Dynamic Waveform Design for Track-Before-Detect Algorithms in Radar

by

Ryan Piwowarski

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2011 by the
Graduate Supervisory Committee:

Antonia Papandreou-Suppappola, Chair
Chaitali Chakrabarti
Narayan Kovvali

ARIZONA STATE UNIVERSITY

December 2011

ABSTRACT

In this thesis, an adaptive waveform selection technique for dynamic target tracking under low signal-to-noise ratio (SNR) conditions is investigated. The approach is integrated with a track-before-detect (TBD) algorithm and uses delay-Doppler matched filter (MF) outputs as raw measurements without setting any threshold for extracting delay-Doppler estimates. The particle filter (PF) Bayesian sequential estimation approach is used with the TBD algorithm (PF-TBD) to estimate the dynamic target state. A waveform-agile TBD technique is proposed that integrates the PF-TBD with a waveform selection technique. The new approach predicts the waveform to transmit at the next time step by minimizing the predicted mean-squared error (MSE). As a result, the radar parameters are adaptively and optimally selected for superior performance. Based on previous work, this thesis highlights the applicability of the predicted covariance matrix to the lower SNR waveform-agile tracking problem.

The adaptive waveform selection algorithm's MSE performance was compared against fixed waveforms using Monte Carlo simulations. It was found that the adaptive approach performed at least as well as the best fixed waveform when focusing on estimating only position or only velocity. When these estimates were weighted by different amounts, then the adaptive performance exceeded all fixed waveforms. This improvement in performance demonstrates the utility of the predicted covariance in waveform design, at low SNR conditions that are poorly handled with more traditional tracking algorithms.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Research Motivation

The ability to accurately track a moving target through space is a commonly understood benefit to many real world applications. These can include commercial flight control, rocket guidance, or military airspace security amongst many others. Radar provides one means of performing tracking, where, in general, a radio-frequency signal is transmitted from a base station and the energy reflected off of the target in question is recorded and analyzed to establish estimates on the target's position and velocity. There are many approaches to implementing radar tracking and attempting to improve overall performance under various operating conditions remains an active field of research.

A common approach is to treat the target's position and velocity as random parameters and the transmission and reception of radar waveforms as a discrete time process. By making use of appropriate dynamic modeling equations for the target's state and the base station's observations, the estimation of the target's state reduces to a discrete time stochastic filtering problem. If the state process and observation process equations are both linear and the noise is modeled as Gaussian, then the well known Kalman filter [1] can be used to track the target's state. The conversion between the target position and velocity in space to and from target range and range-rate from the base station is, however, only linear in the one dimensional case. For two or three dimensional tracking, a nonlinear stochastic filtering method needs to be used. One promising approach is the use of particle filters [2, 3] to overcome the linear model restriction which also allows for the freedom of employing non-Gaussian noise models.

As applied to the tracking problem, particle filtering allows for the estimation of the posterior probability distribution of the target state which can then be used to calculate a target state estimate [4]. This approach has been extended to include the tracking of multiple targets [5]. Similar filtering has been employed in a variety of track-before-detect (TBD) scenarios. TBD has the advantage of more efficiently incorporating all available information into the target state estimate, but comes at the cost of additional complexity. With classical radar tracking, the ambiguity function (AF) observations, derived from the reflected waveform, are thresholded to provide a range and range-rate estimate of the target before being used to further estimate the target state. For non-radar tracking applications, the observations are derived from some other source like an IR camera pointed towards the general position of the target. In TBD, there is no thresholding of the observations (i.e., AF measurements or image pixels); all observations are used. TBD tracking has been examined in general [6, 7], and explored for various specific conditions: unknown target amplitude [8], amplitude flucuations [9], extended targets [10] and complex measurements [11] are only a few examples.

A more recent development in radar based tracking is the on-line adaptation of the transmitted signal with the intent of improving tracking performance. Modern radar systems have the flexibility to change frequency and envelope characteristics of the transmitted signal to adapt to various dynamic operating conditions, and algorithms to choose those waveform characteristics are being actively researched. A common approach is to make use of a waveform library [12–14], establishing an algorithm to choose from that library at each time step to optimize some criteria. Radar based tracking algorithms have been developed to minimize the mean-squared error (MSE) under high SNR conditions [15, 16]. Low SNR radar environments pose additional challenges to both tracking and dynamic waveform selection and are the focus of this thesis.

1.2 Low SNR Target Tracking

Under high SNR conditions, the AF observations over the delay-Doppler plane can be thresholded with a reasonable expectation that the largest value will correspond to the target's true delay and Doppler. Additionally, the Cramér-Rao lower bound of the target state estimate can be directly approximated by evaluating the curvature of the AF at the origin for each waveform in the library, allowing for the selection of the optimal waveform at each time step. However, in low SNR environments, the peak AF observation can no longer be expected to be anywhere near the target's true delay and Doppler and a new method for tracking and waveform selection is necessary.

The approach used in this thesis is to employ TBD methods to overcome the low SNR constraint. By avoiding the thresholding of the AF observations, all available information is included in the state estimate instead of that gained from solely the largest observation, leading to improved performance. Additionally, we will make use of a recursive algorithm to calculate the predicted MSE [17, 18] of the tracker for each potential waveform in a fixed library of linear frequency-modulated waveform candidates. Using this lower bound, we will determine the optimal waveform to use at each time step according to a predefined error weight. With this approach, a tracking radar is able to adaptively and autonomously determine the best waveform to use based on the current covariance of the state estimate as well as the state and process models, minimizing the total MSE.

This application of the predicted covariance to the low SNR radar tracking problem was first proposed in [19, 20]. The contributions of this thesis include: an independent implementation of the algorithm, focusing primarily on the waveform selection performance without the additional complications of target detection or

estimation of dependent measurement correlations. This implementation has the additional benefit of being significantly faster in operation, which allowed for the exploration of a wide variety of parameter options not previously examined. Finally, the idea of "entire plane" ambiguity function observations with adaptive grid spacing is introduced and explored along side the traditional fixed grid approach.

1.3 Thesis Organization

This thesis is ordered as follows. Chapter 2 covers the basic principles of estimation of any general hidden system, the application of particle filters to the problem and the specific application of the system estimation to the two-dimensional (2-D) tracking problem. Chapter 3 introduces classical tracking, explains its dependence on high SNR operating conditions and presents the track-before-detect approach as a means of overcoming that restriction. Chapter 4 describes the class of waveforms to be used in the adaptive waveform library and defines the set of ambiguity function delay-Doppler observation points to be used in the observation model for two general cases: full delay-Doppler grid and fixed grid. Chapter 5 presents a means of predicting the tracking estimation error at each time step and provides an algorithm that takes advantage of this predictive ability to optimally choose waveform parameters. Chapter 6 presents simulation results while chapter 7 provides some conclusive remarks and discusses potential future work.

Chapter 2

ESTIMATION OF A HIDDEN SYSTEM

In this thesis, we will be estimating the state of a hidden system. This section provides a general introduction to system estimation, it discusses particle filtering [2], and it applies this general description to the specific case of 2-D tracking with radar.

2.1 System Estimation

It is often the case where it is desired to estimate parameters of interest of a hidden dynamic system, where "hidden" means that the only available observations of the system are both a function of the system state and corrupted by noise. Examples are wide ranging and include:

1. Temperature estimation using noisy thermometers.
2. Animal population estimation using limited population sampling data.
3. Estimating dynamic channel response for wireless communications.
4. Estimation in a global positioning system.
5. Tracking moving targets with inexact radar measurements.

In many practical cases, the dynamic state of the hidden system can be modeled as a stochastic process, known as the "state process". At any particular time, k, the unknown state process can be fully described by an N_x-dimensional state vector $\mathbf{x}_k$. In general, this state process can be continuous or discrete, but only the discrete case will be examined here, where the time parameter k is limited to non-negative integers. If the system state is from $\mathbb{R}^{N_x}$, the set of real numbers of

dimensionality N_x, then the state at time $k \geq 0$ is a random column vector with a state instance

$$\mathbf{x}_k \triangleq [x_{k,0},\ x_{k,1},\ \ldots,\ x_{k,N_x-1}]^T \tag{2.1}$$

and $p(\mathbf{x}_k)$ is the probability density function of $\mathbf{x}_k$.

With the actual state hidden, we must make use of noisy observations to estimate the state. These observations can also be modeled as a stochastic process, known as the "observation process". At time $k \geq 0$, the observation process measurements can be fully described by a length N_z column vector $\mathbf{z}_k$. If the observations are similarly from $\mathbb{R}^{N_z}$, then the observation at time $k \geq 0$ is a random vector with observation instance:

$$\mathbf{z}_k \triangleq [z_{k,0},\ z_{k,1},\ \ldots,\ z_{k,N_z-1}]^T \tag{2.2}$$

and $p(\mathbf{z}_k)$ is the probability density function of $\mathbf{z}_k$. Additionally, $\mathbf{z}_0 \triangleq \mathbf{0}$, by definition.

With the state and observation processes defined, we look at the problem of estimating the history of state values from the history of observation values

$$\hat{\mathbf{x}}_{0:k|k} = g(\mathbf{z}_{1:k}) \tag{2.3}$$

where $g(z_{1:k})$ is a function of a ll the observations, from time step 1 to k.

Of the many available cost functions to choose from, if the desire is to minimize the mean-squared error of the estimate, then

$$g(\mathbf{z}_{1:k}) = E[\mathbf{x}_{0:k}|\mathbf{z}_{1:k}] \tag{2.4}$$

which requires knowledge of $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$.

In order to efficiently estimate the state history as each observation is made, it is desirable to use a recursive formulation for determining the posterior density at

time k from the one at time $k-1$ [1, 25]

$$\begin{aligned} p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) &= \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})}{\int p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) d\mathbf{x}_{0:k}} \\ &= \frac{p(\mathbf{x}_k, \mathbf{z}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k-1}) p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})}{\int p(\mathbf{x}_k, \mathbf{z}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k-1}) p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) d\mathbf{x}_{0:k}} \end{aligned} \tag{2.5}$$

At this point, two additional simplifying assumptions are common: Markov dynamics and memory-less observations. The Markov dynamics assumption is expressed as

$$p(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k-1}) \triangleq p(\mathbf{x}_k|\mathbf{x}_{k-1}) \tag{2.6}$$

Under the assumption of Markov dynamics, the state is related to only the *immediately previous* state through a possibly nonlinear and time varying function $f_k()$, such that

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}, \mathbf{w}_{k-1}) \tag{2.7}$$

where $\mathbf{w}_k \in \mathbb{R}^{N_w}$ is an instance of an independent and identically distributed state process noise sequence of dimension N_w.

The assumption of memory-less observations is expressed as

$$p(\mathbf{z}_k|\mathbf{x}_{0:k}, \mathbf{z}_{1:k-1}) = p(\mathbf{z}_k|\mathbf{x}_k) \tag{2.8}$$

Under the assumption of memory-less observations, the observations are related to only the *current* state through a possibly nonlinear and time varying function h_k, such that:

$$\mathbf{z}_k = h_k(\mathbf{x}_{k,}\mathbf{v}_k) \tag{2.9}$$

where $\mathbf{v}_k \in \mathbb{R}^{N_v}$ is an instance of an independent and identically distributed observation noise sequence of dimension N_v.

With these assumptions in place, the recursive formula for the posterior density becomes

$$p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) d\mathbf{x}_{0:k}} \tag{2.10}$$

Finally, in cases where there is interest in only $\mathbf{x}_k$, the marginal distribution $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ also satisfies a modified recursion relation comprised of two steps. The first step is prediction

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})d\mathbf{x}_{k-1} \tag{2.11}$$

and the second step is update

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})d\mathbf{x}_k} \tag{2.12}$$

With the posterior density determinable, the state estimate can then be calculated.

2.2 Particle Filter Sequential Monte Carlo Approach

In this thesis, we attempt to recursively solve (2.11) and (2.12) with a particle filter approach. If (2.7) and (2.8) were both linear with Gaussian distributed noise, then a more direct Kalman filter could be used [1]. As we will see later, however, the observations are not linear.

A particle filter attempts to directly represent the continuous probability density, $p(\mathbf{x}_k|\mathbf{z}_{0:k})$, and use that representation to recursively update the density for the next time step. We construct a generic probability density by drawing N "particles" from a continuous distribution; $\mathbf{x}^n$ for $n \in \{0, 1, \ldots, N\}$ and then construct a proxy density function as a distribution of impulses at the sampled locations. Note that in this context, $\mathbf{x}^n$ means the nth particle, not $\mathbf{x}$ to the power of n

$$p(\mathbf{x}) \approx \frac{1}{N}\sum_{n=1}^{N}\delta(\mathbf{x} - \mathbf{x}^n) \tag{2.13}$$

As the number of samples approaches infinity, the proxy density will converge to the actual density. For any finite number of samples, however, the proxy

density is a discrete function and therefore does not have a non-zero value for all $\mathbf{x}$ where $p(\mathbf{x})$ is non-zero. The proxy density becomes a useful approximation when it is substituted into an equation that integrates over the actual density. An example equation would be taking the expectation of $\mathbf{x}$.

$$\bar{\mathbf{x}} = E[\mathbf{x}] = \int_{\mathbf{x}} p(\mathbf{x})\mathbf{x}d\mathbf{x} \approx \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}^n \tag{2.14}$$

Even though the proxy density "skips" some values of $\mathbf{x}$, the estimate remains both useful and, most importantly, practical to calculate.

The density of interest for the hidden model estimation problem, described in Equations (2.11) and (2.12), is the posterior density of $\mathbf{x}_k$ given the set of observations up to time step k, or, $p(\mathbf{x}_k|\mathbf{z}_{1:k})$. If this density is readily available to sample from, the estimation problem is only a matter of using a proxy density to evaluate an expectation of $\mathbf{x}_k|\mathbf{z}_{1:k}$. However, in general, it is not a known density or cannot be easily sampled from. We turn instead to importance sampling, where we make use of a new importance density, q, that *can* be sampled from. We thus modify the proxy density to be

$$p(\mathbf{x}) \approx \sum_{n=1}^{N} w^n \delta(\mathbf{x} - \mathbf{x}^n) \tag{2.15}$$

$$w^n \propto \frac{p(\mathbf{x}^n)}{q(\mathbf{x}^n)} \tag{2.16}$$

where now the particles, $\mathbf{x}^n$, are drawn from q, not p, and the weights are normalized before use.

Introducing the specific densities of (2.12), expanding the numerator, and integrating, we have

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{z}_{1:k}) &= \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \\ &\propto p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \end{aligned} \tag{2.17}$$

and, choosing an importance density that can be recursively computed

$$q(\mathbf{x}_k|\mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_{1:k})q(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}), \tag{2.18}$$

we can substitute (2.17) and (2.18) into (2.15) and (2.16) to get

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{n=1}^{N} w_k^n \delta(\mathbf{x}_k - \mathbf{x}_k^n) \tag{2.19}$$

$$w_k^n \propto \frac{p(\mathbf{z}_k|\mathbf{x}_k^n)p(\mathbf{x}_k^n|\mathbf{x}_{k-1}^n)p(\mathbf{x}_{k-1}^n|\mathbf{z}_{1:k-1})}{q(\mathbf{x}_k^n|\mathbf{x}_{k-1}^n, \mathbf{z}_{1:k})q(\mathbf{x}_{k-1}^n|\mathbf{z}_{1:k-1})} \tag{2.20}$$

$$w_k^n \propto w_{k-1}^n \frac{p(\mathbf{z}_k|\mathbf{x}_k^n)p(\mathbf{x}_k^n|\mathbf{x}_{k-1}^n)}{q(\mathbf{x}_k^n|\mathbf{x}_{k-1}^n, \mathbf{z}_{1:k})} \tag{2.21}$$

Now the weights and the proxy density can be recursively evaluated. A common choice for the importance density, that is also simple to implement, is to choose $q(\mathbf{x}_k^n|\mathbf{x}_{k-1}^n)$, which reduces (2.21) to

$$w_k^n \propto w_{k-1}^n p(\mathbf{z}_k|\mathbf{x}_k^n) \tag{2.22}$$

Algorithm 1 provides a pseudo-code for implementing a basic particle filter using these results. The re-sampling is to avoid particle degeneracy as the simulation proceeds [2] and each loop of the algorithm ends with advancing the particles according to (2.7).

