it is good to be aware of different methods available and because a reasonable particle filtering model can be developed for any of them.

- Directional Antennas: These antennas produce the strongest signal when the receiver is pointed toward the transmitter. As they turn away they quickly lose the signal entirely. Yagi and Quad antennas are examples.
- Doppler Based: These systems exploit doppler properties of waves to recover direction given that the transmitter and receiver are moving in relation to each other.
- Homing Direction Finders (Homing DFs): These systems give a left-right, front-back, or null indication. This gives an indication of which direction to turn to point toward the transmitter and "home in" on the signal.

One example of a homing DF is the L-Per from L-Tronics. This type of device uses four antenna elements: two passive and two active antennas. One passive and one active antenna are mounted on each side of a beam (see Figure 3.2). This configuration can conceptually be thought of as two antennas, one on the left and one on the right. The unit essentially measures the phase difference between the two incoming signals. If the transmitter is to the left, the left indicator light comes on, and vice-versa for a transmitter to the right. This results in a conceptual reception pattern shown in Figure 3.3. The cardiods represent which signal from the two antennas the phase comparator sees first. When both signals are equal the reading is considered a "null", which means that the transmitter is either directly in front or directly behind the receiver. This results in a null ambiguity and must be dealt with.

## 3.2 Sensor Processing

### 3.2.1 Noisy Sensor Readings

Although each measurement taken with a homing DF unit yields a heading to the transmitter, this reading is inherently noisy [7]. Noise sources include many factors. Specifically the radio measurements can themselves be off by several degrees in either

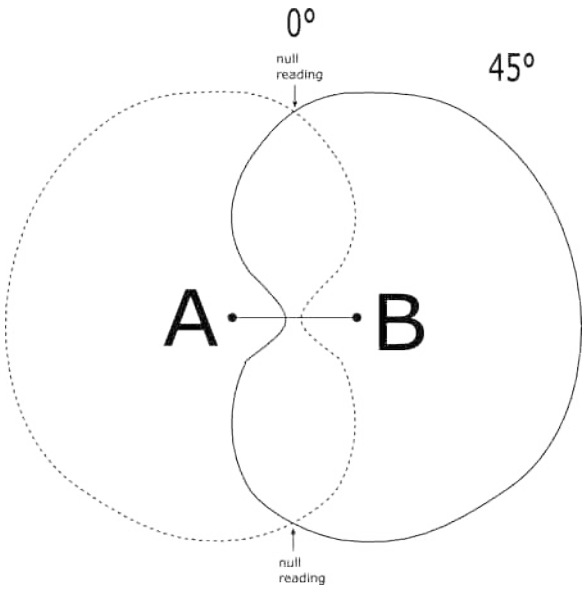


Figure 3.3: A switched cardiod antenna reception pattern [24].

direction. This error is compounded by the fact that the compass used to take heading measurements can also be off by a few degrees. This combines for an average error of about plus or minus a dozen degrees. As distances from the transmitter increase, a few degrees can lead to the estimate of the transmitter position being off significantly.

Readings are also affected by multipath and occlusions. Multipath is a common problem in many radio-based applications. Reflections can be found in many environments. In the presence of reflections the readings taken from the homing DF unit can point toward the reflection rather than toward the transmitter. This means that the transmitter can actually appear to be in two or more locations simultaneously. Much of the problem of multipath is overcome by performing RDF from the air. Additionally, multipath is more pernicious in urban environments with large buildings acting as excellent reflectors. We constrain our approaches to wilderness environments.

Occlusions occur whenever line of sight between the transmitter and reciever cannot be established. There could be a hill, cliff, or anything else in the way. Readings taken when line of sight is blocked may not correlate with the true position of the

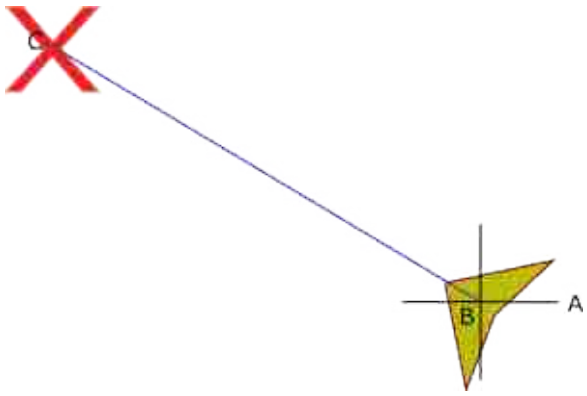


Figure 3.4: Heading of the UAV when a null is measured.

transmitter. Airborne RDF will have some multipath and occlusions, and automated RDF approaches should be equipped to handle these probabilistic readings. One way to deal with noisy readings is to model the readings as probabilistic pertubations of the ideal sensor behavior.

### 3.2.2 Nonlinear Sensor Readings

As illustrated in Figure 3.4 the heading of the transmitter from the UAV location is given by

$$\angle ABC = tan^{-1}\left(\frac{C_y - B_y}{C_x - B_x}\right) \tag{3.1}$$

where C is the transmitter location, B is the UAV location, and A is used to represent an origin. Since the readings are polar, the transformation to Cartesian is non-linear. We see this by the arctangent function. Although the distribution of the angle measured by the RDF unit is likely Gaussian with respect to the polar domain (ignoring the possible presence of terrain interference in many locations), this non-linear transformation results in a non-Gaussian distribution in the Cartesian domain.

When the UAV begins searching, it takes sensor readings. We leverage a model of how the transmitter and RDF equipment interact to use sequential RDF readings and develop an estimate of where the transmitter is. We restrict attention to the case

where the transmitter is stationary. The autopilot system provides us with filtered estimates of UAV position and heading. Our RDF equipment provides us with a heading to the transmitter. We then combine all of these elements using the particle filter.

### 3.2.3 Bayesian Reasoning

We leverage Bayesian reasoning to estimate where the transmitter is during any given time while the UAV is searching [31]. Bayes law states

$$Pr(A|B) = \frac{Pr(B|A)Pr(A)}{Pr(B)} \tag{3.2}$$

where A represents an estimate of the transmitter location and B represents an RDF reading based on UAV location and heading. For sequential estimation Bayes law can be thought of as

$$posterior = \frac{likelihood \times prior}{normalizing\ constant} \tag{3.3}$$

We use the probabilities associated with the RDF system to obtain this estimate as well as some confidence value associated with our estimate. We answer the question, "What is the probability distribution of the transmitter position given a collection of RDF readings?"

### 3.2.4 Nonlinear Bayesian Tracking

The Bayes rule above is put into service by the Bayes Filter. Recursive Bayesian Filtering attempts to estimate a state $x$ by recursively evaluating incoming sensor readings $z$. The general state and sensor models are given below. For many applications such as Bearings Only Tracking (BOT) these equations do not lend well to simplification as a linear approximation.

The state evolution model is given by

$$x_k = f_k(x_{k-1}, v_{k-1}) \tag{3.4}$$

where $x_k$ is the state at time $k$, which is a function $f_k$ of the state $x$ and process noise $v$ at time $k-1$. The sensor model represents the sensor measurement $z_k$ at time $k$ as a function $h_k$ of the state $x$ and sensor noise $n$ at time $k$:

$$z_k = h_k(x_k, n_k) \tag{3.5}$$

### 3.2.5 Particle Filter

The particle filter is a form of the Bayes Filter that represents using samples of these distributions. These samples are known as particles. The particles essentially allow us to combine several readings and their associated probabilities, as well as system and sensor noise knowledge [10]. This is possible even given non-linear functions $f_k$ and $h_k$, and non-Gaussian distributions of $v_k$ and $n_k$. We use this information to determine the probability that a particular particle represents the true location of the transmitter. For example one transmitter can actually appear to be in two or more places because of multi-path. Through resampling [3] we can change the location of particles as a function of our observations. The particles with the highest probability and the closest groupings represent a good estimate of the true location of the transmitter. Particle filters approach the optimal (w.r.t. true position distribution given our readings) estimate as we use more particles [31].