Algorithm 1 Basic Particle Filter

Assume an initial estimate of value μ_s
for $n = 1$ to N **do**
  Draw $\mathbf{x}^n$ from $p(\mathbf{x}_0)$
  $w^n = 1/N$
end for
for $k = 1$ to K **do**
  Measure $\mathbf{z}_k$
  for $n = 1$ to N **do**
    Update: $w^n = w^n p(\mathbf{z}_k|\mathbf{x}^n)$
  end for
  Sum the weights: $w_{total} = \sum_{n=1}^{N} w^n$
  for $n = 1$ to N **do**
    Normalize: $w^n = w^n/w_{total}$
  end for
  Estimate: $\hat{\mathbf{x}}_k = \sum_{n=1}^{N} w^n \mathbf{x}^n$
  Resample particles
  for $n = 1$ to N **do**
    Draw process noise $\mathbf{y}$ from $p(\mathbf{w})$
    Advance: $\mathbf{x}^n = f(\mathbf{x}^n, \mathbf{y})$
  end for
end for

2.3 The Two Dimensional Radar Tracking Problem

Figure 2.1 displays the ideal radar system model being used in this thesis. A target at an unknown location in the x-y plane is moving at an unknown velocity. Meanwhile, a transmitter at the origin periodically transmits a modulated waveform and records any reflections from the target. Using reflection recordings and an *a priori* probability distribution of the target state, a posterior distribution can be calculated. The posterior distribution is then forwarded in time to the next waveform transmission time, becoming the new prior, and the process is repeated. In this way, the target state can be tracked.

To start, we define the entire state of the target at time step k as

$$\mathbf{x}_k \triangleq [x_k, y_k, \dot{x}_k, \dot{y}_k]^T \tag{2.23}$$

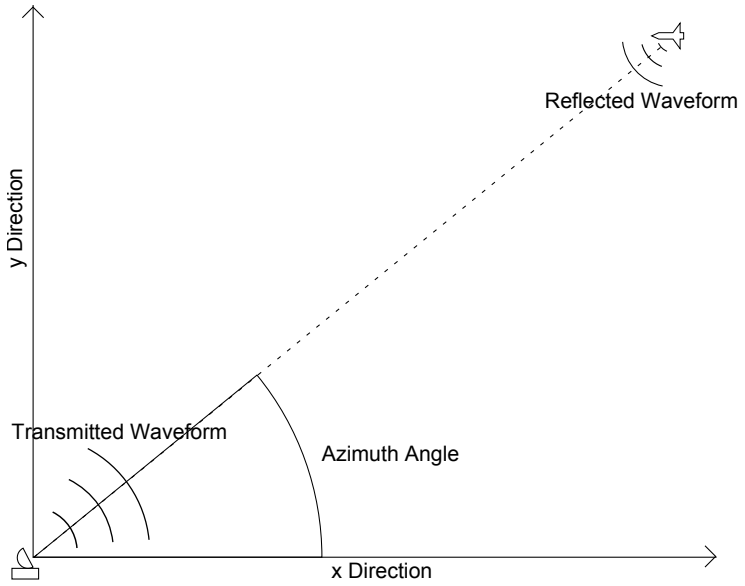


Figure 2.1: A target is moving in the x-y plane while a radar base station located at the origin transmits periodic waveforms and records any reflections.

where the T denotes transpose and the components of the column vector are, respectively, the x-coordinate, y-coordinate, velocity in the x direction, and velocity in the y direction.

However, the target state is dynamic in time. Not only does its position change with time, but so does its velocity. How the target's state changes over time is modeled by the state process. In this case, the state process is simply a linear function of the previous state corrupted with additive white noise

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k \tag{2.24}$$

$$\mathbf{x}_0 \sim \text{Normal}(\mu_s, \mathbf{C}_s) \tag{2.25}$$

where

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.26}$$

and Δt is the fixed time interval, in seconds, between state transitions. Note that the model simply adds the current x and y velocities at each step back into the positions. Equation (2.25) signifies that the initial state is drawn from a normal distribution with mean μ_s and covariance matrix $\mathbf{C}_s$.

Unexpected variations in velocity and position, due to headwinds for example, are incorporated and modeled by the process noise $\mathbf{w}_k$, of the same size and description as $\mathbf{x}_k$.

Deriving the state process noise covariance matrix is the final step in establishing the complete state process model. Using the constant velocity model assumption, the velocity of the target is assumed to be constant except for the perturbations introduced by a random variable, while the position is derived from integrating the velocity over time. If given the variance of the Gaussian distributed velocity perturbation, the question is then: what is the variance of the position perturbation, and what is the covariance of the position and velocity perturbation random variables. In this model, the x and y velocities are considered independent, so the derivation will follow for one dimension and is easily simply duplicated to cover the two dimensional case.

If we subdivide the Gaussian velocity perturbation P, with variance σ_v^2, into N Gaussian random variables P_i, with $i = 1, \ldots, N$, and each with variance σ_v/N, then the total position perturbation can be calculated as

$$S = \frac{\Delta t}{N}(P_1 + 2P_2 + \cdots + NP_N) \tag{2.27}$$

and the variance of S is then, as $N \to \infty$,

$$\text{Var}(S) = \left(\frac{\Delta t}{N}\right)^2 \sum_{i=1}^{N} i^2 \frac{\sigma_v^2}{N} = \sigma_v^2 \sum_{i=1}^{N} \left(\frac{\Delta t}{N} i\right)^2 \frac{1}{N} \to \sigma_v^2 \int_0^{\Delta t} x^2 dx = \sigma_v^2 \frac{(\Delta t)^3}{3} \tag{2.28}$$

Using a similar process for the covariance

$$\text{Covar}(S, P) = \sigma_v^2 \frac{(\Delta t)^2}{2} \tag{2.29}$$

And extending these results to the two dimensional case, the full covariance matrix of the process noise is now

$$\mathbf{Q} = \sigma_v^2 \begin{bmatrix} \frac{(\Delta t)^3}{3} & 0 & \frac{(\Delta t)^2}{2} & 0 \\ 0 & \frac{(\Delta t)^3}{3} & 0 & \frac{(\Delta t)^2}{2} \\ \frac{(\Delta t)^2}{2} & 0 & \Delta t & 0 \\ 0 & \frac{(\Delta t)^2}{2} & 0 & \Delta t \end{bmatrix} \tag{2.30}$$

such that

$$\mathbf{w}_k \sim \text{Normal}(\mathbf{0}, \mathbf{Q}) \tag{2.31}$$

Given parameters σ_v^2, Δt, along with initial condition parameters μ_s and $\mathbf{C}_s$, a particular state process is defined. The observation process depends on the manner in which target estimates are determined. A traditional approach as well as the track-before-detect approach will be described in the next section.

Chapter 3

TRACK BEFORE DETECT

In this section, we will cover two methods of establishing an observation process from the radar measurements. First, we describe the traditional approach to estimating the target state under the assumption of high SNR conditions. Then an alternative estimation approach, track-before-detect (TBD), is introduced in an attempt to loosen this high SNR restriction.

3.1 Traditional Target State Estimation

The transmitted waveform is first defined as

$$s_e(t) \triangleq \sqrt{2}\text{Re}[\sqrt{E}s(t)e^{j\omega_c t}] \tag{3.1}$$

where Re signifies taking the real part of the complex function, $\sqrt{E}$ is the amplitude of the signal, $s(t)$ is the complex envelope of the waveform and ω_c is the carrier frequency in radians.

The total round-trip travel time of the waveform, τ_a, from radar station to target back to radar station is defined as

$$\tau_a \triangleq \frac{2}{c}R = \frac{2}{c}\sqrt{x^2 + y^2} \tag{3.2}$$

where c is the speed of light, R is the radial distance from the station to the target, and (x, y) is the 2-D coordinate of the target position.

The radial velocity, ν, is simply the rate of change of R with time

$$\nu \triangleq \frac{dR}{dt} = \frac{x\dot{x} + y\dot{y}}{\sqrt{x^2 + y^2}} \tag{3.3}$$

where $(\dot{x}, \dot{y})$ is the 2-D coordinate for the target velocity.

If we assume that the reflection process is frequency-independent and linear, and that $\nu/c << 1$ and $WT << c/(2\nu)$, where WT is the time-bandwidth product of the waveform, then from [21], the received signal can be approximated as

$$r_e(t) \approx \sqrt{2}\text{Re}[\sqrt{E}\tilde{b}s(t-\tau_a)e^{j\rho_a t}e^{j\omega_c t}] \tag{3.4}$$

where $\tilde{b}$ is a function of the reflective properties of the target, ρ_a is the Doppler shift after reflection and

$$\rho_a \triangleq -\frac{2\omega_c}{c}\nu \tag{3.5}$$

For the rest of this thesis, $\sqrt{E}$ and $\tilde{b}$ are both assumed equal to 1, so that

$$s_e(t) \triangleq \sqrt{2}\text{Re}[s(t)e^{j\omega_c t}] \tag{3.6}$$

$$r_e(t) \approx \sqrt{2}\text{Re}[s(t-\tau_a)e^{j\rho_a t}e^{j\omega_c t}] \tag{3.7}$$

At the receiver, the received waveform is demodulated to produce the received waveform complex envelope

$$r(t) = s(t-\tau_a)e^{j\rho_a t} \tag{3.8}$$

Traditionally, under assumed high SNR, the next step at this point is to first estimate τ_a and ρ_a through the use of a maximum likelihood estimator. These estimates represent the nonlinear noisy observations of the target state

$$\begin{aligned} \tilde{\tau}_a &= \frac{2}{c}\sqrt{x^2+y^2} + v_\tau \\ \tilde{\rho}_a &= -\frac{2\omega_c}{c}\frac{x\dot{x}+y\dot{y}}{\sqrt{x^2+y^2}} + v_\rho \end{aligned} \tag{3.9}$$

where v_τ and v_ρ are additive white noise used to model the error of the estimates.

To obtain the delay and Doppler estimates, we first define the ambiguity function (AF) as the Fourier transform of the time correlation of the transmitted and

received complex envelopes

$$\begin{aligned} \mathsf{AF}(\tau,\rho) &= \int_{-\infty}^{\infty} r(t)s^*(t-\tau)e^{-j\rho t}dt \\ &= \int_{-\infty}^{\infty} s(t-\tau_a)s^*(t-\tau)e^{j(\rho_a-\rho)t}dt \end{aligned} \tag{3.10}$$

The likelihood function is then the magnitude squared of the ambiguity function.

$$|\mathsf{AF}(\tau,\rho)|^2 \tag{3.11}$$

by defining

$$\begin{aligned} \tau_d &= \tau - \tau_a \\ \rho_d &= \rho - \rho_a \end{aligned} \tag{3.12}$$

and making the substitution

$$z = t - \tau + \frac{\tau_d}{2} \tag{3.13}$$

it becomes clear that (3.10) is a function of the delay and Doppler "distance" from the true values, τ_a and ρ_a.

$$\mathsf{AF}_s(\tau_d,\rho_d) = \int_{-\infty}^{\infty} s(z-\frac{\tau_d}{2})s^*(z+\frac{\tau_d}{2})e^{-j\rho_d t}dz \tag{3.14}$$

Note that (3.11) is maximum at $\tau_d = 0$, $\rho_d = 0$, or equivalently, when $\tau = \tau_a$, $\rho = \rho_a$. This means the ambiguity function, in the absence of noise, is centered on the true delay and Doppler of the target.

In practice, the values of the ambiguity function are determined by passing the received signal through a matched filter bank fixed for τ's of interest with the Fourier transform of the results evaluated for the ρ's of interest. It is from the output of the ambiguity function, evaluated at particular discrete points, that the delay and Doppler estimates are determined. If the maximum output likelihood value is above some threshold, a target is classified as present. The estimates are then fed into a tracking algorithm to further estimate the target state over time. This approach can then be understood to be a "detect-before-track" solution to estimating the target state.

3.2 Track-Before-Detect Observation Model

As the previous section mentioned, the peak of the ambiguity function occurs at the true delay and Doppler of the target. However, the AF has significant side lobes that can not be ignored in general. Any noise present in the received waveform will manifest itself as local peaks and troughs of the AF throughout the delay-dopper plane. As long as that noise is small enough, the global peak of the noisy AF will serve as a reasonable estimate of the true peak of the noiseless AF. However, in cases where the noise is too high, local peeks throughout the delay-Doppler plane can easily mask and overpower the true noiseless peak (see Figure 4.3). What is needed in this scenario is another means of taking measurements of the target state that accounts for this.

In this thesis, we attempt to overcome this low SNR restriction by using all of the evaluated discrete noisy AF values, instead of simply choosing the largest one. By evaluating our estimates over the entire delay-Doppler plane, we move directly from received waveform to estimated target state, without ever first estimating τ_a or ρ_a. This approach, where both probability of detection of a target and target state estimates can be evaluated concurrently is called "track-before-detect".

We can now proceed to formally defining the observation model used. There are two primary measurements in this model: the recorded reflection of the transmitted waveform, and the azimuth angle. Both are functions of the target state and corrupted by noise in some way. For clarity, the subscript k has been dropped in this section, but it should be understood that the measurements at each time step k are derived from the target state at time step k.

The azimuth measurement is straightforward

$$z_\theta = h_\theta(\mathbf{x}) + v_\theta = \tan^{-1}(y/x) + v_\theta \tag{3.15}$$

where

$$v_\theta \sim \text{Normal}(0, \sigma_\theta^2) \tag{3.16}$$

The noisy ambiguity function, evaluated at particular discrete points, makes up the remainder of the observations in our observation model.

$$z_{AF}(m) = h_{AF}(\mathbf{x}) + v_{AF}(m) = |\text{AF}(\tau_m, \rho_m)|^2 + v_{AF}(m), \text{for } m = 1 \text{ to } M-1 \tag{3.17}$$

where M is the total number of scalar observations making up the observation vector and

$$\mathbf{v}_{AF} = [v_{AF}(1), v_{AF}(2), \ldots, v_{AF}(M-1)]^T \sim \text{Normal}(\mathbf{0}, \mathbf{R}_{AF}) \tag{3.18}$$

In this thesis, all of the ambiguity function measurement noise is modeled as independent so that

$$\mathbf{R}_{AF} = \sigma_{AF}^2 \mathbf{I} \tag{3.19}$$

where σ_{AF}^2 is the ambiguity function noise power and $\mathbf{I}$ is the $(M-1)$ by $(M-1)$ identity matrix.

It should be noted that we are adding the noise to the evaluated ambiguity function, whereas a more correct model would add it to the received waveform signal prior to the entering the filter bank. In that case, the noise would no longer be Gaussian and would no longer be independent, and this adjustment was made in the interest of simplification.

Concatenating the azimuth measurement to the ambiguity function measurements, we have the complete length M measurement vector

$$\mathbf{z} = [z_{AF}(1), z_{AF}(2), \ldots, z_{AF}(M-1), z_\theta]^T \tag{3.20}$$

Additionally, the azimuth measurement noise is independent of the ambiguity function noise so that

$$\text{Covar}(\mathbf{z}) = \mathbf{R} = \begin{bmatrix} \sigma_{AF}^2 \mathbf{I} & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \tag{3.21}$$

Given the parameters σ_θ^2, σ_{AF}^2, ω_c and the complex waveform envelope $s(t)$, a particular observation model is defined.