The particle filter can be conceptualized as Figure 3.5, where $x_0 \ldots x_n$ represent the state at time $t = 0, 1, \ldots, n$. The sensor readings are represented by $z_1 \ldots z_n$. The state transition $f$ shows how the state changes over time. For the RDF problem we present, we have constrained the transmitter so that it does not move (see

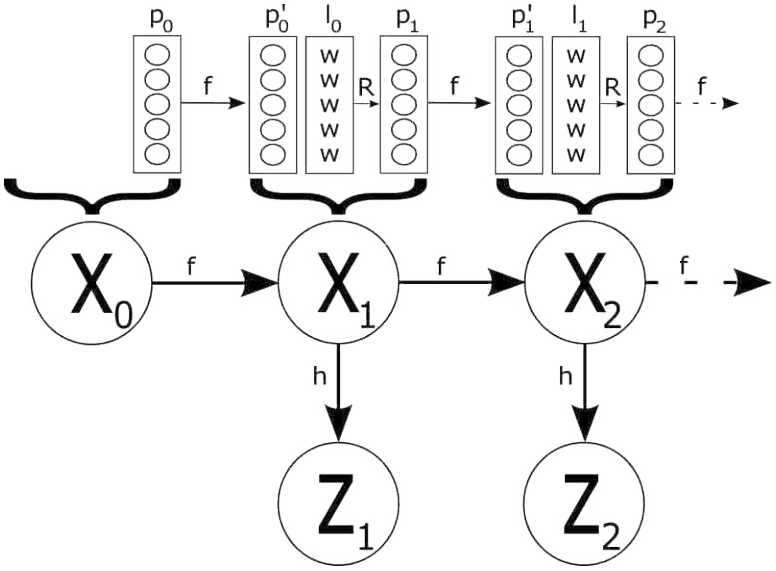


Figure 3.5: A representation of how the particle filter approximates the true state.

Section 3.2.6) therefore $f_t = f_{t-1}$. The sensor function $h$ represents how measurements relate to the true state. The estimate of the state over time is represented by $p_0 \dots p_n$ where $p_0$ is a vector and there are $n$ such vectors. Each vector is a collection of particles, which provide an estimate for characteristics of the true state at time $t$. $p'_0 \dots p'_n$ are the predictions of the state at time $t+1$. So, $p'_{t+1}$ is the predicted state at time $t+1$ given only sensor readings up to time $t$. $l_0 \dots l_n$ are sets of weights $w$ corresponding to each particle. The weights are updated using a Bayesian likelihood estimator [3]:

$$w_t^i = w_{t-1}^i * \frac{p(z_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, z_k)} \tag{3.6}$$

The importance distribution is represented by $q(x_k^i|x_{k-1}^i, z_k)$, which is often approximated as $p(x_k^i|x_{k-1}^i)$ to simplify sampling [3]. This simplifies the weight update to the following:

$$w_t^i = w_{t-1}^i * p(z_k|x_k^i) \tag{3.7}$$

We discover $p(z_k|x_k^i)$ empirically through controlled field tests with a known transmitter location.

We define the function R as the resampling function where a new set of particles is generated using the weights of the particles to compute a cumulative distribution function (CDF) and randomly drawing from it. A graphical plot of R can be seen in Figure 3.6 as we used it.

### 3.2.6 System Model

We restrict attention to the case where the transmitter does not move (remains stationary). However due to observation noise we cannot be sure of the position. To help with this resampling our system model includes some noise but remains centered at the same location. We have tried the Normal distribution with various standard deviations. Thus our system function $F$ can be characterized by $p(x_k|x_{k-1}) = N(x_{k-1}, \Sigma_k)$, where $x_k$ is the position of the transmitter at time $k$ and $x_{k-1}$, is the position at the previous time step. $\Sigma_k$ is the covariance matrix characterizing how wide the distribution is. For our case $\Sigma_k$ is a scaled identity matrix. Additionally $\Sigma_k = \Sigma_{k-1}$. We must be able to sample from this pdf during the system update phase of the filter. Results of various parameters can be found in Chapter 4.

To make the system model more robust we have implemented a Multiple Model Particle Filter (MMPF). Specifically this means that our system model probability distribution is a combination of two normal distributions. Since we have confined our system to only allow for stationary targets, the mean of both distributions is centered about $x_{k-1}$. The variance of the first distribution is very small, meaning that most particles at time $k$ are close to the particles at time $k-1$. The variance of the second distribution is larger and attempts to populate space far away with sparse particles. This gives us resistance against converging to a local maximum because there is some particle presence far away from the maximum.

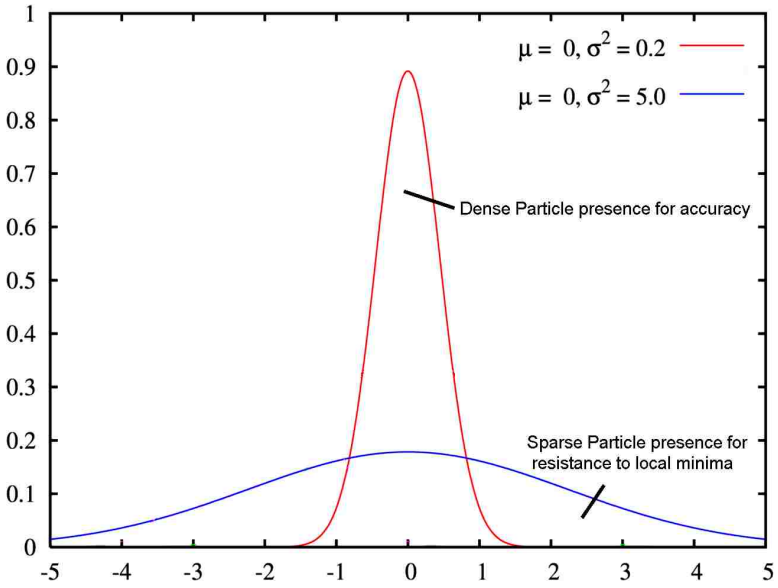


Figure 3.6: Graph of the Normal pdf with mean $\mu$ and variance $\sigma^2$. The MMPF implemented is a mixture of two Normal distributions with the same mean but one large $\sigma$ and the other small.

### 3.2.7 Sensor Model

We propose that the sensor model be based on the sensor physics derived from Figure 3.3. With some work we can know that when the sensor reads a null the aircraft is heading toward the transmitter. We compare various distributions to see which most accurately models the true sensor. The model will be based on the probability of the transmitter location by computing the angle (using arctangent) between the UAV position and the estimated transmitter location. For the sensor model we will compute the probability of the sensor reading given our transmitter estimate, $p(z_k|x_k)$, where $z_k$ is the sensor reading at time $k$. Instead of sampling from the Normal pdf (as in the system model) we compute the probability using the cumulative distribution function for the Normal.

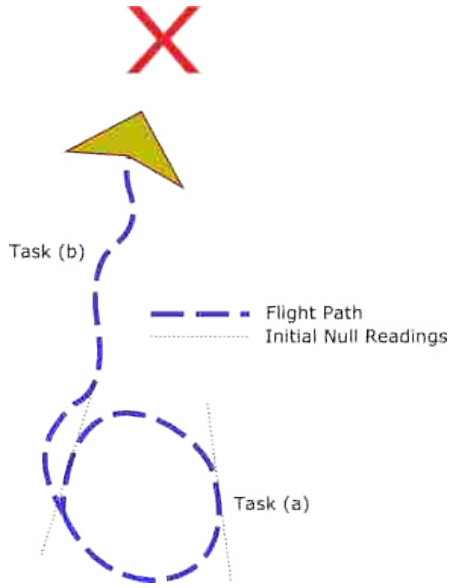


Figure 3.7: Representation of the UAV trying to take a straight line path to the transmitter to accomplish the shortest time to intercept.

## 3.3 Control

UAV control is accomplished by the Kestrel Autopilot (see Section 3.1.1) using the Virtual Cockpit GUI (Figure 3.1).

### 3.3.1 Time to Intercept

Because the shortest distance between two points is a line, the best path from the launch site to the transmitter will usually be the direct one. The control we propose to implement for this will be an adaptation of what is normally done for manned RDF (see Figure 3.7). With the dual-dipole setup this means that we first *(a)* resolve the null ambiguity and then *(b)* "surf" or "ride" the null to the transmitter.

We accomplish the first task *(a)* by commanding the UAV to fly in a circle. The BYU autopilot already supports the ability to orbit a waypoint. As we fly in a circle we will observe various left and right indications as well as two null readings, which are labeled in Figure 3.7. We can resolve which null faces the transmitter

because as we turn left from the transmitter, the RDF unit will tell us to turn right (and vice-versa). If we are facing away and we turn left it will tell us to keep turning left.

The second task is accomplished by heading in the direction of the null (towards the transmitter) which is determined during step *(a)*. As long as the reading is null we continue straight. Whenever the RDF unit tells us to turn we turn in that direction. This can be done by commanding the UAV to fly a heading which the autopilot already supports. Once the left and right fluctuations become too rapid this means that we are near the transmitter and have likely reached it. Additional confirmation may be had by video relay or other means.