Chapter 4

TBD AMBIGUITY FUNCTION MEASUREMENTS

The TBD will be implemented using a particle filter, whose steps are given in Algorithm 1 in Chapter 2. The update step of Algorithm 1 requires evaluating the ambiguity function over every observed point of the infinite delay-Doppler plane, but it leaves us with the choice of which points to observe. This thesis approaches that choice in two ways.

First, we try to approximate making use of the entire plane. For arbitrary ambiguity functions, this could lead to an unbound surface area of the delay-Doppler plane needing to be observed. However, leveraging the properties of the AF of the LFM waveform being used here, we present a solution so that only a finite area need be observed. Additionally, the plane is broken up into a discrete grid with the grid spacing dynamically adapted based on the waveform to result in roughly the same number of measurements across all waveforms.

Second, a fixed grid is established and remains constant over time and choice of waveform. A grid of delay-Doppler points, with fixed grid spacing, is centered on the expected starting position of the target and spans roughly the state space, range and range-rate, the target is expected to occupy.

This section will define the radar waveform class to be used, detail the above approaches to defining the set of AF observations and refine the tracking algorithm.

4.1 Waveforms

Finalizing the observation model requires specifying the waveforms to be used. In particular, the complex waveform envelope, $s(t)$, needs to be defined. In this thesis, all waveform envelopes are linear frequency modulated Gaussian pulses.

$$s(t) = \left(\frac{1}{\pi T^2}\right)^{\frac{1}{4}} \exp\left[-\left(\frac{1}{2T^2} - jb\right)t^2\right] \tag{4.1}$$

Note that the envelope is characterized by the parameters T and b, T is the standard deviation of the Gaussian envelope and is proportional to the duration of the pulse and b is the chirp rate of the LFM waveform. Figure 4.1 displays the LFM chirp $s(t)$ in Equation (4.1) in the time-domain for $T = 1$, $b = 4$. Note that the actual radar simulations uses T values that are much smaller and b values that are much larger than $T = 1$ and $b = 4$.

Combining (4.1) and (3.14) and evaluating the expression, we get

$$|\mathsf{AF}_s(\tau_d, \rho_d)|^2 = \mathsf{exp}\left[-\frac{1}{2}\left(\frac{\tau_d^2}{T^2} + T^2(\rho_d - 2b\tau_d)^2\right)\right] \tag{4.2}$$

Figure 4.2 displays the ambiguity function of $s(t)$ in Equation (4.1) for the same parameters, $T = 1$, $b = 4$.

The Doppler resolution, σ_ρ, is a measure of the spread of the ambiguity function along the $\tau_d = 0$ axis. To characterize it, we first set $\tau_d = 0$ in (4.2) to get

$$|\mathsf{AF}_s(0, \rho_d)|^2 = \mathsf{exp}\left[-\frac{1}{2}\left(T^2\rho_d^2\right)\right] \tag{4.3}$$

Note that (4.3) is now in the form of a Gaussian. The Doppler resolution is defined as what would be the standard deviation of that Gaussian, allowing for a familiar characterization

$$\sigma_\rho = \frac{1}{T} \tag{4.4}$$

Similarly, the delay resolution is a measure of the spread of the ambiguity function along the $\rho_d = 0$ axis. Setting $\rho_d = 0$ in (4.2), we get

$$|\mathsf{AF}_s(\tau_d, 0)|^2 = \mathsf{exp}\left[-\frac{1}{2}\left(\frac{\tau_d^2}{T^2} + 4T^2b^2\tau_d^2\right)\right] \tag{4.5}$$

and

$$\sigma_\tau = \left(\frac{1}{T^2} + 4T^2b^2\right)^{-\frac{1}{2}} \tag{4.6}$$

From (3.2) and (3.5), we can determine the range and range-rate resolution

$$\sigma_r = \frac{c}{2}\sigma_\tau = \frac{c}{2}\left(\frac{1}{T^2} + 4T^2b^2\right)^{-\frac{1}{2}} \tag{4.7}$$

$$\sigma_v = \frac{c}{2}\frac{\sigma_\rho}{\omega_c} = \frac{c}{2}\frac{1}{\omega_c T} \tag{4.8}$$

Later in this thesis, we will make use of a waveform library where each waveform is characterized in terms of σ_r and σ_v, but we need T and b to be able to generate the ambiguity function. Rearranging (4.7) and (4.8), for T and b, we get

$$T = \frac{c}{2}\frac{1}{\omega_c \sigma_v} \tag{4.9}$$

$$b = \pm\frac{\left(\frac{c^2}{4}\frac{1}{\sigma_r^2} - \frac{1}{T^2}\right)^{\frac{1}{2}}}{2T} \tag{4.10}$$

Earlier work [16] makes use of the ambiguity function observations by finding the singular delay-Doppler pair that produces the maximum peak. This corresponds to the maximum likelihood estimator of the target's true delay and Doppler. The problem with that approach was that it is only applicable in high SNR conditions. High noise will cause the ambiguity function side-lobes to create local peaks away from the true values, making the identification of the "true" peak impossible. Figure 4.3 demonstrates this graphically for $\sigma_{AF}^2 = \{0, .01, .5\}$. For comparison, this thesis mostly uses $\sigma_{AF}^2 = \{4, 0.4\}$; clearly, simply choosing the maximum grid point value will not suffice. Later work [19] extended the adaptive waveform approach by trying to incorporate *all* grid-point evaluations of the noisy ambiguity function into the estimate. This is a continuation of, and in some cases a simplification of, that approach.

Because the complex envelope of the waveform has been normalized, all waveforms in the library will have equal power. As the duration of the waveform

is inversely proportional to the range-rate resolution and the bandwidth is inversely proportional to the range resolution, the waveforms will *not* have an equal time-bandwidth product. We will see, however, that the library waveform with the largest time-bandwidth product, $\sigma_r = 10, \sigma_\nu = 3$, is not always the optimal choice.

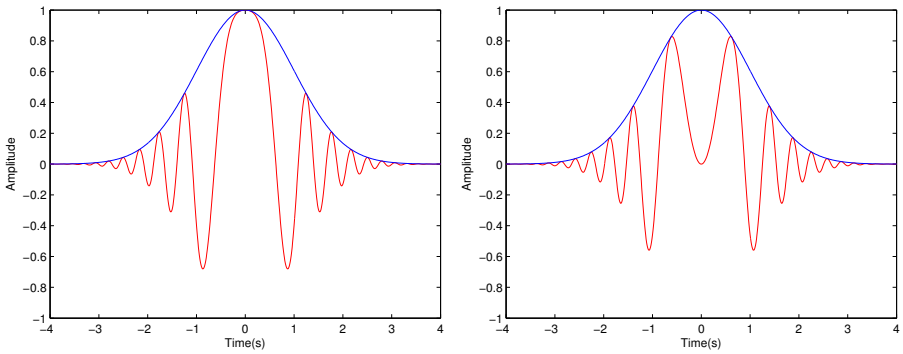


Figure 4.1: Amplitude, real (left) and imaginary (right) parts of $s(t)$ in Equation (4.1) for $T = 1$, $b = 4$

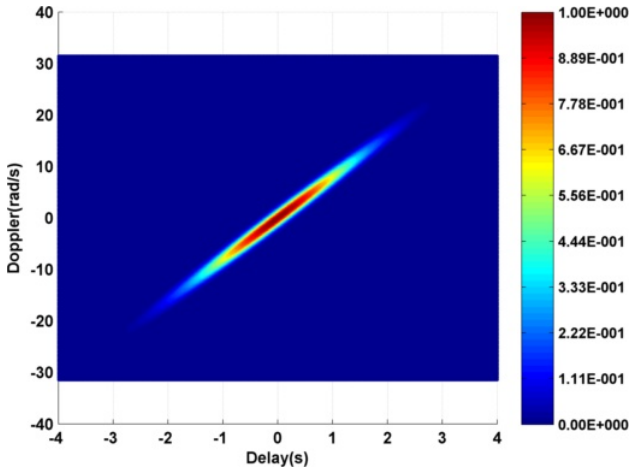
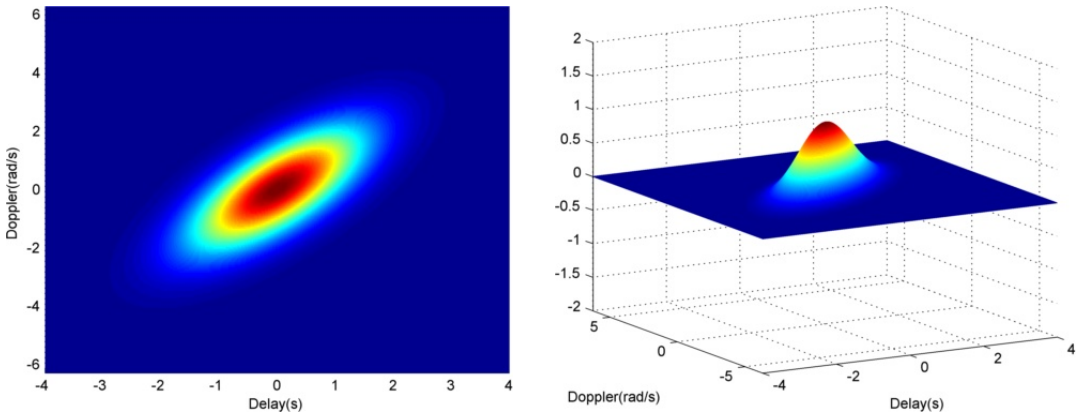
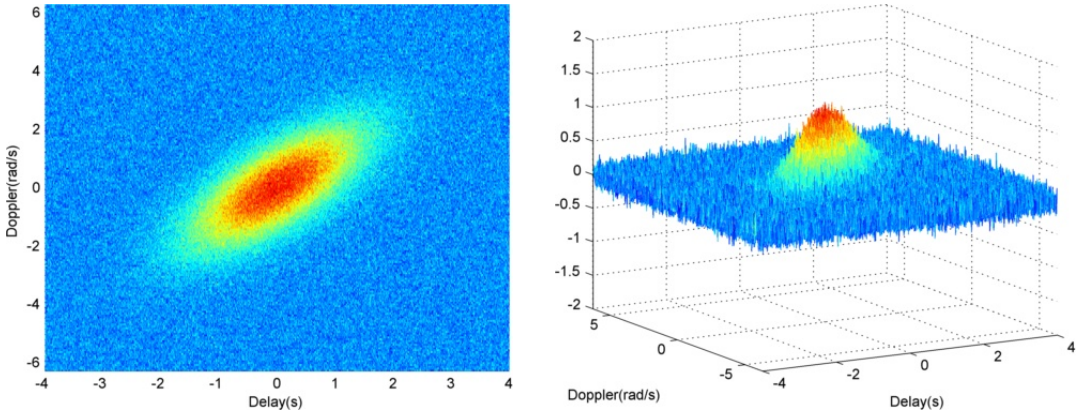


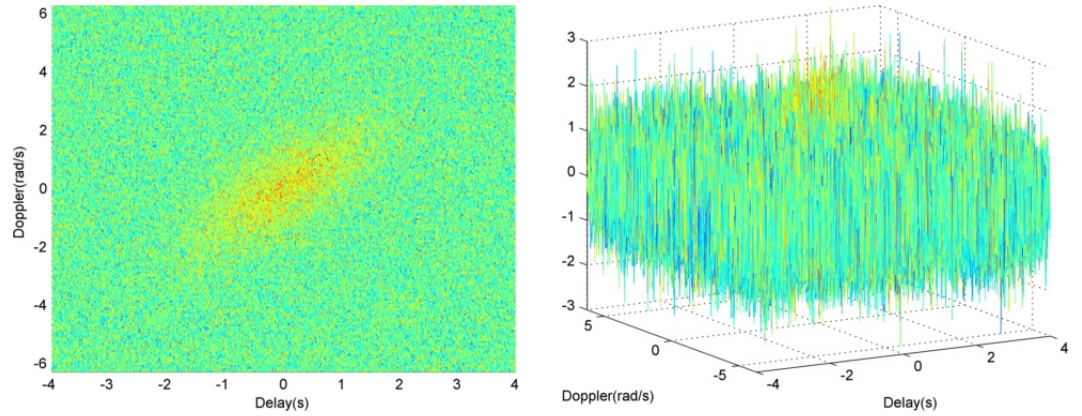
Figure 4.2: Magnitude squared of the LFM signal in Equation (4.1) using $T = 1$, $b = 4$



(a) Ambiguity function with $\sigma^2_{AF} = 0$



(b) Ambiguity function with $\sigma^2_{AF} = 0.01$



(c) Ambiguity function with $\sigma^2_{AF} = 0.5$

Figure 4.3: AF plots of the LFM signal in Equation (4.1) with $T = 1$ and $b = 0.5$, for varying values of σ^2_{AF}. It can be seen that, as the observation noise increases, it gets harder to find a single peak value in the AF domain and the maximum likelihood results in a larger estimation error.

4.2 Entire Delay-Doppler Plane

In order to implement Algorithm 1, we first need to take a closer look at $p(\mathbf{z}_k|\mathbf{x}^n)$, used in the update stage. This is the probability that the measurements made at time step k would be made conditioned that the particle $\mathbf{x}^n$ reflected the true state of the target. From (3.20) and (3.21) we see that the individual measurements of the ambiguity function are independent of each other and the azimuth measurement so that

$$p(\mathbf{z}_k|\mathbf{x}^n) = p(z_{\theta,k}|\mathbf{x}^n)p(\mathbf{z}_{AF,k}|\mathbf{x}^n) = p(z_\theta|\mathbf{x}^n)\prod_{d\in D} p(z_{AF,k}(d)|\mathbf{x}^n) \tag{4.11}$$

$$d \triangleq (d_\tau, d_\rho) \tag{4.12}$$

where d is a coordinate in the delay-Doppler plane, D is the set of all points in the plane to be evaluated, and $z_{AF,k}(d)$ is the measurement taken from the noisy ambiguity function of $s(t)$. Again dropping the subscript k for simplicity, and noting that all noise is Gaussian, we can directly present the probability of the azimuth measurement

$$\begin{aligned} p(z_\theta|\mathbf{x}^n) &= (2\pi\sigma_\theta^2)^{-0.5}\text{exp}\left[-\frac{1}{2\sigma_\theta^2}(z_\theta - \mu_\theta)^2\right] \\ \mu_\theta &\triangleq \text{tan}^{-1}(y^n/x^n) \end{aligned} \tag{4.13}$$

For the ambiguity function measurement probability, we need to define the set of coordinates, D, within the delay-Doppler plane that will be taken as measurements. Ideally, we would want every point to be evaluated, but as the ambiguity function is continuous, we must settle for a discrete sampling of points. We start with a regular grid spacing covering the entire plane

$$D = \{(M\delta_\tau, N\delta_\rho)|M = -\infty \text{ to } +\infty, N = -\infty \text{ to } +\infty\} \tag{4.14}$$

where δ_τ and δ_ρ are the constant delay and Doppler grid spacings, and M and N are integers. Clearly, this is not any better as we are still have an infinite number of points to evaluate. Thankfully, [6] gives us a solution for reducing the number of necessary evaluation points.

As the particle weights in Algorithm 1 are normalized before use, it is only necessary to determine their values to within a constant of proportionality. In this case, we divide by the probability of observing the ambiguity measurements conditioned on there being no target. This substitutes the likelihood ratio for the ambiguity measurements for their actual probability, allowing the update step of the algorithm to be rewritten

$$w^n = w^n \frac{p(\mathbf{z}|\mathbf{x}^n)}{p(\mathbf{z}_{AF})} = w^n p(z_\theta|\mathbf{x}^n) \frac{p(\mathbf{z}_{AF}|\mathbf{x}^n)}{p(\mathbf{z}_{AF})} \tag{4.15}$$

Expanding the AF likelihood ratio, similar to (4.11), and substituting in the fact the noise is Gaussian we get

$$\begin{aligned} \frac{p(\mathbf{z}_{AF}|\mathbf{x}^n)}{p(\mathbf{z}_{AF})} &= \prod_{d \in D} \frac{p(z_{AF}(d)|\mathbf{x}^n)}{p(z_{AF}(d))} \\ &= \prod_{d \in D} \exp\left[-\frac{1}{2\sigma_{AF}^2} \left\{ (z_{AF}(d) - |\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n)|^2)^2 - z_{AF}^2(d) \right\} \right] \end{aligned} \tag{4.16}$$

where τ^n and ρ^n are the delay and Doppler positions of the nth particle.

$$\begin{aligned} \tau^n &= \frac{c}{2}\sqrt{(x^n)^2 + (y^n)^2} \\ \rho^n &= -\frac{2\omega_c}{c} \frac{x^n \dot{x}^n + y^n \dot{y}^n}{\sqrt{(x^n)^2 + (y^n)^2}} \end{aligned} \tag{4.17}$$

Noting that any grid point where $\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n) = 0$ results in exp(0), (4.16) can be simplified by evaluating only over the grid points where $\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n)$

is non-zero

$$\frac{p(\mathbf{z}_{AF}|\mathbf{x}^n)}{p(\mathbf{z}_{AF})}$$
$$= \prod_{d\in D^n} \exp\left[-\frac{1}{2\sigma_{AF}^2}\left\{(z_{AF}(d) - |\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n)|^2)^2 - z_{AF}^2(d)\right\}\right]$$
$$= \prod_{d\in D^n} \exp\left[\frac{1}{2\sigma_{AF}^2}|\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n)|^2\left\{|\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n)|^2 - 2z_{AF}(d)\right\}\right] \tag{4.18}$$

with the evaluated points now being taken from D^n instead of D, where D^n is the set of all coordinates where $\mathsf{AF}_s(d_\tau - \tau^n, d_\rho - \rho^n) > 0$, or, all coordinates where the expected ambiguity function would be non-zero if the nth particle represented the correct target state.

The ambiguity function never actually equals zero, but as d moves away from (τ^n, ρ^n), it converges to zero. Looking at (4.2), we can establish a constant value contour of the ambiguity function by finding the values of τ_d and ρ_d that produce the same value for the exponent, or

$$\alpha^2 = G(\tau_d, \rho_d) = \frac{\tau_d^2}{T^2} + T^2(\rho_d - 2b\tau_d)^2 \tag{4.19}$$

D^n can then be approximated with

$$D^n = \{(\tau_d, \rho_d)|\forall G(\tau_d, \rho_d) < \alpha^2\} \tag{4.20}$$

Figure 4.4 demonstrates the results of choosing only those coordinates with $G(\tau_d, \rho_d) < \alpha^2$. By setting $\alpha = 3$, we are guaranteed that all coordinates with $|\mathsf{AF}_s(\tau_d, \rho_d)|^2 > \exp(-9/2) \approx 0.01$ are included in determining a particle's updated weight value. This limits the total number of observations necessary and allows (4.16) to be evaluated.

The new total *effective* grid, D, is the set of all coordinates that will have to be evaluated for a particular time step. It is simply the union of the coordinates required for every particle update.

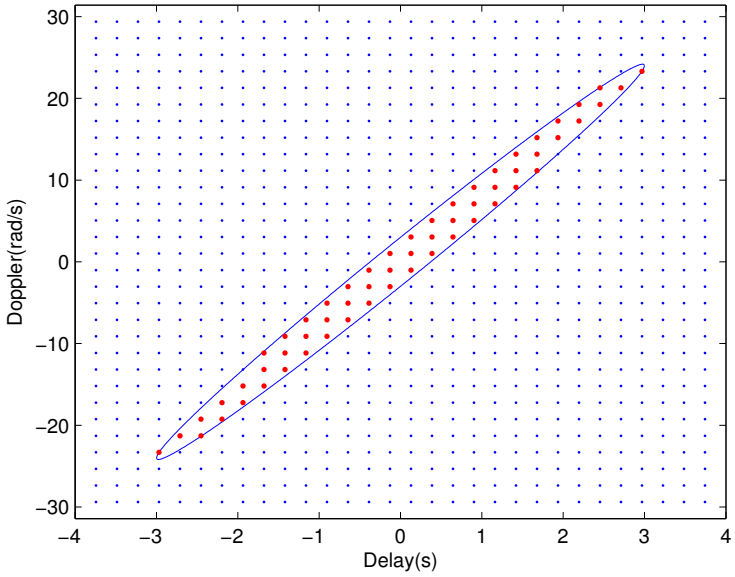


Figure 4.4: A constant contour (blue ellipse) is drawn for $G(\tau_d, \rho_d) = \alpha^2$. D^n is determined by finding the coordinates (red) where $G(\tau_d, \rho_d) < \alpha^2$. The example here is for the AF of an LFM chirp with T=1 and b=4, and we used $\alpha = 3$.

$$D = \bigcup_{n=1}^{N} D^n \tag{4.21}$$

With varying waveform parameters, T and b, the extent of the ambiguity function can change dramatically. To ensure that every waveform produces approximately the same number of valid coordinates, the grid spacing is determined as a function of the waveform parameters. The grid can be made denser or more sparse by dividing with a larger or smaller number. In this thesis, the divisor is always 4.

$$\begin{aligned} \delta_\tau &= T/4 \\ \delta_\rho &= 1/(4T) \end{aligned} \tag{4.22}$$

With the grid fully defined, a method of evaluating the necessary probabilities derived and the process state described in (2.24), the particle filter is more specifically detailed in Algorithm 2. This algorithm was implemented in MATLAB

using the parameters in Table 4.1 and the results for one instance of that simulation over 15 time steps is shown in Figure 4.5. It is important to note that the set of waveform parameters chosen for these simulations are not compatible with the time-bandwidth values used for a real radar application. They would need to be changed to simulate an actual radar installation.

Algorithm 2 Particle Filter TBD Using Full Plane Measurements

Assume an initial estimate of value μ_s
for $n = 1$ to N **do**
  Draw $\mathbf{x}^n$ from $p(\mathbf{x}_0)$
  $w^n = 1/N$
end for
for $k = 1$ to K **do**
  Determine Effective Grid:
  $D = \{\emptyset\}$
  for $n = 1$ to N **do**
    $D^n = \{(\tau_d, \rho_d) | \forall G(\tau_d, \rho_d) < \alpha^2\}$
    $D = D \bigcup D^n$
  end for
  Measure:
  - Measure azimuth, z_θ
  - Take measurements, $\mathbf{z}_{AF}$, for all points defined in D
  for $n = 1$ to N **do**
    Update:
    $w_1 = p(z_\theta | \mathbf{x}^n)$
    $w_2 = \prod_{d \in D^n} \frac{p(z_{AF}(d) | \mathbf{x}^n)}{p(z_{AF}(d))}$
    $w^n = w^n w_1 w_2$
  end for
  Sum the weights: $w_{total} = \sum_{n=1}^{N} w^n$
  for $n = 1$ to N **do**
    Normalize: $w^n = w^n / w_{total}$
  end for
  Estimate: $\hat{\mathbf{x}}_k = \sum_{n=1}^{N} w^n \mathbf{x}^n$
  Re-sample particles
  for $n = 1$ to N **do**
    Draw process noise $\mathbf{y}$ from $p(\mathbf{w})$
    Advance: $\mathbf{x}^n = \mathbf{F}\mathbf{x}^n + \mathbf{y}$
  end for
end for

Carrier frequency	ω_c	1.885×10^{10}
Number of particles	N	1000
Re-sample threshold	N_{min}	200
Process noise power	σ_v^2	5
Seconds between time steps	Δt	1
Initial distribution mean	μ_s	$[5000, 5000, 10, 10]^T$
Initial distribution covariance	$\mathbf{C}_s$	diag$([1, 1, 16, 16])$
Azimuth measurement noise power	σ_θ^2	$\approx 7.62 \times 10^{-7}$
AF measurement noise power	σ_{AF}^2	4
Waveform range resolution	σ_r	30
Waveform range-rate resolution	σ_ν	4

Table 4.1: Parameters used for the results in Figure 4.5 and the assumed parameters when not specified for other simulations.

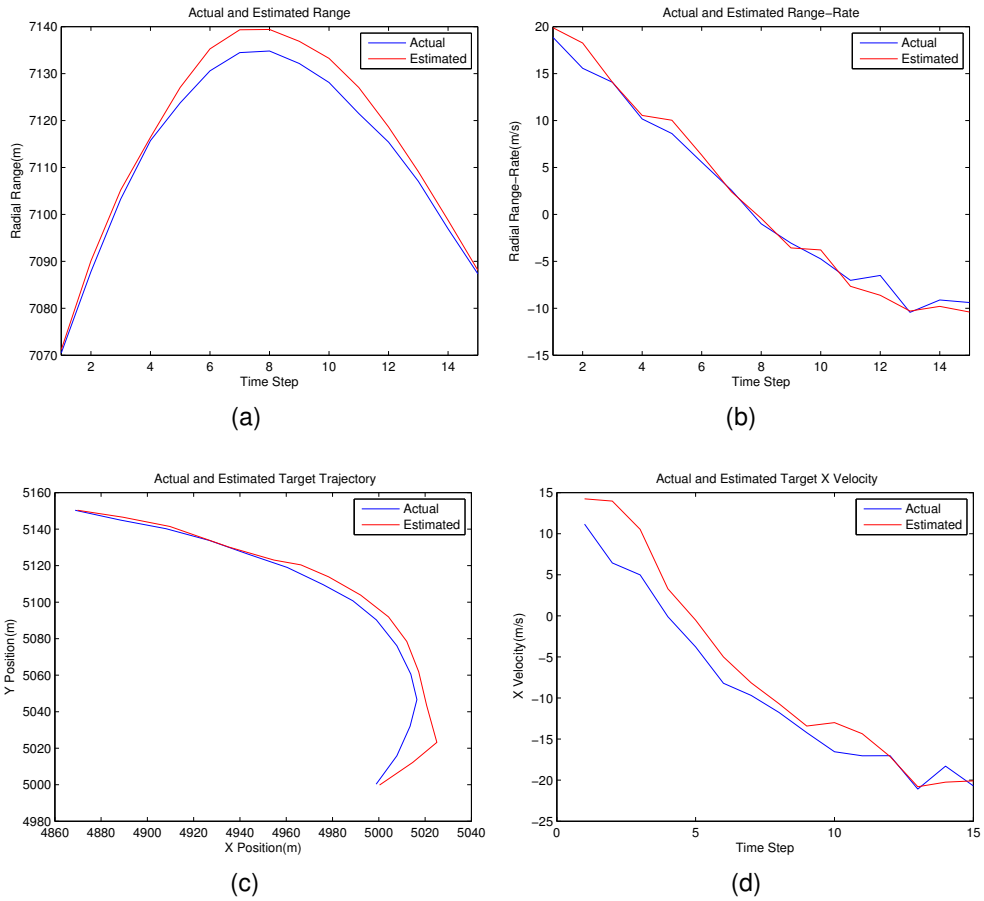


Figure 4.5: Example PF-TBD simulation results for the parameters in Table 4.1. The actual (blue) and estimated (red) target radial range from the radar station (a), rate of change of radial range (b), position in 2-D space (c), and x velocity (d) are shown.

4.3 Fixed Grid

As an alternative to using the entire delay-Doppler plane with variable grid spacing derived from the waveform parameters, we can alternatively predefine a fixed and finite number of grid points. This is a practical restriction as it recognizes the limitations of the matched filter bank used to produce the ambiguity function measurements. Limits along the Doppler axis correspond to bandwidth limitations of the received signal, and limits along the delay axis correspond to having a finite length recording of the received waveform.

In the fixed grid case, the delay-Doppler grid points that will make up the ambiguity function measurements are fixed at

$$\begin{gathered} \delta_r = 20, \delta_\nu = 6 \\ D = \{(r_0, \nu_0) + (M\delta_r, N\delta_\nu) | M = -10:10, N = -10:10\} \end{gathered} \tag{4.23}$$

where (r_0, ν_0) represent the range and range-rate coordinate of μ_s. This centers the grid over the initial expected range and range-rate of the target. The tracking algorithm remains the same as in the entire plane case, simply a different set of delay-Doppler observation points, D, is used.

Chapter 5

TBD DYNAMIC WAVEFORM DESIGN

Previous sections have described the tracking problem and presented the TBD-PF algorithm for tracking under low SNR conditions. In this section we will derive the means of predicting expected errors at each time step and use the results of that derivation to establish a waveform selection algorithm.

5.1 Calculating and Predicting Error

At each time step, the error is normally defined as the difference between the actual state values and the estimated state values

$$\text{err}_k \triangleq \mathbf{x}_k - \hat{\mathbf{x}}_k \tag{5.1}$$

Here, we are only interested in the squared error, and we wish to reduce it to a scalar for easier comparison later. We introduce an error weight row vector, $\mathbf{g}$, and define the squared error to be

$$\text{err}_k^2 \triangleq \mathbf{g}(\mathbf{x}_k - \hat{\mathbf{x}}_k)^2 \tag{5.2}$$

If we know the covariance matrix of $\mathbf{x}_k$, and our estimate is the mean of the distribution of $\mathbf{x}_k$

$$\begin{aligned} \mathbf{M}_k &\triangleq \text{Covar}(\mathbf{x}_k) \\ \hat{\mathbf{x}}_k &= E[\mathbf{x}_k] \end{aligned} \tag{5.3}$$

we can also determine the mean squared error as

$$\begin{aligned} \mathbf{M}_k &\triangleq \text{Covar}(\mathbf{x}_k) \\ \overline{\text{err}}^2 &= E[\text{err}_k^2] = \mathbf{g}(\text{diag}(\mathbf{M}_k)) \end{aligned} \tag{5.4}$$

We now introduce new notation to distinguish between the *a priori* and *a posteri* covariance matrices of $\mathbf{x}_k$. The *a priori* covariance will be denoted $\mathbf{M}_{k+1|k}$ and is the forward transformed covariance of the previous *a posteri* covariance, $\mathbf{M}_{k|k}$, with

$$\mathbf{M}_{k+1|k} = \text{Covar}(\mathbf{F}\mathbf{x}_k + \mathbf{Q}) = \mathbf{F}\mathbf{M}_{k|k}\mathbf{F}^T + \mathbf{Q} \tag{5.5}$$

What we would like to know now, is how to predict what the squared error will be at the next time step knowing only what the covariance is for the current time step. We start by establishing the lower bound for the covariance matrix. From [17], we are given a recursive relation we use to determine the predicted posterior covariance matrix

$$\mathbf{J}_{k+1} = \mathbf{D}_k^{22} - \mathbf{D}_k^{21}(\mathbf{J}_k + \mathbf{D}_k^{11})^{-1}\mathbf{D}_k^{12} \tag{5.6}$$

where $\mathbf{J}_k$ is the Fisher information matrix at time step k and $\mathbf{P}_k \triangleq \mathbf{J}_k^{-1}$ is the CRLB for any estimator of $\mathbf{x}_k$. For the state and process models used here, the various $\mathbf{D}$ matrices reduce to

$$\begin{gathered}
\mathbf{D}_k^{11} = E\{[\nabla_{\mathbf{x}_k} f^T(\mathbf{x}_k)]\mathbf{Q}^{-1}[\nabla_{\mathbf{x}_k} f^T(\mathbf{x}_k)]^T\} = \mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F} \\
\mathbf{D}_k^{12} = [\mathbf{D}_k^{21}]^T = -E\{\nabla_{\mathbf{x}_k} f^T(\mathbf{x}_k)\}\mathbf{Q}^{-1} = -\mathbf{F}^T\mathbf{Q}^{-1} \\
\mathbf{D}_k^{22} = \mathbf{Q}^{-1} + E\{[\nabla_{\mathbf{x}_{k+1}} h_l^T(\mathbf{x}_{k+1})]\mathbf{R}^{-1}[\nabla_{\mathbf{x}_{k+1}} h_l^T(\mathbf{x}_{k+1})]^T\} = \mathbf{Q}^{-1} + \mathbf{J}_{exp,k+1} \\
\nabla_x \triangleq \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial \dot{x}}, \frac{\partial}{\partial \dot{y}}\right]^T
\end{gathered} \tag{5.7}$$

where the subscript of h_l denotes that the observations are dependent on the waveform parameters, being chosen from a library of L options to be detailed later. The above then allows (5.6) to be written as

$$\mathbf{J}_{k+1} = \mathbf{J}_{exp,k+1} + \mathbf{Q}^{-1} - \mathbf{Q}^{-1}\mathbf{F}(\mathbf{J}_k + \mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F})^{-1}\mathbf{F}^T\mathbf{Q}^{-1} \tag{5.8}$$

and applying the matrix inversion lemma

$$\mathbf{J}_{k+1} = \mathbf{J}_{exp,k+1} + (\mathbf{F}\mathbf{J}_k^{-1}\mathbf{F}^T + \mathbf{Q})^{-1} \tag{5.9}$$

The term $\mathbf{J}_{exp,k}$ is the *expected* amount of new information at time step k before the actual observation has taken place, and is taken with respect to $p(\mathbf{x}_k, \mathbf{z}_k)$. The *actual* amount of information gathered at step k depends on the specific $\mathbf{z}_k$ and is not represented in any of the above equations. The final result of (5.9), $\mathbf{J}_{k+1}$, can then be seen as a forward transformation of current information, $\mathbf{J}_k$, plus the expected new information, $\mathbf{J}_{exp,k+1}$.