Task *(b)* also requires a momentum element that enables the aircraft to keep flying in a generally consistent direction. This is just a low pass filter (LPF) of the heading. This will allow it to overcome temporarily erroneous readings caused by multi-path reflections or signal noise, etc. In the event that signal is lost for a longer period of time, the UAV will need to switch back to task *(a)* and possibly employ some other signal finding strategy.

### 3.3.2 Time to Localization

Rescue workers need to be present at a crash site in order to render aid. We therefore assert that if the UAV can minimize localization time then the rescuers can begin traveling to the site as soon as it can be found. The task of video relay serves to confirm the site and to help plan the rescue, but this can be carried out after the rescuers have begun to travel toward the site.

We hypothesize and empirically show that when RDF readings are taken for the purpose of localization, excellent resolution in terms of location can be gained by traveling perpendicular to the direction of the transmitter or in other words, on a circle centered at the radio transmitter with some radius $d$ (Figure 3.8). This may

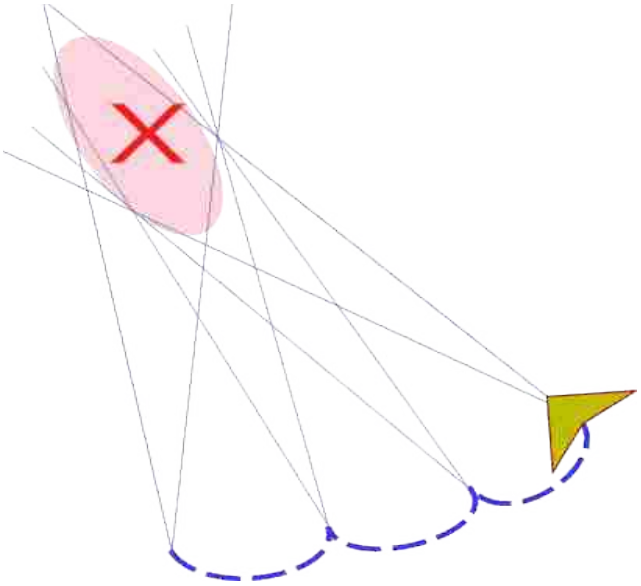

Figure 3.8: Representation of the UAV approximating a circular flight pattern to estimate the transmitters location. A time to localization approach.

seem counterintuitive to the casual observer because the UAV is not heading towards the transmitter. This means that after a first reading is taken, then the UAV should travel perpendicular to that and after some time it can take another reading. It should then travel perpendicular to the line defining the second reading, and so on. If readings could continuously be taken then we would end up with a circular path around the transmitter. We test various flight patterns to verify which flight paths are reasonable in terms of resolution for the localization task. This is dependent on the way we combine readings; see Section 3.2.2. Due to the nature of our RDF equipment, the UAV cannot take readings while it is flying perpendicular to the transmitter and therefore we will use a polygonal approximation of a circle.

## 3.4 Validation

The particle filter and homing algorithms are evaluated in computer simulation and with real world test readings. For the real world test, we use both metrics defined below and compare the homing and particle filter methods to each other. In addition

to measuring time we will also consider distance traveled by the UAV. We expect that the distance traveled by the UAV during localization using the particle filter may be several times less than the distance traveled by the homing algorithm. We expect that both methods will be a significant improvement in terms of time over a manned search.

### 3.4.1 Metrics

We believe that there are two significant metrics by which to classify and measure algorithms for RDF search. These are *(a)* Time To Intercept (TTI) and *(b)* Time To Localization (TTL). These represent two different measures in that *(a)* means the amount of time it takes between when the UAV is launched until the UAV is directly above the transmitter, and *(b)* means time from launch until the target's location has been obtained with some degree of confidence. We believe that both of these metrics are important, and our approaches will be geared toward both, with emphasis on the latter.

### 3.4.2 Other Localization Methods

In addition to the homing and particle filter algorithms we implemented some of the following alternatives for comparison. These other methods may prove to be simpler or less computationally intense.

- Grid Filter - each sensor reading is modeled as a cone. As more readings are taken the cones are added on top of each other. The probabilities are added in an accumulator type approach. The regions with the highest values are expected to be the best guess of transmitter location.
- Line Filter - linearization based on computing intersections of radio readings.

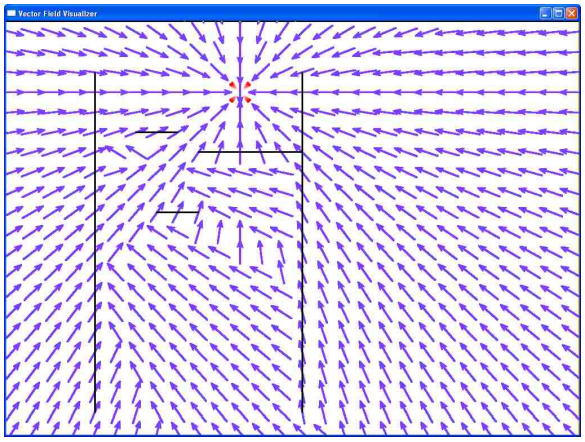


Figure 3.9: Screen shot from the RAH-DEE-OH simulator showing a simple Vector Field. This represents a non-linear radio field computed using raytracing techniques. The red X indicates the position of the simulated transmitter. The lines act as simple occlusions as well as reflectors.

- Unscented Kalman Filter (UKF) - A sigma point Kalman filter that gives a better nonlinear approximation compared to KF or EKF which only propagate the mean through the non-linearity [35].

### 3.4.3 Simulation

We have implemented the various TTI and TTL methods in simulation. The simulation environment RAH-DEE-OH is a 2D flat world that simulates the ability of the UAV to maneuever and a simplified model of the radio sensors (Figures 3.9 and 3.10). We used the simulator to develop and implement a UAV flight controller. We tested the filtering approaches mentioned above in our simulation environment. This environment allows us to compare different methods empirically based on either metric. We were also able to hone control methods before putting a platform in the air. The RAH-DEE-OH simulator is implemented using Visual C++ and DirectX.

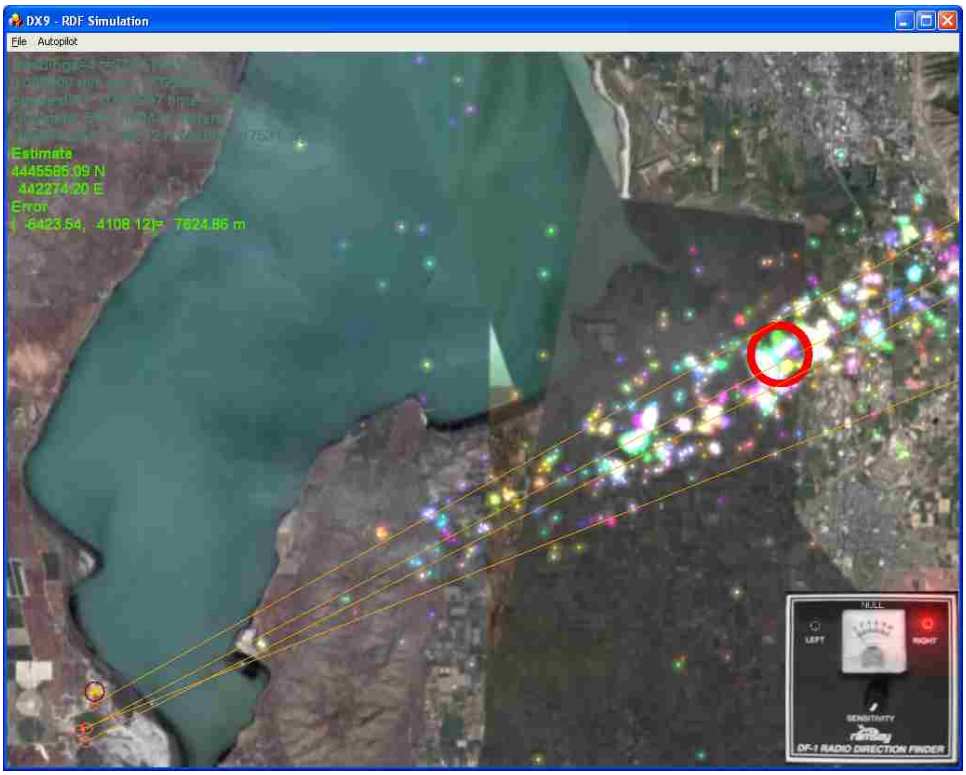


Figure 3.10: An agent autonomously gathers readings in the RAH-DEE-OH simulator.