Using (5.9) we can determine what the predicted covariance would be, starting with a know covariance, by substituting $\mathbf{M}_{k|k} = \mathbf{P}_k = \mathbf{J}_k^{-1}$

$$\mathbf{J}_{k+1} = \mathbf{J}_{exp,k+1} + (\mathbf{F}\mathbf{M}_{k|k}\mathbf{F}^T + \mathbf{Q})^{-1} \tag{5.10}$$

substituting in (5.5)

$$\mathbf{J}_{k+1} = \mathbf{J}_{exp,k+1} + \mathbf{M}_{k+1|k}^{-1} \tag{5.11}$$

and taking the inverse to define

$$\tilde{\mathbf{P}}_{k+1,l} = (\mathbf{J}_{exp,k+1} + \mathbf{M}_{k+1|k}^{-1})^{-1} \tag{5.12}$$

where the l in the subscript to $\tilde{\mathbf{P}}_{k+1,l}$ indicates that the predicted covariance is dependent on the chosen waveform.

Note that even though $\mathbf{J}_k^{-1}$ was taken to be equal to $\mathbf{M}_{k|k}$, $\mathbf{J}_{k+1}^{-1}$ is *not* taken to be equal to $\mathbf{M}_{k+1|k+1}$. This is because $\mathbf{M}_{k|k}$ is the covariance of $\mathbf{x}_k$ taking all observations up to time step k into account, $\mathbf{z}_{1:k}$, while $\tilde{\mathbf{P}}_{k+1,l}$, the output of (5.12), is the expected covariance incorporating expected information, but without knowing what the observation at time $k+1$ actually is. If we were to run many simulations, we would find that, instead of equality,

$$\tilde{\mathbf{P}}_{k+1,l} = E[\mathbf{M}_{k+1|k+1}] \tag{5.13}$$

Put another way, we can use (5.9) to determine what the predicted covariance will be for any future time step, but we must start the recursion with a known

covariance. When the known covariance is for time step k and we are only looking for the predicted covariance one step forward, it reduces to (5.12).

We can now predict the minimum possible *expected* squared error, also the mean squared error, to be

$$E[\text{err}_{k+1}^2] = \overline{\text{err}}_{k+1}^2 = \mathbf{g}(\text{diag}(\tilde{\mathbf{P}}_{k+1,l})) \tag{5.14}$$

Turning now to the evaluation of $\mathbf{J}_{exp,k+1}$, we note it is an expectation over a continuous probability density we do not have access to. Making use, once again, of the proxy density afforded us by the particle filter, it can be approximated as

$$\mathbf{J}_{exp,k+1} \approx \sum_{n=1}^{N} w^n [\nabla_{\mathbf{x}_{k+1}^n} h_l^T(\mathbf{x}_{k+1}^n)] \mathbf{R}^{-1} [\nabla_{\mathbf{x}_{k+1}^n} h_l^T(\mathbf{x}_{k+1}^n)]^T \tag{5.15}$$

Recalling that $\mathbf{R}$ is diagonal and separating out the azimuth measurement from the ambiguity function measurements, we get

$$\begin{gathered} \mathbf{J}_{exp,k+1} \approx \frac{1}{\sigma_\theta^2} \sum_{n=1}^{N} w^n \mathbf{D}_\theta \mathbf{D}_\theta^T + \frac{1}{\sigma_{AF}^2} \sum_{n=1}^{N} w^n \mathbf{D}_{AF} \mathbf{D}_{AF}^T \\ \mathbf{D}_\theta = \nabla_{\mathbf{x}_{k+1}^n} h_\theta^T(\mathbf{x}_{k+1}^n) \\ \mathbf{D}_{AF} = \nabla_{\mathbf{x}_{k+1}^n} h_{AF,l}^T(\mathbf{x}_{k+1}^n) \end{gathered} \tag{5.16}$$

Unfortunately, $\mathbf{J}_{exp}$ is a function both of the waveform being used and the distribution of the particles. This prevents any closed form solution. Instead, it will have to be evaluated at each time step for each simulation run, if there is a desire to predict the error. We can, however, make a first order approximation of the best case scenario by setting $\mathbf{J}_0 = \mathbf{C}_s^{-1}$ and evaluating $\mathbf{J}_{exp}$ only for $\mathbf{x}_k$. By recursively applying (5.9), we can calculate an approximation to the expected performance for a given error weight vector and given waveform for any future step k. Additionally, by using the adaptive waveform selection algorithm to be described in the next section, we can also approximate the expected performance using the adaptive algorithm's choices. In Section 6, this technique is used to estimate expected performance which then serves as a reference point for the actual simulation results.

5.2 Waveform Selection

With the results from the previous section, we are in a position to construct an algorithm for choosing a particular waveform from a library of waveforms at each time step, as presented in Algorithm 3.

As the particle filter progresses normally, the particles themselves reflect $p(\mathbf{x}_k|\mathbf{z}_{1:k-1})$ at the start of the main loop in Algorithm 2. The covariance of the particles at this point is then also the *a priori* covariance for step k, $\mathbf{M}_{k|k-1}$. This covariance is used along with an estimated $\mathbf{J}_{exp}$ for each waveform option to determine an expected squared error according to (5.16). Finally, the minimum estimated error is found and those waveform parameters that produced it are used for the next set of measurements, $\mathbf{z}_k$. The only adjustment necessary to have Algorithm 2 perform optimal waveform selection is to insert a call to Algorithm 3 immediately prior to the 'Determine Effective Grid' step.

Algorithm 3 Waveform Selection

- At time step k, obtain state estimate $\hat{\mathbf{x}}_k = \sum_{n=1}^{N} w_k^n \mathbf{x}_k^n$
- Compute $\mathbf{M}_{k|k} = \sum_{n=1}^{N} w_k^n (\mathbf{x}_k^n - \hat{\mathbf{x}}_k)(\mathbf{x}_k^n - \hat{\mathbf{x}}_k)^T$

for $l = 1$ to L **do**

- Calculate predicted covariance $\tilde{\mathbf{P}}_{k+1,l}$ using (5.12)
- Obtain predicted MSE $e_{k+1,l} = \mathbf{g}(\text{diag}(\mathbf{P}_{k+1,l}))$ using (5.14)

end for

- Choose signal $s(t) = \arg\{\min_l e_{k+1,l}\}$ from the library that minimizes the predicted MSE

Chapter 6

SIMULATION RESULTS

6.1 Entire Delay-Doppler Plane

Section 5.1 provided an algorithm for predicting the expected performance of the tracking algorithm for both fixed and adaptive waveform choices. The left hand side of Figures 6.1 to 6.8 show the results of this weighted MSE of the expected performance (MSE-EP). The waveform library used was composed of 10 unique range resolutions and 10 unique range-rate resolutions, as presented in Table 6.1, for a total of 100 unique waveform options. The three best performing fixed waveforms from that library are plotted along with the adaptive waveform results. For each figure, green represents $\sigma_r = 10, \sigma_\nu = 3$, blue represents $\sigma_r = 10, \sigma_\nu = 30$, red represents $\sigma_r = 30, \sigma_\nu = 3$, and black represents the results of the adaptive algorithm.

The simulations were run over varying parameters. Two different starting covariance matrices were used; one "small", one "large". Two Δt's were used; 1 and 0.2. And four different error weights: one focusing on position only, one on velocity only, one weighting the two equally and a final one weighting them so that the position and velocity portions of the error are scaled closer to each other. In all cases, the SNR was set to approximately -6 dB. Again, Figures 6.1 to 6.8 show the results for the various combinations of these parameters. One key observation is that the error weight plays an important role in the relative performance of the adaptive algorithm when compared to any fixed waveform. We see that in most cases where we focus only on position or velocity, a fixed waveform does as well as the adaptive algorithm. However, when the error weight works to "balance" the contributions from the position delta or the velocity delta to the overall error,

there is a clear advantage to being able to change waveforms over time. This is explored further with additional test cases in the following section using the fixed grid. Additional overall observations follow.

There are a few important observations to be noted about the MSE-EP (left hand side) plots. First, the best fixed waveform is not always the same under different simulation parameters. For example, the waveform with the smallest range resolution (blue) has the lowest error in Figure 6.7a, the smallest range-rate resolution (red) has the lowest in Figure 6.7c, and setting both to the smallest option (green) has the lowest in Figure 6.4c. This is an important, although not surprising, result because if one fixed waveform always had the best performance, there would be no need for agile waveform selection.

Second, the adaptive waveform performance is sometimes actually worse than the best fixed waveform, although not by much. Figures 6.1a and 6.4c demonstrate this. This is a result of the adaptive waveform algorithm being myopic, that is, it is only taking the immediate next time step into consideration. There are times when, for example, focusing on range-rate will give you the best error for the immediate next time step, while ignoring range can establish an error that will decay with time, Figure 6.4c, or that will remain persistent, Figure 6.1a. In practice, we will see that the errors introduced as a result of using a particle filter to approximate an efficient estimator effectively obscures these minor differences. In the event that these error differences become significant, perhaps with the use of very large particle counts, the algorithm would need to be extended to look more than one time step in advance.

Finally, there are cases where the adaptive performance not only matches the best fixed waveform, but actually exceeds it. Figures 6.4c and 6.8c demonstrate this. In the cases examined, the adaptive performance will eventually converge on

the best fixed performance. The predicted advantage of the adaptive waveform algorithm is then two fold: not only will the algorithm adapt itself to perform as well as the best fixed waveform for any set of simulation parameters, it can also reach any best asymptotic performance sooner than any fixed waveform.

We can now compare actual simulation results against the MSE-EP. The right hand side of Figures 6.1 to 6.8 display the weighted MSE from 200 simulations while using the same parameters that produced the plots to their left. The simulation results are clearly not as smooth as the predicted errors, which is a result of running only a finite number of iterations in the simulations. Figure 6.9 demonstrates convergence for additional iterations. The general pattern of the fixed waveform simulation results, however, reasonably matches their predicted performance. This justifies the use of the predicted covariance in the adaptive waveform selection algorithm. For each parameter set, the relative predicted performance of the adaptive algorithm is mirrored in the simulation results. Finally, Figure 6.10 demonstrates the adaptive selection of waveform parameters for various error weights.

The principle result of this thesis is that the two predicted advantages of the adaptive algorithm are also present in the simulations results. First, that, within expected variance of the simulation error due to a finite number of iterations, the adaptive waveform performance is no worse than the best fixed waveform performance under all simulation parameters; and second, under some simulation parameters, the adaptive performance exceeds the best fixed waveform performance. The later simulation parameters are demonstrated in Figures 6.8d and 6.14d.

σ_r values	10.0	11.3	12.8	14.4	16.3	18.4	20.8	23.5	26.6	30.0
σ_ν values	3.0	3.9	5.0	6.5	8.3	10.8	13.9	18.0	23.2	30.0

Table 6.1: Range and range-rate values used to compose the 100 entry waveform library.

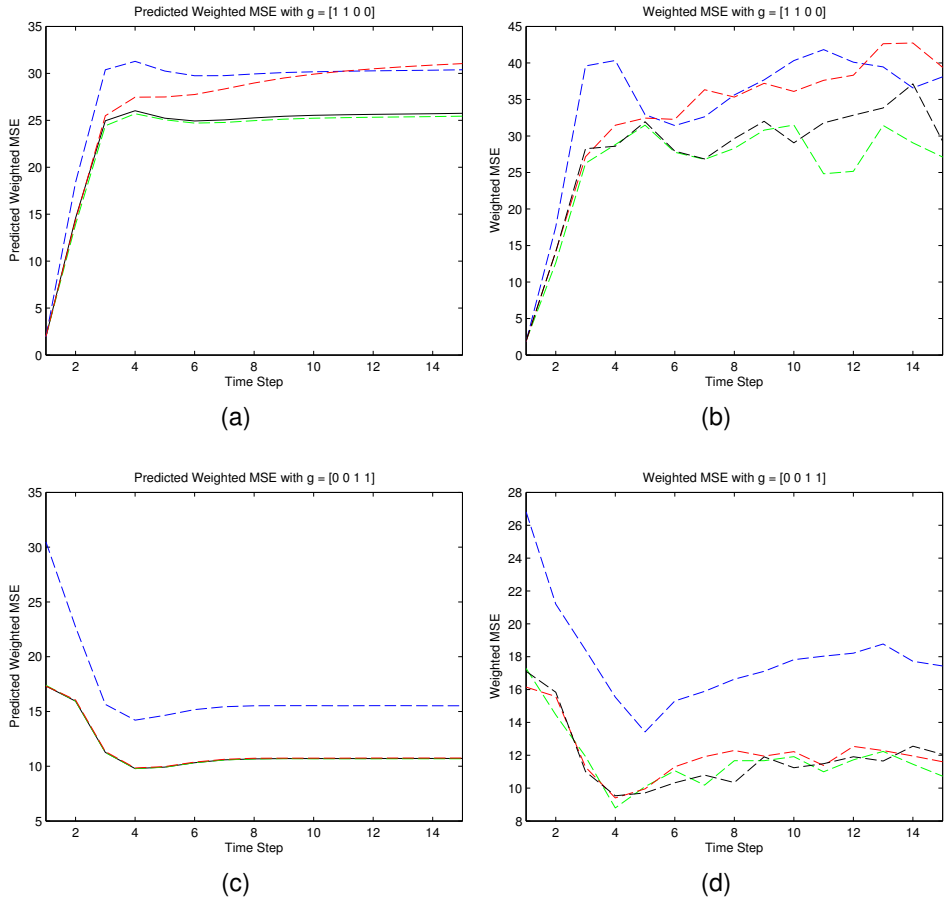


Figure 6.1: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([1, 1, 16, 16]), \Delta t = 1$.