# Chapter 4

## Results

### 4.1 Datasets

The methodology for gathering readings used in the datasets was to use the Ramses DF1C Foxhound Direction Finder. The receiver was tuned to a frequency of 135.175 MHz which is the Provo Automated Weather Observation Station (AWOS) transmitter. The transmitter has a known position of approximately 40 degrees 12 minutes 50.25 seconds North and 111 degrees 43 minutes and 35.91 seconds. Readings were gathered using the Ramsey DF1 unit built as mentioned above. When the "null" readings were found using the unit, a compass reading and a GPS position were recorded. We used a vehicle to drive between reading locations, which were often miles apart.

#### 4.1.1 Surround

The Surround Dataset compromises a series of readings taken from the ground around the Provo, UT airport in November 2006 (Figure 4.1). All readings were within 1 mile of the transmitter. Readings were taken from the roads surrounding the airport. As a result a full 360 degrees of readings was achieved. No other dataset has such a high degree of angular separation between readings. This set may represent a scenario where the techniques in this paper are employed to find a transmitter at close range. In other words in the "last mile" of the search.

Figure 4.1: Visualization of the Surround Dataset. Locations of the readings are shown by the circles surrounding the airport. The lines indicate the bearing of the readings. The icon in the center of the airport approximates the known location of the transmitter (shown only for reference).

### 4.1.2 Lake

The Lake Dataset is a series of readings taken on the west side of Lake Utah (Figure 4.2). The transmitter is on the east side of Lake Utah. The readings were taken at a general range of more than 10 miles. The readings represent about a 45-degree sweep with respect to the transmitter. This set approximates an orthogonal path planning methodology. The Lake Dataset represent a search scenario where the location of the transmitter is roughly known to within a dozen or two dozen miles. It also represents a scenario where direct navigation to the transmitter may not be possible or desired (because the distance or terrain is too difficult).

### 4.1.3 Hills

The Hills Dataset represents the most challenging set for a number of reasons. First the readings were taken from a much greater distance compared to the other sets.

Figure 4.2: Visualization of the Lake Dataset. Locations of the readings are shown on the left side of the map and the bearings are represented by the lines. The icon on the right side of the map approximates the known location of the transmitter (shown only for reference).

All readings were taken from the ground at a range of more than 20 miles. The area where the readings were taken was immediately adjacent to mountains on the west side. These mountains function as excellent reflectors and generate a source of multipath. Two distinct modes are apparent in the data as a result. Lastly all the readings were taken in a constricted area, the result being that the angular separation of the readings is minimal and represents only a few degrees (about 5 degrees).

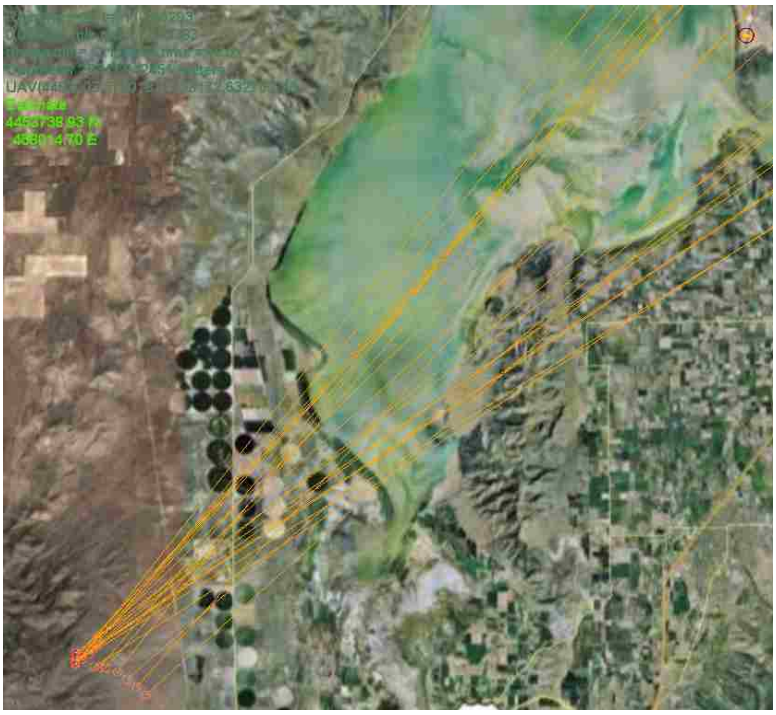

Figure 4.3: Visualization of the Hills Dataset. Locations of the readings are shown on the lower left side of the map and the bearings are represented by the lines. The icon on the top right side of the map approximates the known location of the transmitter (shown only for reference).

## 4.2 Filtering Performance

### 4.2.1 Grid Filter

The Grid Filter is attractive because it is very simple conceptually and very easy to implement. No resampling is required. The disadvantage is that lots of CPU cycles are wasted on grid cells that have very low probability. The algorithm that we used for the Grid Filter is shown in the pseudocode in Algorithm 1.

The probability of a reading given a particular transmitter position $x$ which is a 2 dimensional vector (northing and easting represented by i,j in the pseudocode) is taken from the normal distribution pdf that is given by

$$p(z_t|x) = \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{(\theta-\mu)^2}{2\sigma^2}} \tag{4.1}$$

**Algorithm 1** The Grid Filter

```
// Initialize an NxN grid of weights to normalized values (total = 1.0)
for i = 0 to N − 1 do
    for j = 0 to N − 1 do
        weights_0[i][j] = 1/(N^2)
    end for
end for
// Loop through z the vector of our sensor readings
for each z do
    for i = 0 to N − 1 do
        for j = 0 to N − 1 do
            weights_k[i][j] = weights_{k−1}[i][j] * p(z_t|x^{i,j})
        end for
    end for
    Normalize all weights
    x̂_k = weighted mean of weights_k
end for
```

| Grid Size | Grid cell area ($km^2$) |
|---|---|
| $5^2$ | 46.58608516 |
| $10^2$ | 11.64652129 |
| $25^2$ | 1.863443406 |
| $50^2$ | 0.465860852 |
| $100^2$ | 0.116465213 |
| $500^2$ | 0.004658609 |
| $1000^2$ | 0.001164652 |

Table 4.1: Grid Density.

where $\theta$ is the measured angled according to the sensor equipment and $\mu$ is the actual angle between the location of the reading and the center of the grid cell. Last $\sigma$ is the standard deviation representing the accuracy of the device used to take readings. For use with the grid filter $\sigma = 0.05$ (calculated by taking readings against our known transmitter position). When computing probabilities discontinuties between 0 and $2\pi$ must be accounted for.

Results using the Grid Filter can bee see in Figure 4.4 and show that the Grid Filter does better as distance from the transmitter decreases and when the grid size is above some minimum value. The search area we used for our tests was $34.127km$ by $34.127km$ in size. Grid density for select test values is shown in Table 4.2.1.

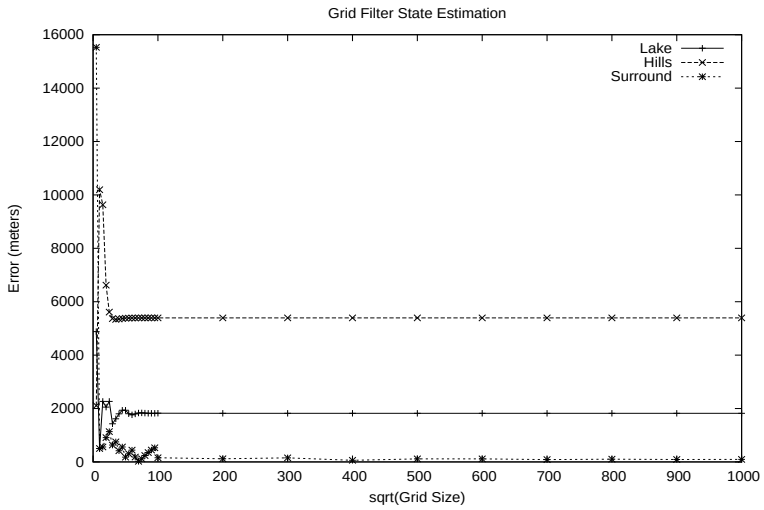


Figure 4.4: Grid filter performance on different datasets

On the datasets we see that the Grid Filter did quite well on the surround dataset, even better than both the Particle Filter and the Line Filter. We can also see that the results were good with a relatively small grid of less than 200x200. Results on the Lake dataset were perhaps useful. Performance on the Hills dataset was far less than desirable and was probably due to limited resolution for the grid cell probabilities. For out tests double precision was used but increasing precision and choosing a high grid density would likely improve results.

### 4.2.2 Line Filter

The line filter is a very simple algorithm we implemented to see how it compared with more established filtering techniques. The algorithm is not computationally intensive as long as there are not too many readings. This is due to the fact that finding the intersection of lines is a relatively simple operation (and can even be implemented in hardware for exceptionally fast runtimes). The resulting collection of intersections can then be treated with some statistical method such as finding the mean, median, or mode. This can be used to estimate the transmitter position. In our case we used the median. Results (Figure 4.5) were impressive for the surround dataset and

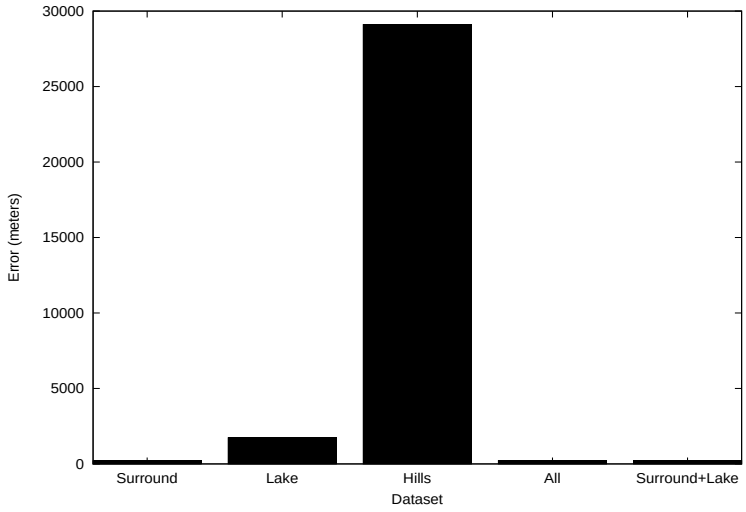


Figure 4.5: Line filter performance on different datasets

runtimes were orders of magnitude smaller than either the Particle Filter or Grid Filter method. The Lake Dataset performance was useful. The Hills dataset was dismal. Interestingly, when we combined different data sets together, the Line Filter worked very well if the Surround set was included.

Each reading contains a position (northing, easting) and a heading $\theta$ these elements combined define a line in two-space. The line is defined as intersecting the northing and easting point and having a slope related to the heading of the reading. This information is then processed using the algorithm presented in Algorithm 2.

**Algorithm 2** The Line Filter

```
for each line A in Readings do
  for each line B in Readings do
    if A!=B then
      Find intersection point of A and B // Allow only one
      Add intersection point to set C
    end if
  end for
end for
Compute median of set C
```

**Algorithm 3** The Unscented Kalman Filter (adapted from [35])

// **Initialize with:**

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

$P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]$

$\hat{x}_0^a = E[x^a] = [\hat{x}_0^T \ 0 \ 0]^T$

$$P_0^a = E[(x_0^a - \hat{x}_0^a)(x_0^a - \hat{x}_0^a)^T] = \begin{bmatrix} P_0 & 0 & 0 \\ 0 & P_v & 0 \\ 0 & 0 & P_n \end{bmatrix}$$

**for** $k = 1$ to $\infty$ **do**

// **Calculate Sigma Points:**

$\chi_{k-1}^a = \left[ \ \hat{x}_{k-1}^a \quad \hat{x}_{k-1}^a \pm \sqrt{(L+\lambda)P_{k-1}^a} \ \right]$

// **Time update:**

$\chi_{k|k-1}^x = F\left[\chi_{k-1}^x, \chi_{k-1}^v\right]$

$\hat{x}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \chi_{i,k|k-1}^x$

$P_k^- = \sum_{i=0}^{2L} W_i^{(c)} \left[\chi_{i,k|k-1}^x - \hat{x}_k^-\right]\left[\chi_{i,k|k-1}^x - \hat{x}_k^-\right]^T$

$\gamma_{k|k-1} = H\left[\chi_{k|k-1}^x, \chi_{k-1}^n\right]$

$z_k^- = \sum_{i=0}^{2L} W_i^{(m)} \gamma_{i,k|k-1}$

// **Measurement update equations:**

$P_{\tilde{z}_k \tilde{z}_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[\gamma_{i,k|k-1} - \hat{z}_k^-\right]\left[\gamma_{i,k|k-1} - \hat{z}_k^-\right]^T$

$P_{x_k z_k} = \sum_{i=0}^{2L} W_i^{c} \left[\chi_{i,k|k-1} - \hat{x}_k^-\right]\left[\gamma_{i,k|k-1} - \hat{z}_k^-\right]^T$

$K = P_{x_k z_k} P_{\tilde{z}_k \tilde{z}_k}^{-1}$

$\hat{x}_k = \hat{x}_k^- + K(z_k - \hat{z}_k^-)$

$P_k = P_k^- - K P_{\tilde{z}_k \tilde{z}_k} K^T$

**end for**

### 4.2.3 Unscented Kalman Filter

The Unscented Kalman Filter (Algorithm 3) represents a good compromise between computational efficiency and power to represent non-linear sensor models. The Extended Kalman filter only propagates the mean of the state estimate through the non-linear models. The UKF and other sigma point Kalman Filters improve upon this by propagating a set of well chosen points through the non-linear model (Unscented Transform). The EKF captures the posterior mean and covariance to the first-order (Taylor series expansion) for any non-linearity. The UKF can accurately capture the mean and covariance (Figure 4.6) to the 3rd order [35]. The UKF incorporates a scaling parameter $\lambda$ defined as

$$\lambda = \alpha^2(L + \kappa) - L \tag{4.2}$$

Where $\alpha$ determines determines the spread of sigma points about the mean we used $\alpha = 0.001$. $\beta$ incorporates prior knowledge about the distribution, for Gaussian distributions $\beta = 2$. $\kappa$ is a secondary scaling parameter usually set as $\kappa = 0$. In our tests we found that the UKF did quite well but was parameter sensitive. For instance the initial covariance of the starting distribution heavily influence results but there did not seem to be an intuition for what covariance should be selected. Figure 4.7 shows the influence of the covariance. For this test the mean of the initial distribution was a random variable with a gaussian distribution.

An example of the estimate progression for the Lake dataset can be seen in Figure 4.8. Results for the Surround dataset were somewhat useful. Results for the lake dataset were second only to the Particle Filter. Results for the Hills dataset were probably not useful.

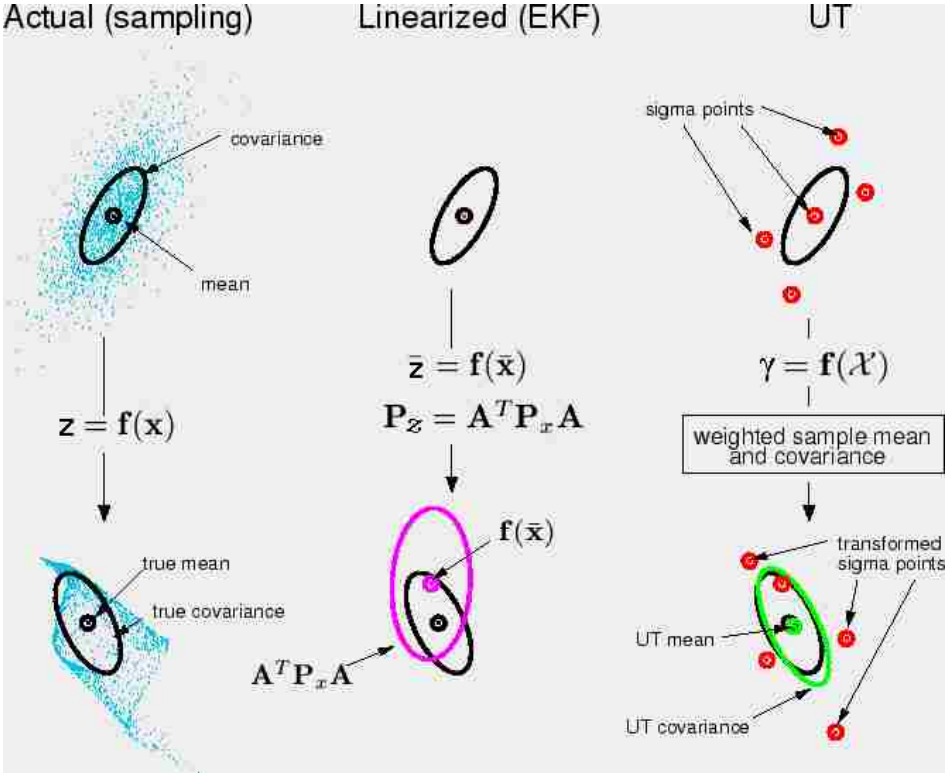


Figure 4.6: The UKF does a better job than the EKF at capturing the true mean and covariance (notation adapted from Wan [35]).