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

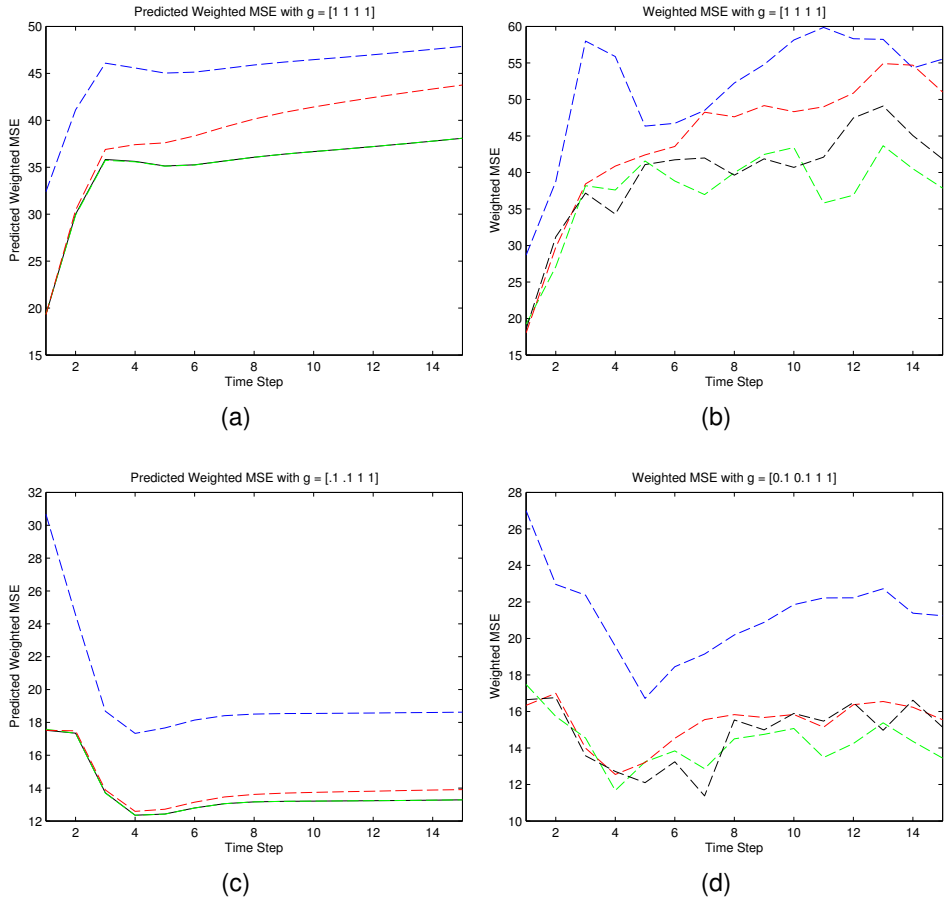


Figure 6.2: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([1, 1, 16, 16]), \Delta t = 1$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

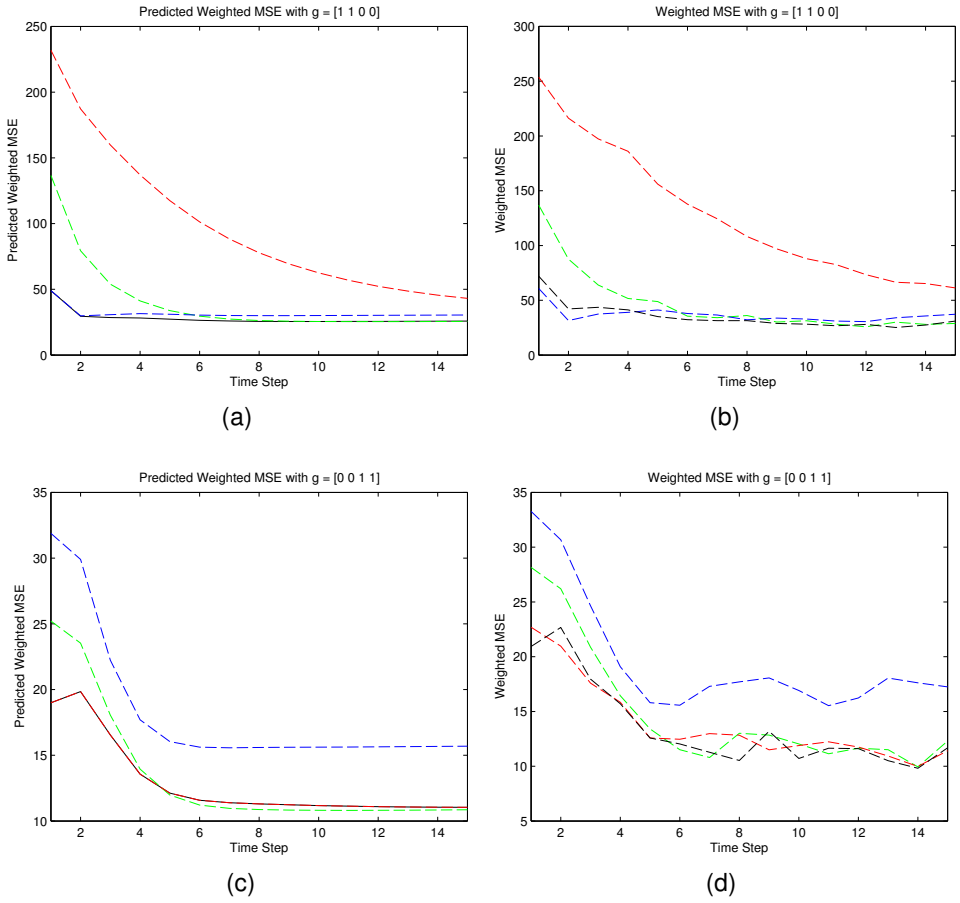


Figure 6.3: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = 1$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

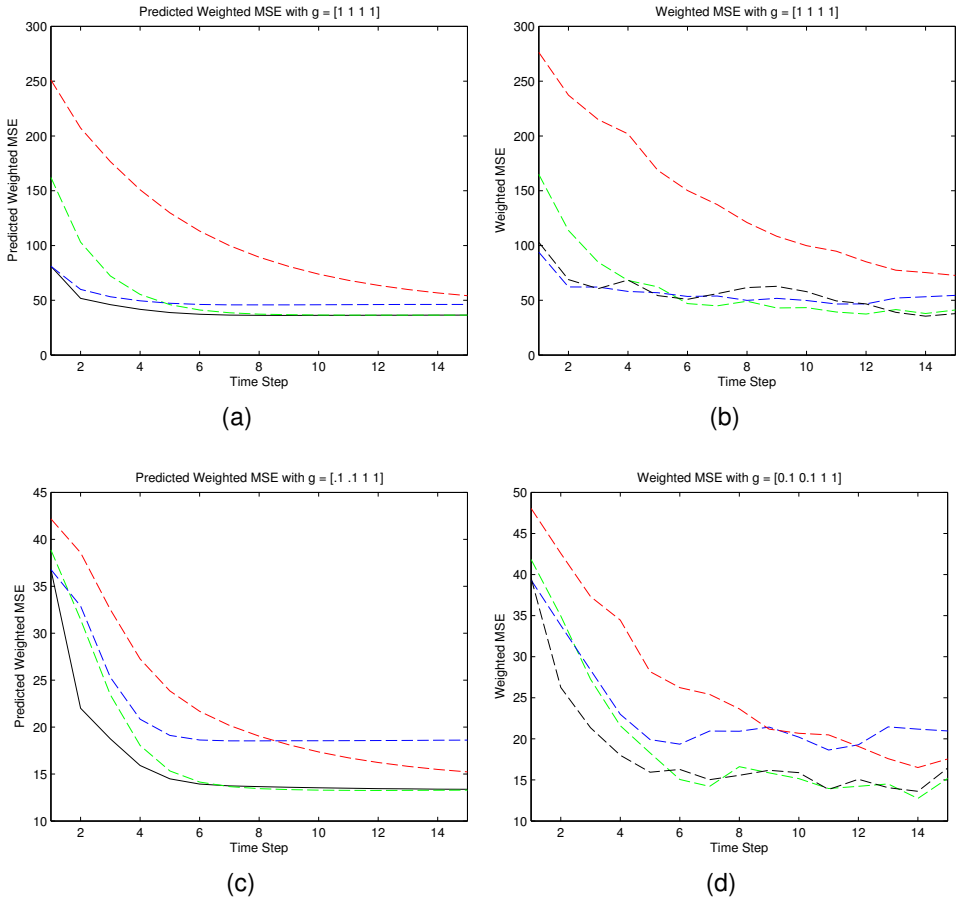


Figure 6.4: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = 1$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

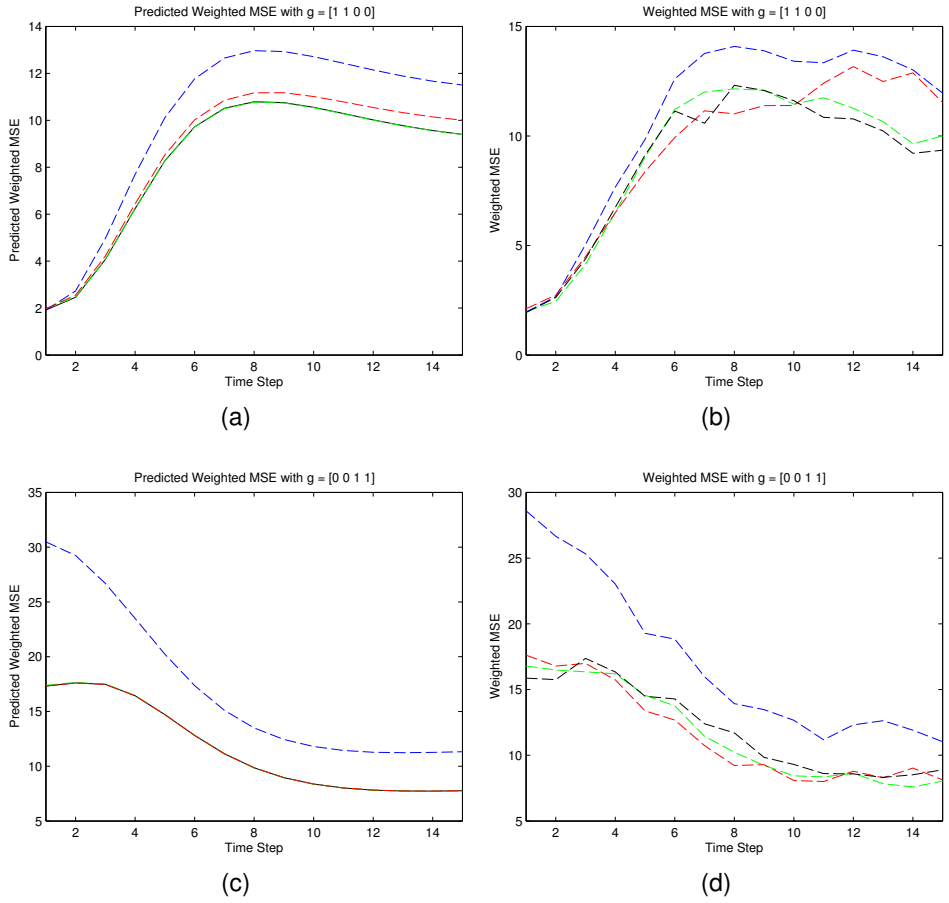


Figure 6.5: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([1, 1, 16, 16]), \Delta t = .2$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

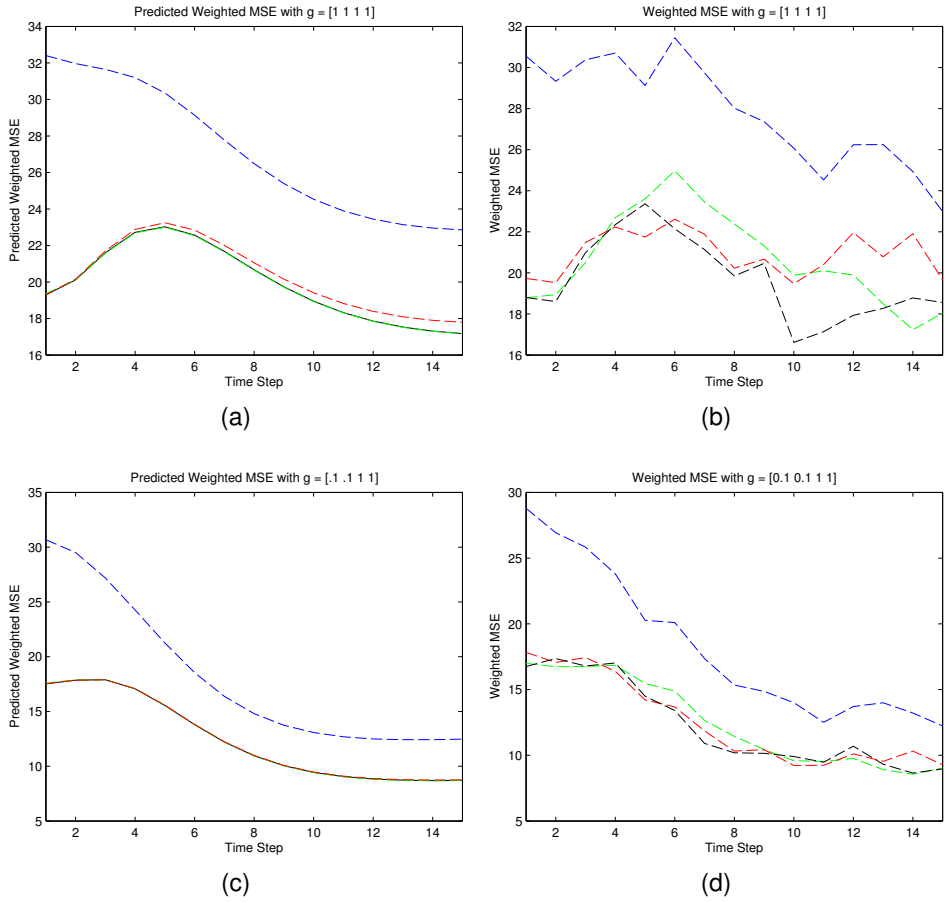


Figure 6.6: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([1, 1, 16, 16]), \Delta t = .2$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