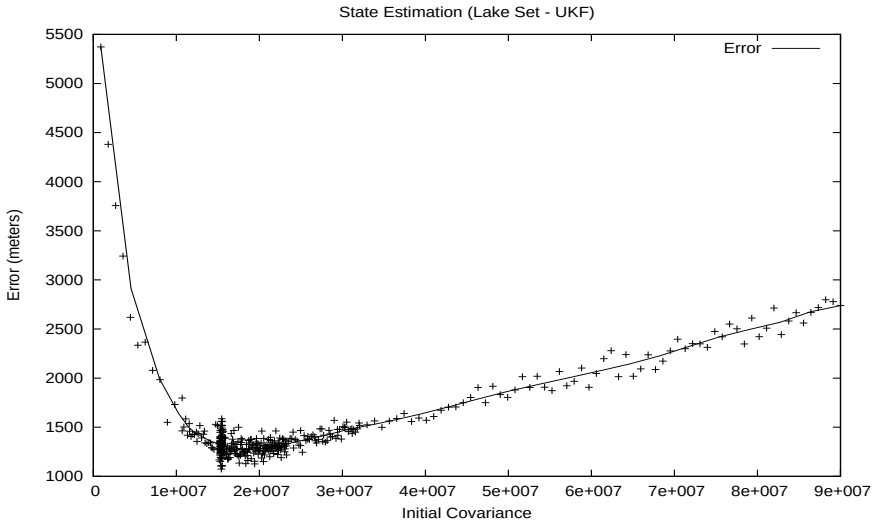


Figure 4.7: Influence of Initial Covariance using the Lake set. The mean of the initial covariance was a Gaussian distribution about the known location of the transmitter.

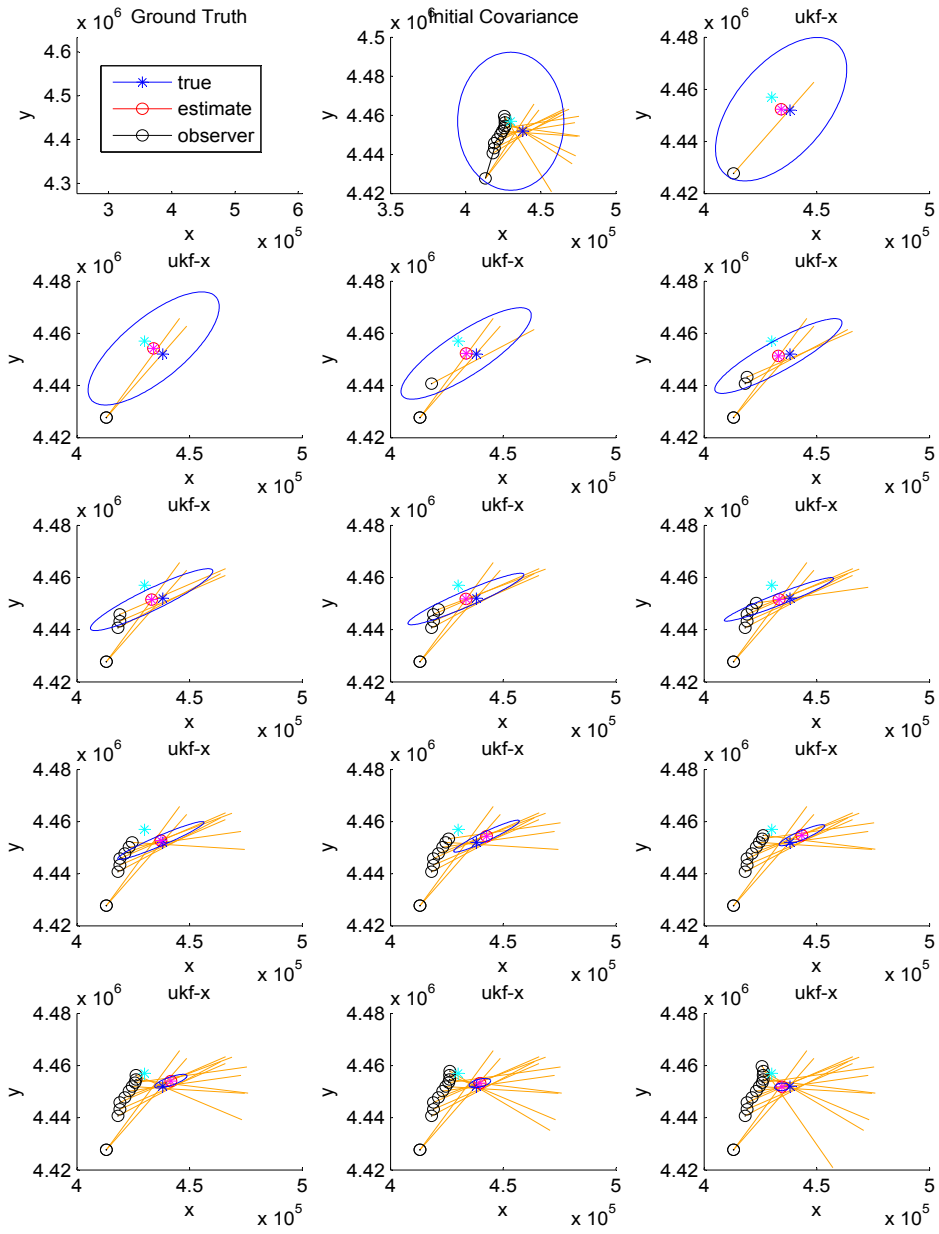


Figure 4.8: Graphical representation of the progression of the state estimate using the Unscented Kalman Filter (UKF). The covariance of the state estimate reduces as more readings are considered.

### 4.2.4 Particle Filter

The Particle Filter did well on the datasets. Accuracy of transmitter location estimates were highly dependent on the number of particles used. Increasing the number of particles improved performance but followed a law of diminishing returns. As increasing the number of particles dramatically affects run time performance, it is wise to pick the lowest number of particles that still gives desirable accuracy. Figure 4.9 shows results of all three datasets using from 100 particles to 10,000.

It is important to note that errors shown are averages over 1000 runs. This means that although the performance of 1000 or 2000 particles appears good, the variance is very high. This is illustrated by Figure 4.10 which shows reduction of standard deviation versus number of particles on the Lake Set. Since the particle filter is meant to be used in an on-line realtime environment we want the variance to be quite low. This means that for the Lake Set choosing over 4000 particles gives significant improvement over using only 1000 or 2000 particles.

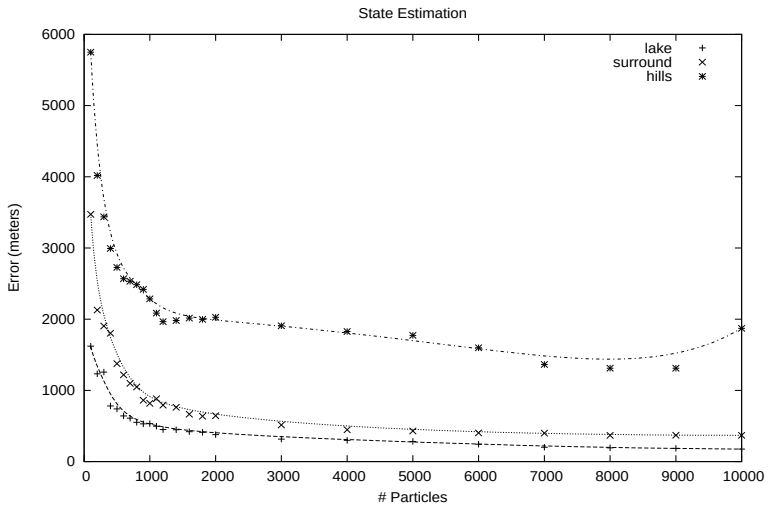


Figure 4.9: State Estimation error versus number of particles for three datasets. These curves approximate an arctangent where initially increasing particle numbers greatly improves performance but quickly yields diminishing returns as particle numbers grow larger.

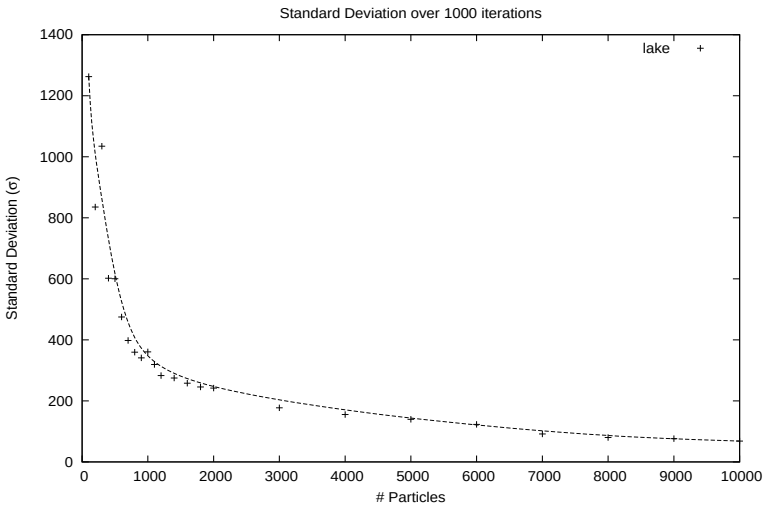


Figure 4.10: Standard Deviation in error estimates for the Lake dataset. We see that increasing the number of particles guarantees more consistent performance of the Particle Filter

In analyzing performance, another variable to consider was the modeled system noise. We found that filter performance was not very sensitive to this parameter. Although we found a "best value" for the system noise, several values yielded reasonable results. Figure 4.11 shows that sigma of 90 had the least error.

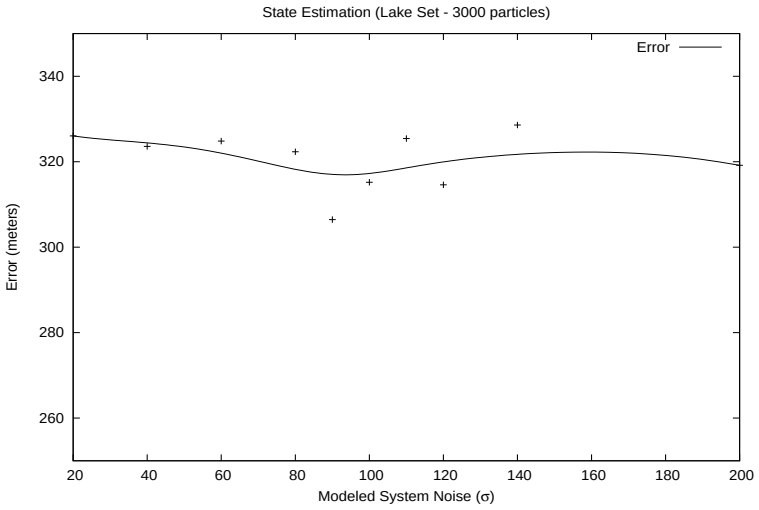


Figure 4.11: The Particle Filter as formulated is not very sensitive to the modeled system noise parameter. For the Lake dataset error was within a few meters for all values tried. Sigma of 90 yielded the lowest error.

**Summary**

For simple datasets, simple filtering algorithms will often give reasonable performance. Figure 4.12 shows that for the Surround dataset all of the filters performed well. In fact the Grid Filter and Line Filter both did better than the Particle Filter. For more difficult datasets, the Particle Filter quickly showed vast improvement over the other filters. For the Lake dataset, the Particle Filter gave very nice performance. For the Hills dataset, performance was not perfect, but it was far better than any of the other filters. The figure also shows that filter performance can be a tradeoff between computation time and accuracy. The Line Filter runs very fast and produces acceptable accuracy for some datasets. The UKF represents a good blend between accuracy and computational efficiency. The Particle Filter (and in some cases the Grid Filter) runs much slower but estimates are quite accurate.

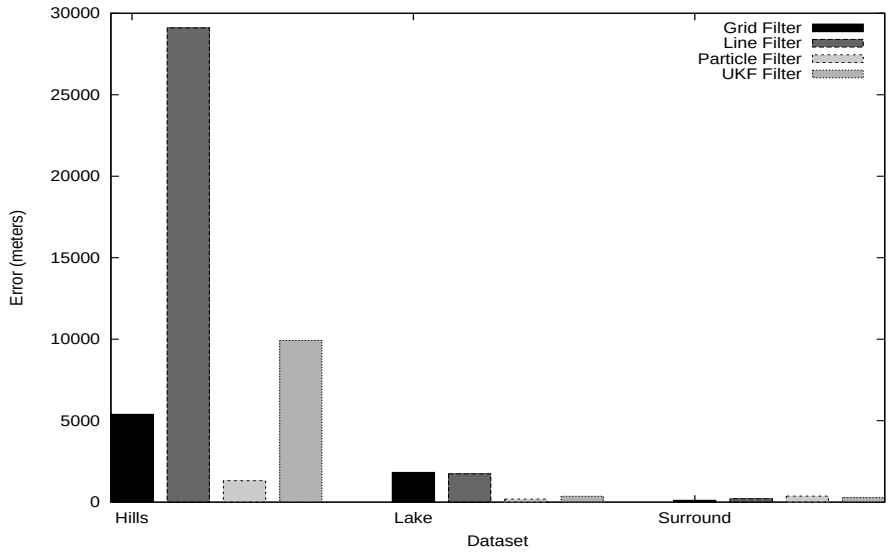


Figure 4.12: Head to Head comparison of Filtering methods. Line Filter and Grid Filter do reasonably well on simple datasets. The Particle Filter does far better on more complex datasets. The UKF also does well and is computational less demanding than the Particle Filter.

## 4.3 RDF versus Visual Search

It is likely that a radio sensor alone cannot be relied upon to find transmitters without oversight from an expert operator and sometimes other search methods. It may be useful to think of the radio sensor being used to perform a "coarse" search and then using some other method for the "fine" or local search. It is not our purpose to specify exactly how this other type of search would be carried out. One possibility is WiSAR personnel on the ground searching. Another alternative is to have the UAV flying overhead using a video camera with an operator to identify the transmitter (such as a crashed plane). For evaluation purposes we will compare against a visual search conducted using the UAV with a video camera on board.

We have identified two important metrics for evaluating Filter and Control performance, Time to Intercept (TTI) and Time to Localization (TTL). Both metrics rely on an outside source (such as a UAV operator or personnel on the ground) to confirm that the transmitter has been localized or intercepted. The transmitter is considered localized when the operator knows where the transmitter is (can see it in video or personnel on ground confirm visual). The transmitter is considered intercepted only if the UAV is in the immediate vicinity of the transmitter when localization is achieved.

We define TTI to be the time required to travel from the UAV's intial location to the transmitter location. This time is dominated by the distance between the initial location and transmitter location. The sensor and radio propagation properties are not always ideal and we must rely on the UAV operator to overcome "loops" and other anomalies while skiing the gradient toward the transmitter. Additionally there may be some time for the operator to perform the final "near field" search of the transmitter which is not neglibile.

$$t_{totalsearchtime} = (position_{transmitter} - position_{uavinitial})/velocity_{uav} \quad (4.3)$$

$$+ \; t_{lostinloops} + t_{nearfieldsearchtime} \quad (4.4)$$

If we define TTL to be the time required to gather RDF readings plus the time to do a non-radio based search to confirm location,

$$t_{totalsearchtime} = t_{rdfsearchtime} + t_{othersearchtime} \quad (4.5)$$

For comparison purposes we will present some theoretical search times using the UAV to visually search an area. We make no guarantees about whether the sensor or operator can actually detect the transmitter (crashed airplane) and realize that their are many factors such as terrain or foliage that may make it impossible to see the transmitter. Instead, we assume that if the transmitter falls within the video sensor footprint the operator will be able to identify it successfully. Our hope is to show some rough calculations on how long it would take to exhaustively cover a given area. We specify exhaustive in the sense that the camera flies over everything in the search area. There is no guarantee that anything actually would be seen.

The information we present corresponds to utilizing a mini-UAV flying wing similar to those frequently employed by researchers at BYU currently. These vehicles fly at a cruise airspeed of 14 meters/second. Their nominal operational altitude (for our purposes) is 100m AGL. The planes are typically equipped with a 60-degree fov (field of view) lens and offer resolution of about 640x480 pixels. When flying at 100m AGL the sensor footprint is roughly 115m x 115m. This configuration means that in one hour of time the UAV can search an area approximately 5.796 $km^2$.

When quantifying performance for the UKF filter on the lake dataset we found that an initial normal distribution with $\sigma = 3925m$ yielded best results. According to

rules for normally distributed data this means that 99% of the time the target can be found within $10.11km$ of $\mu_0$ our intial guess. A circular area with radius $10.11km$ has an area of $321.15km^2$. Using the UAV and camera sensor described above, it would take approximately 56 hours (2.3 days) to search the area. If we reduce the search area so that we have 95% confidence that the transmitter will be found, we get an area of $185.92km^2$ with a search time of 33 hours (1.34 days).