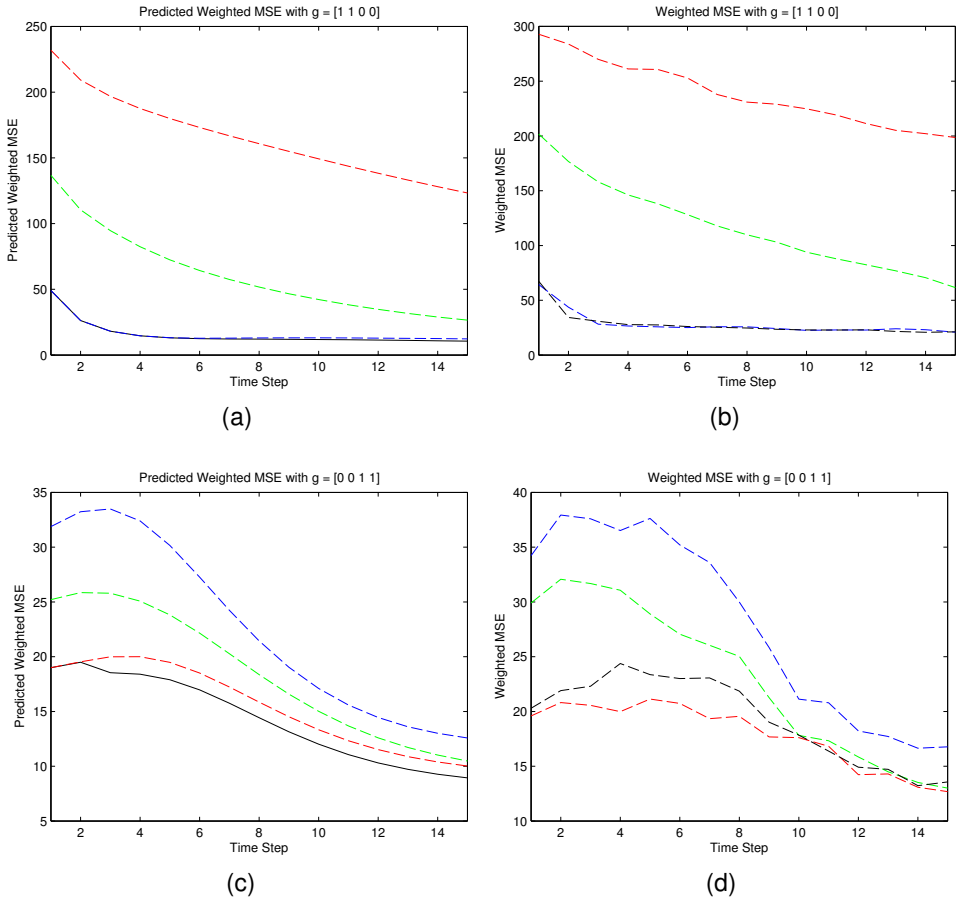


Figure 6.7: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = .2$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

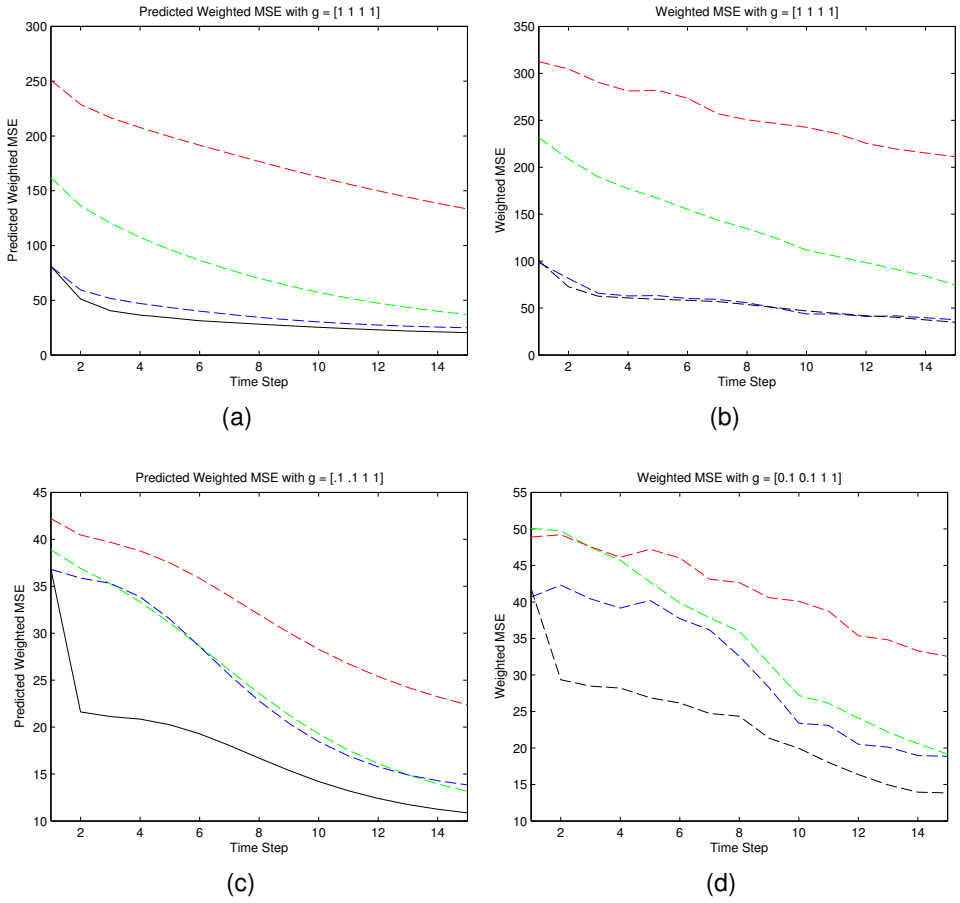
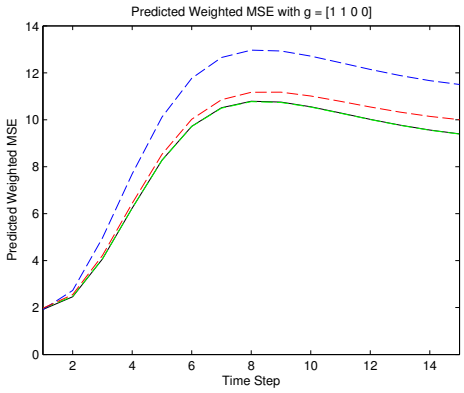
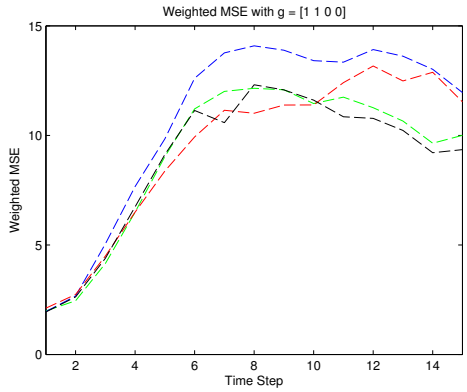


Figure 6.8: Entire plane MSE-EP (left) and simulated (right) results over 200 iterations for $N = 300, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = .2$

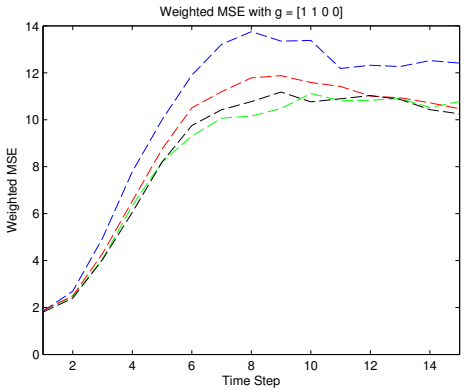
	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive



(a)

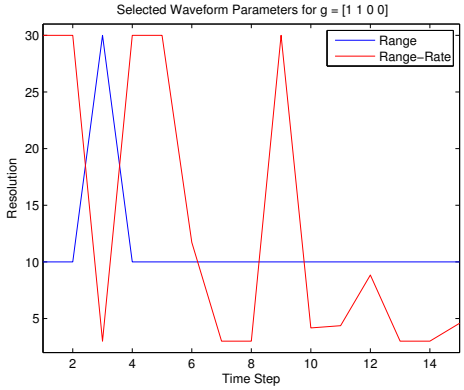


(b)

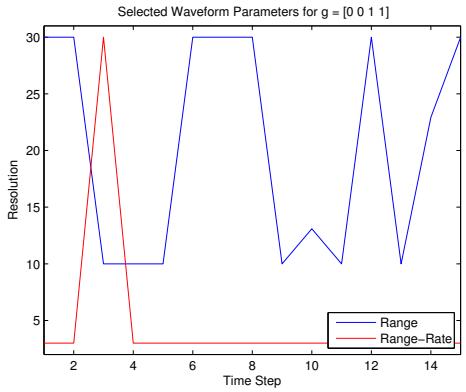


(c)

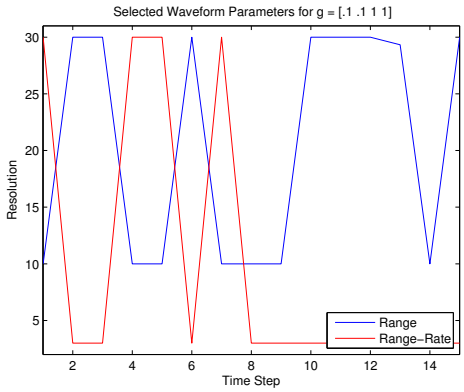
Figure 6.9: (a)MSE-EP performance is approached as the particle count and iteration count increase. (b) $N = 300$, 200 iterations. (c) $N = 500$, 1000 iterations.



(a)



(b)



(c)

Figure 6.10: Selected waveform parameters for one iteration of the simulation with various error weights.

6.2 Fixed Grid

The same simulation parameters as used in the previous section were also run in the fixed grid case with two changes. In the entire plane approach, there was a guaranteed minimum number of non-zero AF observations for each particle due to the adaptive grid spacing. In the fixed grid case, there is no such guarantee. As a result, much fewer delay-Doppler grid points actually contribute useful information to the state estimate at each time step. To counteract this, the number of particles was increased and the AF noise power was reduced: $N = 1000, \sigma^2_{AF} = 0.4$. Figures 6.11 to 6.14 show the results. Unlike the previous subsection, only the simulation results are shown, there are no predicted performance plots. However, similarities between the predictions for the entire plane case and the fixed grid simulation results are apparent.

Once again, the two primary results of this thesis persist. The adaptive waveform performance is at least as good as the best fixed waveform performance and, in some cases, exceeds the best fixed waveform performance. For the fixed grid case, some additional simulations were run to further examine cases where the adaptive performance clearly improves on the fixed performance. As previously mentioned, this occurs when the quantitative contributions of the position and velocity components of the error closely match each other, ruling out a simple choice of a position of velocity oriented waveform as a consistent best choice. This is largely a function of the error weight, which is used to scale the position and velocity contributions to the overall error. Figure 6.15 introduces two new error weights over two new Δt's to demonstrate that this condition can be found over a significant range of parameter choices. It is within this range that the benefit of the adaptive waveform selection algorithm is most pronounced.

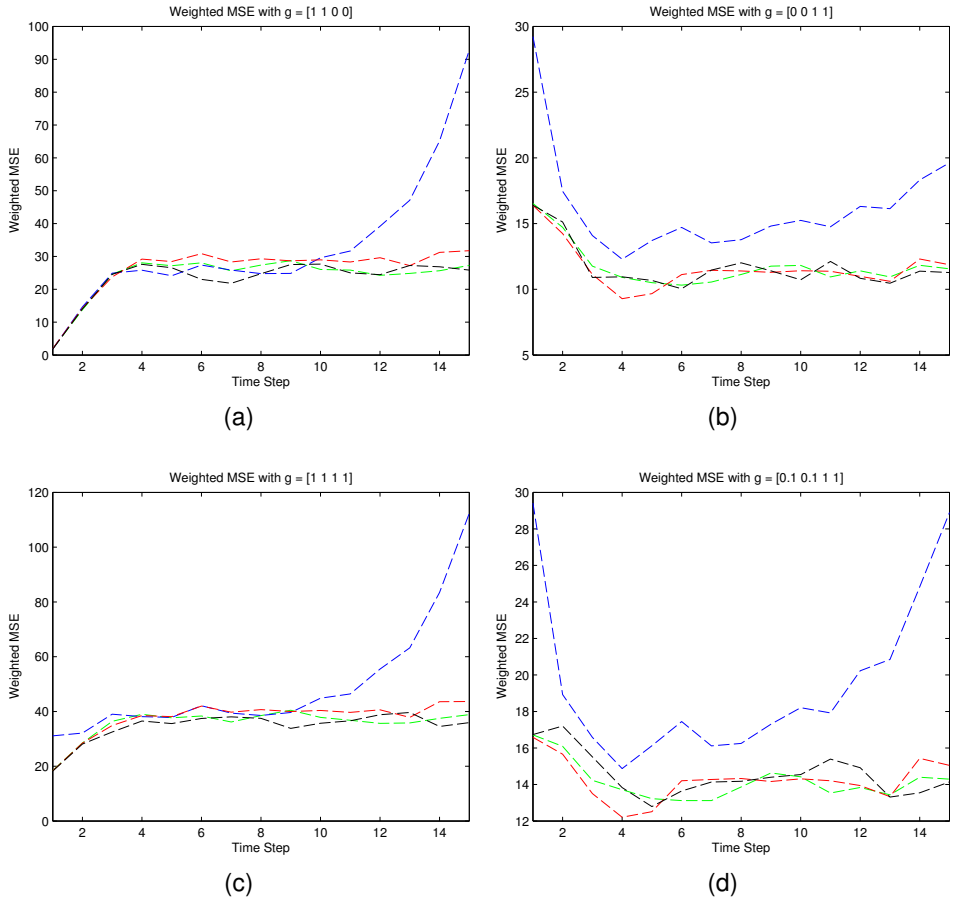


Figure 6.11: Fixed grid simulation results over 400 iterations for $N = 1000, \mathbf{C}_s = \text{diag}([1, 1, 16, 16]), \Delta t = 1$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

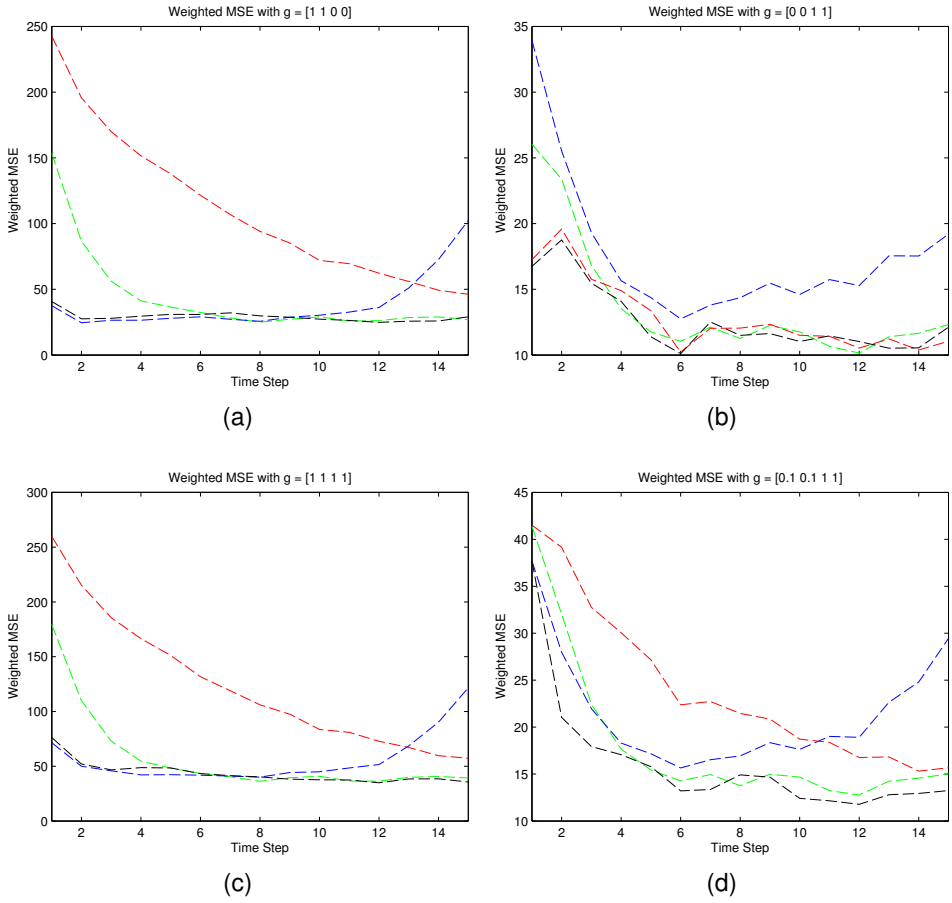


Figure 6.12: Fixed grid simulation results over 400 iterations for $N = 1000, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = 1$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