The RDF sensor and filtering techniques we have presented have the potential to greatly reduce the search area. For example, after approximately 40 minutes of gathering sensor data the transmitter was located to within $1.1km$ radius or an area of $3.80km^2$. It would only take 39 minutes to search this area with the camera sensor given our assumptions previously stated. This approach represents a savings of about 55 hours compared to the scenario previously framed.

# Chapter 5

## Conclusions and Future Work

Bayesian Filter methods can be successfully employed to localize radio transmitters given noisy sensor readings. These applications include but are not limited to finding downed aircraft (ELTs), fox hunts, enemy localization, avalanche victims, and lost individuals with personal beacons. We demonstrated how to employ a simple Homing Direction Finder unit to take bearing readings and locate radio transmitters. Datasets collected represent near ($< 1$ mile), midrange ($10-20$ miles), and long range ($> 20$ miles) readings. The Sequential Importance Resampling (SIR) Particle Filter seemed to perform the best under a wide range of circumstances. The Unscented Kalman Filter also performed well in many cases. For the near dataset some simpler filtering methods also worked adequately while maintaining low computational complexity.

## 5.1 Future Work

We believe that much of the work presented here lends itself to future exploration. We introduce some possible avenues for future research. We would especially like to see the possible work mentioned in Section 5.1.3 developed into a commercial product.

### 5.1.1 User Interaction

The Particle Filter may lend itself very well to user interaction. Possible ideas are to allow the user to specify the prior distribution using GUI elements. This could be placing particles individually, placing groups of particles, or whole distributions of particles interactively in real time. As the filtering process progresses the user could introduce particles into unexplored and likely areas or kill off particles in unlikely areas. In this way the user may be able to look at a terrain map and intuitively recognize areas of high or low-probability based on elevation, areas of occlusion due to terrain, and areas of multipath due to terrain. The user may be able to provide insight that would be very difficult to build into the system and sensor model directly.

### 5.1.2 Sensor Characterization

Time spent increasing the accuracy and resolution of the test sensor would improve results. Additionally, more controlled tests would allow a more complete picture of the sensor character. Using this information the sensor model embedded in the Particle Filter could be improved. For example more research in the areas of how the sensor behaves in multipath environments and how the distribution of reading headings is affected by range and broadcast power of the transmitter. This would enhance knowledge of the sensor and ultimately improve filtered transmitter estimates.

### 5.1.3 Radio Sensor Integration

Integrating the sensor on to an airframe would allow further sensor characterization as mentioned above. It would also allow automation of taking heading readings (for this work all readings were taken manually). Future work would include demonstrating better methods to gather readings.

# Bibliography


[1] T. Abbott. Functional Categories for Future Flight Deck Designs. Technical report, NASA, 1993.

[2] V. J. Aidala and S. E. Hammel. Utilization of Modified Polar Coordinates for Bearing-Only Tracking. *IEEE Trans. Automatic Control*, 28(3):283–294, March 1983.

[3] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50(2), February 2002.

[4] M. Arulampalam, B. Ristic, N. Gordon, and T. Mansell. Bearings-Only Tracking of Manoeuvring Targets Using Particle Filters. *EURASIP Journal on Applied Signal Processing*, 2004:2351–2365, 2004.

[5] Y. Bar-Shalom and X. Li. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., New York, NY, USA, 2001.

[6] T. Bréhard and J. Le Cadre. A New Approach for the Bearings-Only Problem: Estimation of the Variance-to-Range Ratio. In *7th International Conference on Information Fusion*, Stockohlm, Sweden, 2004.

[7] Civil Air Patrol. ELT/EPIRB Search, October 1991. http://members.gocivilairpatrol.com/media/cms/.

[8] Federal Telecommunications Standards Committee. Federal Standard 1037C: Glossary of Telecommunications Terms (FED-STD-1037C). National Communications System Technology Program Office, Arlington, Virginia, 1996.

[9] E. Dartanner. Locating Non-distress ELTs and EPIRBs, 1987. http://ltronics.com/lhtm-2.htm.

[10] A. Doucet, N. Gordon, and N. de Freitas. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 175 Fifth Avenue, New York, NY 10010, 2001.

[11] Arnaud Doucet, Simon Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *STATISTICS AND COMPUTING*, 10(3):197–208, 2000.

[12] E. Frew, C. Dixon, B. Argrow, and T. Brown. Radio Source Localization by a Cooperating UAV Team. In *AIAA Infotech*, Sept 2005.

[13] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEEE Proceedings F*, 140(2):107–113, 1993.

[14] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P-J. Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50(2), 2002.

[15] P. Guttorp and R. Lockhart. Finding the Location of a Signal: A Bayesian Analysis. *Journal of the American Statistical Association*, 83(402):322–330, June 1988.

[16] A. Haug. A Tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear and Non-Gaussian Processes. Technical report, MITRE Corporation, January 2005.

[17] M. Hernandez. Optimal Sensor Trajectories in Bearings-Only Tracking. *Proceedings of the Seventh International Conference on Information Fusion, FUSION 2004*, 2, 2004.

[18] J. Jonsson. Cognitive Models of Pilot Categorization and Prioritization of Flight-Deck Information. In *NASA TP 3528*, 1995. URL: http://techreports.larc.nasa.gov/ltrs/ltrs.html.

[19] R. Karlsson. Various Topics on Angle-Only Tracking Using Particle Filters. In *Reglermöte 2002 – Preprints*, pages 220–225, May 2002.

[20] R. Karlsson and F. Gustafsson. Recursive Bayesian Estimation: Bearings-Only Applications. *Radar, Sonar and Navigation, IEEE Proceedings*, 152:305–313, October 2005.

[21] X. Lin, T. Kirubarajan, Y. Bar-Shalom, and S. Maskell. Comparison of EKF, Pseudomeasurement and Particle Filters for a Bearing-only Target Tracking Problem. *Proceedings of SPIE*, 4728, 2002.

[22] D. Martin. *Communication Satellites*. The Aerospace Press, El Segundo, CA, fourth edition, 2000.

[23] P. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, Inc., 1979.

[24] J. Moell and T. Curlee. *Transmitter Hunting: Radio Direction Finding Simplified*. McGraw-Hill, Inc, Two Penn Plaza, New York, NY 10121-2298, 1987.

[25] Y. Oshman and P. Davidson. Optimization of Observer Trajectories for Bearings-Only Target Localization. *IEEE Transactions on Aerospace and Electronic Systems*, 35:892–902, 1999.

[26] A. Parsch. Directory of U.S. Military Rockets and Missiles, 2003. http://www.designation-systems.net/dusrm/app1/oq-14.html.

[27] J. Passerieux and D. Van Cappel. Optimal Observer Maneuver for Bearings-Only Tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 34:777–788, 1998.

[28] K. Pine, F. He, K. Sammut, and M. Evans. Radio direction finding for maritime search and rescue. In *Asian Control Conference (ASCC)*, volume 2, pages 723–730, Melbourne, Australia, July 2004. IEEE.

[29] M. Ridley, E. Nettleton, S. Sukkarieh, and H. Durrant-Whyte. Tracking in Decentralised Air-Ground Sensing Networks. *In Proc. of the Fifth Int. Conference on Information Fusion*, 1:616–623, July 2002.

[30] J. Sinclair. *Radio Signal Finding*. McGraw-Hill, Two Penn Plaza, New York, NY 10121-2298, 2001.

[31] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[32] O. Tremois and J. Le Cadre. Optimal Observer Trajectory in Bearings-Only Tracking for Manoeuvring Sources. *Radar, Sonar, and Navigation, IEEE Proceedings*, 146:31–39, 1999.

[33] F. Ulaby. *Fundamentals of Applied Electromagnetics*. Pearson Prentice Hall, Pearson Education, Inc. Upper Saddle River, NJ 07458, 2004 media edition, 2004.

[34] Brigham Young University. MAGICC Lab Website. http://www.et.byu.edu/groups/magicc/.

[35] E. Wan and R. Van Der Merwe. The Unscented Kalman Filter for Nonlinear Estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pages 153–158, 2000.

[36] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter. Technical Report TR 95-041, University of North Carolina at Chapel Hill, 2004.

[37] N. West. 'Venona': the British Dimension. *Intelligence and National Security*, 17:117–134, Jan 2002.

[38] L. Wills. The Influence of Technology on the Theory of Warfare: Clausewitz revises "On War" After the Battle of Britain. *National Defense Library Special Collections*, 1993.