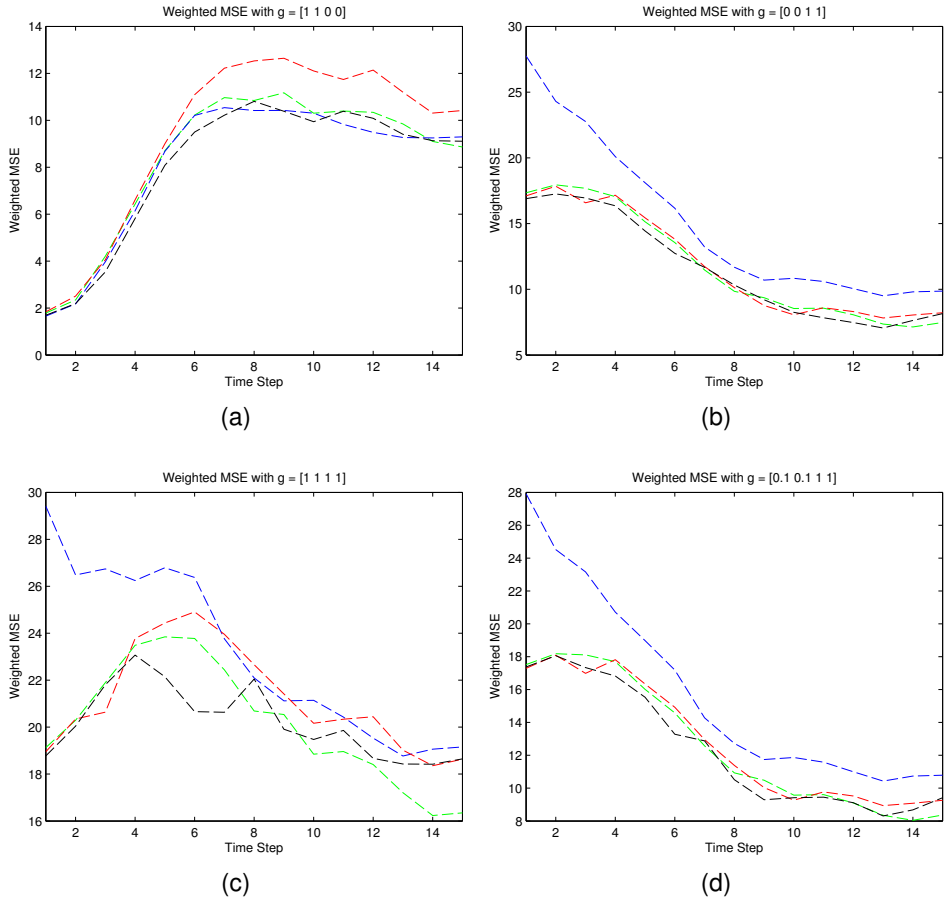


Figure 6.13: Fixed grid simulation results over 400 iterations for $N = 1000, \mathbf{C}_s = \text{diag}([1, 1, 16, 16]), \Delta t = .2$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

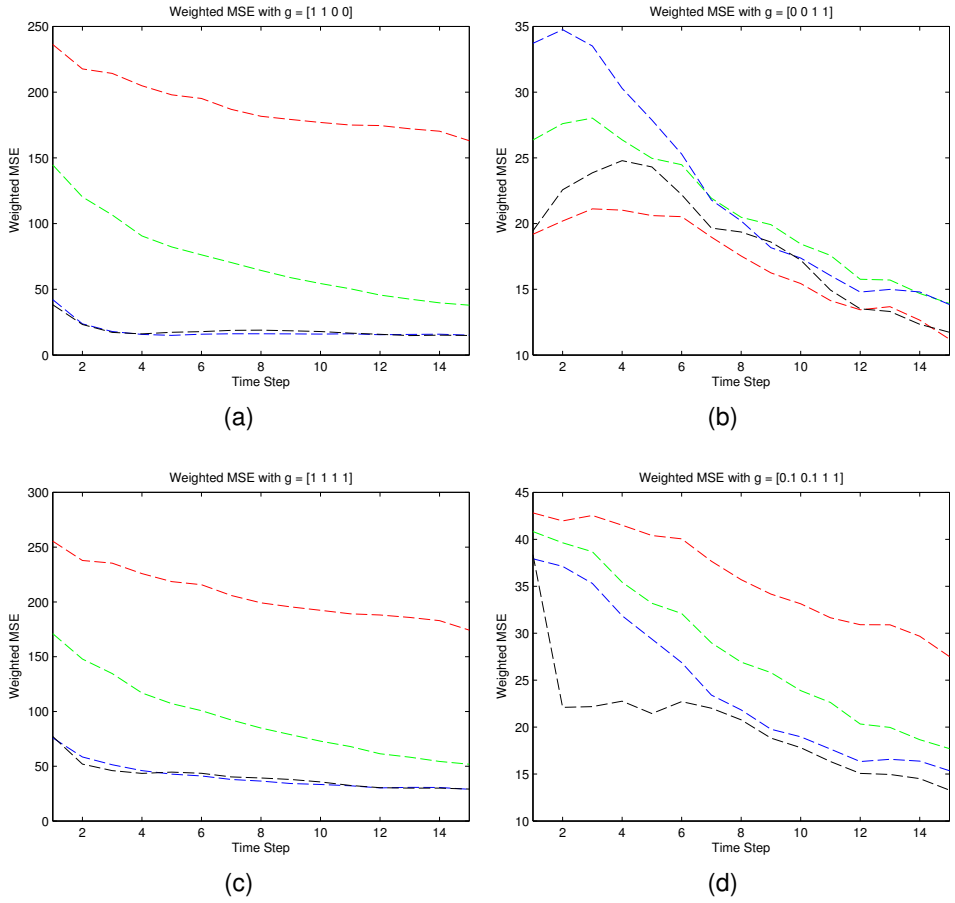


Figure 6.14: Fixed grid simulation results over 400 iterations for $N = 1000, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = .2$

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

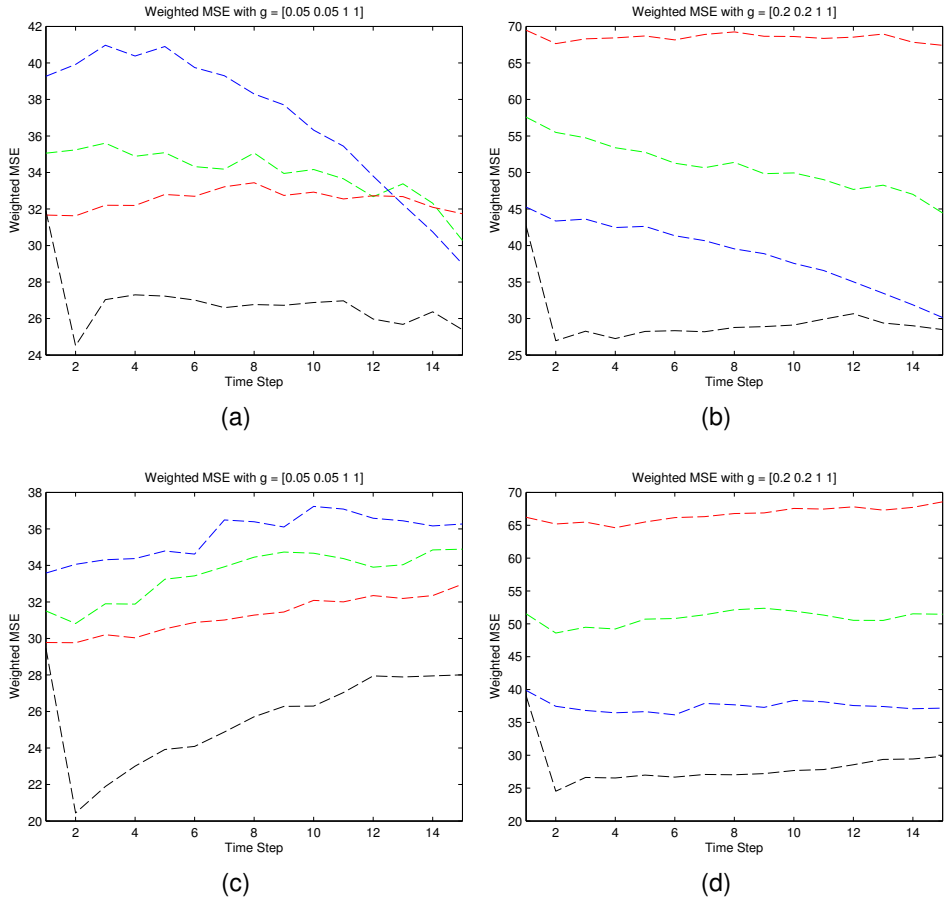


Figure 6.15: Fixed grid simulation results over 400 iterations for $N = 1000, \mathbf{C}_s = \text{diag}([255, 255, 16, 16]), \Delta t = .05$ (top) and $\Delta t = .02$ (bottom)

	Range Resolution	Range-Rate Resolution
Green	10 m	3 m/s
Blue	10 m	30 m/s
Red	30 m	3 m/s
Black	Adaptive	Adaptive

Chapter 7

CONCLUSIONS

In this thesis, we have examined one approach to tracking a 2-D target with an agile waveform track before detect radar system in the presence of high power white noise. Preliminary information on generic estimation of a hidden stochastic system was presented along with a description of a basic particle filter algorithm to carry out such estimation. The particular 2-D tracking problem under consideration here was then described in detail. Both the state process model for the unknown target state and the observation model used to make noisy measurements of that target state were presented. The need for a TBD approach making use of more than just the maximum AF measurement was explained as a means of being able to track under low SNR conditions. The complex envelopes of the LFM waveforms used in the simulations were characterized in terms of range and range-rate resolutions. The particle filter algorithm was then extended to incorporate the specifics of the observation model for two separate cases: an approximation to using the entire delay-Doppler plane involving a dynamic grid spacing, and a limited model using only a finite region of the delay-Doppler plane with a fixed grid spacing. The means of calculating the predicted covariance matrix for each waveform candidate at each time step was derived for the observation model being used, and the resulting recursive equation was used to estimate the predicted expected performance of the tracker as well as being an integral part of the on-line waveform selection algorithm. Finally, simulations of the tracking algorithm were run for both fixed waveforms and adaptive waveform selections and the results were compared to the predicted expected performance.

The primary results of this work are that two attributes of the agile waveform TBD algorithm presented here were predicted by the expected performance

estimates and were confirmed by the simulations. When only position or velocity estimates are required, we showed that TBD with fixed waveforms perform as well as the adaptive waveform TBD. When both position and velocity estimates are required, we showed that the adaptive waveform TBD outperformed the fixed waveform TBD. In particular, in cases where the error weight roughly "balances" the position and velocity contributions to the overall error, a single "best case" fixed waveform is difficult to find and the advantage of an adaptive algorithm becomes pronounced.

There are two immediate potential areas for future work involving this algorithm. First, the noise source in this thesis was simplified to the point of being independent of both waveform and grid spacing, neither of which would be true in a more accurate model. Incorporating a more accurate observation noise covariance and distribution could have a significant impact on the algorithm's performance. Second, only LFM waveforms were considered. The approach is conceptually easy to extend to a wide variety of additional waveform classes such as hyperbolic, exponential, etc. In practice, evaluating (5.16) in closed form for a particular waveform class can be non-trivial.

REFERENCES

[1] G. Welch and G. Bishop, "An introduction to the Kalman filter," TR 95-041, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599-3175, July 2006.

[2] M. S. 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*, vol. 50, February 2002.

[3] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.

[4] Y. Boers and J. N. Driessen, "Particle filter based detection for tracking," in *Proceedings of the American Control Conference*, pp. 4393–4397, 2001.

[5] C. Hue, J.-P. L. Cadre, and P. Perez, "Performance analysis of two sequential Monte Carlo methods and posterior Cramer-Rao bounds for multi-target tracking," *International Conference on Information Fusion*, pp. 464–473, 2002.

[6] D. J. Salmond and H. Birch, "A particle filter for track-before-detect," in *American Control Conference*, vol. 5, pp. 3755–3760, 2001.

[7] M. G. Rutten, B. Ristic, and N. J. Gordon, "Comparison of particle filters for recursive track-before-detect," in *International Conference on Information Fusion*, pp. 169–175, 2005.

[8] M. Rollason and D. Salmond, "A particle filter for track-before-detect of a target with unknown amplitude," in *IEEE Target Tracking: Algorithms and Applications*, vol. 14, pp. 1–4, 2001.

[9] M. G. Rutten, N. J. Gordon, and S. Maskell, "Recursive track-before-detect with target amplitude fluctuations," in *IEE Proceedings of Radar, Sonar and Navigation*, vol. 52, pp. 345–352, October 2005.

[10] Y. Boers, H. Driessen, J. Torstensson, M. Trieb, R. Karlsson, and F. Gustafsson, "A track before detect algorithm for tracking extended targets," tech. rep., THALES Nederland, B.V., Hengelo, The Netherlands.

[11] S. J. Davey, B. Cheung, and M. G. Rutten, "Track-before-detect for sensors with complex measurement," in *International Conference on Information Fusion*, (Seattle, WA), pp. 619–625, July 2009.

[12] S. Suvorova, S. D. Howard, W. Moran, and R. Evans, "Waveform libraries for radar tracking applications: Maneuvering targets," in *IEEE Conference on Information Sciences and Systems*, pp. 1424–1428, March 2006.

[13] S. Suvorova, S. D. Howard, and B. Moran, "Generalized frequency modulated waveform libraries for radar tracking applications," in *Asilomar Conference on Signals, Systems and Computers*, pp. 151–155, 2009.

[14] D. Cochran, S. Suvorova, S. D. Howard, and B. Moran, "Waveform libraries," *IEEE Signal Processing Magazine*, pp. 12–21, January 2009.

[15] S. Sira, Y. Li, A. Papandreou-Suppappola, D. Morrell, D. Cochran, and M. Rangaswamy, "Waveform-agile sensing for tracking: A review perspective," *IEEE Signal Processing Magazine*, vol. 26, pp. 53–64, January 2009.

[16] S. P. Sira, A. Papandreou-Suppappola, and D. Morrell, "Dynamic configuration of time-varying waveforms for agile sensing and tracking in clutter," *IEEE Transactions on Signal Processing*, vol. 55, pp. 3207–3217, July 2007.

[17] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on Signal Processing*, vol. 46, pp. 1386–1396, May 1998.

[18] M. L. Hernandez, A. D. Marrs, N. J. Gordon, S. R. Maskell, and C. M. Reed, "Cramer-Rao bounds for non-linear filtering with measurement origin uncertanity," *International Conference on Information Fusion*, pp. 18–25, 2002.

[19] J. J. Zhang, Q. Ding, S. Kay, A. Papandreou-Suppappola, and M. Rangaswamy, "Agile multi-modal tracking with dependent measurements," in *Asilomar Conference on Signals, Systems and Computers*, pp. 1653–1657, November 2010.

[20] J. J. Zhang, A. Papandreou-Suppappola, and M. Rangaswamy, "Multi-target tracking using multi-modal sensing with waveform configuration," in *IEEE International Conference on Acoustics Speech and Signal Processing*, (Dallas, TX), pp. 3890–3893, March 2010.

[21] H. L. V. Trees, *Detection, Estimation, and Modulation Theory Part III*. John Wiley & Sons, 2001.

[22] Y. Boers and J. N. Driessen, “Multitarget particle filter track before detect application,” in *IEE Proceedings - Radar, Sonar and Navigation*, vol. 151, pp. 351–357, 2004.

[23] R. C. Warren, “A Bayesian track-before-detect algorithm for IR point target detection,” DSTO-TR-1281, Weapons Systems Division, DSTO, Australia, 2002.

[24] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill, 2002.

[25] D. Morrell, “Filtering of stochastic processes.” Lecture notes for Stochastic Filtering, 2003.